Nonlinear System Identification using Structured Kernel Based Models

Tillmann Falck

Jury:
Prof. Dr. Yves Willems, chairman
Prof. Dr. Johan A.K. Suykens, promotor
Prof. Dr. Bart De Moor, co-promotor
Prof. Dr. Joos Vandewalle
Prof. Dr. Moritz Diehl
Prof. Dr. Joris De Schutter
Prof. Dr. Johan Schoukens
(Vrije Universiteit Brussel)
Dr. Kristiaan Pelckmans
(Uppsala University)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor in Engineering

April 2013

D/2013/7515/34
ISBN 978-94-6018-648-6

Preface

In the end, this thesis took longer than anticipated, but now it is finally coming to an end. Even though especially the last phase turned out to be quite difficult for me, when I am looking back, I am looking back to several years of fond memories.

The first time I was in Leuven was on my way back from a vacation at the Belgian coast. I told my friends in the car that I had seen a PhD position in the town we were just passing and whether it was okay to stop and have a look at the place for an hour. Everyone agreed and I certainly liked what I saw. Back at home I prepared an application and Johan was quick to invite me for an interview and to offer the position that finally led to this text. I really want to thank Johan not only for offering me that position in the first place, but also for continuously supporting me along the way as my promotor. He is always a source of valuable advice and always takes time to discuss one's problems. I would also like to thank my co-promotor, Bart De Moor. Even though I did not interact with him as closely as I did with Johan, I greatly appreciate the work environment he helped creating. He, among with the all the other professors at SISTA, makes sure that there is always enough funding and encourages everyone to use the many opportunities that the group is offering. Joos is not only the head of the group, he also is a huge driver towards interaction. I still remember a BBQ at his house and I really appreciate all the interaction arising from IAP dysco and its study days. The first time I met Johan Schoukens, I think, was during his Franqui lectures and I am really grateful for having him on my jury. He always provides good feedback and discussions with him are always interesting. It is also nice to know that wherever one goes in the

context of system identification, one can be sure that someone from Brussels is there as well, most often also Johan himself, be it a DYSCO workshop, the ERNSI or Benelux meetings, the CDC, SYSID or the IFAC WC. I would also like to thank Kristiaan, he alway encouraged me, was open to discussions and played an important role to get me started at SISTA along with Marcelo. Together with Johan he is the person who continuously helped me from my very first day at SISTA until the completion of this thesis. Joris De Schutter was one of the late additions to my jury, I really appreciate that he agreed to this position. Not only did he provide some valuable feedback to the text, but he is also genuinely interested in the, at times obscure methods, I came up with and sometimes seems to be more positive about their application than I am myself. Moritz Diehl also joined my jury as an additional member and provided extremely good feedback on parts of the thesis I was not sure anyone would read. Besides his formal involvement in my jury I really benefited from what he achieved within OPTEC. He always managed to invite interesting and renowned people to give lectures and seminars and stimulated interaction by organizing BBQs and retreats or just by introducing as many people to each other as possible. As final member of my jury, I would also like to thank Yves Willems for serving as chairman. Through his kind way of administrating the very final stages he makes sure that there is no additional pressure due to the unknown situation and I really appreciate this.

At SISTA all of the PhD students can really focus on their research and only have to do very little administration. This is due to the amazing help we receive behind the scenes from the administrative staff which I am really grateful for. At times it was still necessary to do some things ourselves but also then Ilse, John, Ida, Lut and others, were always helping to make these things run as smoothly as possible.

The work would really have been dull without all the guys in the tower. Marco is always a incredible resource on new ideas and recent advances, although I have to admit that I could not allows follow the level of mathematical abstraction he achieved in our discussions. Then there is the “window” row with Philippe, Kim, Pieter, Toni and later on Dries. Especially systems of polynomial equations remain to be a mystery for me, but it was always nice to work alongside and to travel with you. A good part of the “life” in the tower was certainly due to the Columbian gang, Mauricio, Fabian, Carlos, Julian and Marcelo (as Chilean associate member). Last but not least there is the rest of the lunch group, Tom, Erik, Kris, Siamak, Rocco and Maarten. It was always a pleasure to be working with you and discuss research as well as the world over lunch.

I would also like to thank other people that made my time in Leuven as enjoyable as it was, Leif, Joachim, Dennis Lin, Aga, the imec group with Victor, Pawel and Sylwia, Arno, Denis and Jenny, Angel, Bart, Joachim, and Jörg and Friederike. The same holds for all the guys at ERNSI which make it a pleasure to work on system identification.

Then there are my friends from Germany, most of which I already know from school, Robert and Zhao Jun, Tilman and Mareike, Dennis and Henrike, Matthias and Sandra, Benno, Oliver, and Katharina. They are still talking to me, even though I often set work before going to Germany and attending a party. Now guys, I cannot hide behind this thesis anymore, feel free to remind me that I should become more active again.

The Chemnitz group still tolerates me, even though I have always been working on something related to this thesis whenever we met and sometimes even kept Anne from meeting you at all. Moving to Stuttgart was so easy because we did not have to look for friends, but friends were already there. To a large extent this is due to Corinna acting as a multiplier. Besides helping us settling in in Stuttgart I would like to thank Corinna for trying to kick my butt and making me finish this thesis as well as keeping Anne happy while I was in a bad mood.

Finally I would like to thank my parents and my brother, without your support I could not have done it. You were always there for me when I needed it and never questioned what I was doing. Mama, dies ist wohl der einzige Satz dieser Arbeit, den du lesen kannst. Ich möchte mich bei dir und Papa ganz doll bedanken. Ihr seid immer für mich da gewesen und habt mich immer unterstützt. Ohne euch hätte ich diese Arbeit weder angefangen noch zu Ende bringen können. Vielen Dank! Then there is Anne. Thanks a lot for staying at my side and trying to support me where you could. You were more patient with me than I could possibly expect. Thanks a lot for giving me the space and the time I needed and thanks for the time you invested proof reading this text, even though it is not the most thrilling text. Thanks for taking care of me.

Abstract

This thesis discusses nonlinear system identification using kernel based models. Starting from a least squares support vector machine base model, additional structure is integrated to tailor the method for more classes of systems. While the basic formulation naturally only handles nonlinear autoregressive models with exogenous inputs, this text proposes several other model structures. One major goal of this work was to exploit convex formulations or to look for convex approximations in case a convex formulation is not feasible.

Two key enabling techniques used extensively within this thesis are overparametrization and nonquadratic regularization. The former can be utilized to handle nonconvexity due to bilinear products. During this work overparametrization has been applied to handle new model structures. Furthermore it has been integrated with other techniques to handle large data sizes and a new approach to recover a parametrization in terms of the original variables has been derived. The latter technique, nonquadratic regularization, is also suitable to construct convex relaxations for nonconvex problems. In this context the major contribution of this thesis is the derivation of kernel based model representations for problems with nuclear norm as well as group-ℓ_1 norm regularization.

In terms of new or improved model structures, this thesis covers a number of contributions. The first considered model class are partially linear models which combine a parametric model with a nonparametric one. These models achieve a good predictive performance while being able to incorporate physical prior knowledge in terms of the parametric model part. A novel constraint significantly reduces the variability of the parametric model part. The second

part of this thesis, that exploits structure to identify a more specific model class, is the estimation of Wiener-Hammerstein systems. The main contributions in this part are a thorough evaluation on the Wiener-Hammerstein benchmark dataset as well as several improvements and extensions to the existing kernel based identification approach for Hammerstein systems.

Besides targeting more restricted model structures also several extensions of the basic model class are discussed. For systems with multiple outputs a kernel based model has been derived that is able to exploit information from all outputs. Due to the reliance on the nuclear norm, the computational complexity of this model is high which currently limits its application to small scale problems. Another extension of the model class is the consideration of time dependent systems. A method that is capable of determining the times at which a nonlinear system switches its dynamics is proposed. The main feature of this method is that it is purely based on input-output measurements. The final extension of the model class considers linear noise models in combination with a nonlinear model for the system. This work proposes a convex relaxations to estimate the noise model as well as a model capturing the system dynamics by solving a joint convex optimization problem.

The final contribution of this thesis is a reformulation of the classical least squares support vector formulation that allows the analysis of existing models with respect to their sensitivity to perturbations on the inputs.

Nomenclature

Abbreviations

(N)FIR	(Nonlinear) finite impulse response model (cf. Tables 2.1, 2.2)
(N)ARX	(Nonlinear) autoregressive model with exogenous input (cf. Tables 2.1, 2.2)
(N)BJ	(Nonlinear) Box-Jenkins model (cf. Table 2.1, 2.2)
(N)ARMAX	(Nonlinear) autoregressive moving average model with exogenous input (cf. Tables 2.1, 2.2)
(N)OE	(Nonlinear) output error model (cf. Tables 2.1, 2.2)
SVD	Singular value decomposition [Golub and Van Loan, 1996]
MIMO	Multiple input multiple output system
MISO	Multiple input single output system
SISO	Single input single output system
SVM	Support Vector Machine
LS-SVM	Least Squares Support Vector Machine
RKHS	Reproducing kernel Hilbert space
OLS	Ordinary least squares

KKT	Karush-Kuhn-Tucker (conditions for optimality, c.f. Chapter 3)
RBF	Radial basis function (kernel, c.f. Table 4.1)
RMSE	Root mean squared error $\left(= \sqrt{\frac{1}{N} \sum_{t=1}^{N} (y_t - \hat{y}_t)^2}\right)$
QP	Quadratic programming (problem)
SOCP	Second order cone programming (problem)
SDP	Semidefinite programming (problem)

Symbols & Notation

$\boldsymbol{x}$, $\boldsymbol{\psi}$	Bold face small letters are (column) vectors
$\boldsymbol{X}$, $\boldsymbol{\Psi}$	Bold face capitals are matrices
$x(t)$	Signal (function of time) with $x : \mathbb{T} \to \mathbb{R}$, where $\mathbb{T}$ is either $\mathbb{Z}$ or $\mathbb{R}$ for discrete and continuous time signals respectively
x_k	Either value of signal $x(t)$ at time $t = k$ or the k-th element of vector $\boldsymbol{x}$
X_{ij}, $(\boldsymbol{X})_{ij}$	The ij-th value of $\boldsymbol{X}$
N, M	Capitals are constants unless denoted otherwise
$\hat{x}$, $\hat{x}(t)$, $\hat{\boldsymbol{x}}$, $\hat{\boldsymbol{X}}$	Estimates of a value, a signal, a vector and a matrix respectively
$\boldsymbol{a}^T$, $\boldsymbol{A}^T$	Transposes of $\boldsymbol{a}$ and $\boldsymbol{A}$ respectively
$\boldsymbol{X}^{-1}$	Matrix inverse of $\boldsymbol{X}$
$\boldsymbol{X}^{\dagger}$	Moore-Penrose pseudo inverse of $\boldsymbol{X}$ [Golub and Van Loan, 1996]
$K(\boldsymbol{x}, \boldsymbol{y})$	Positive definite kernel function
(a, b)	Tuple
$\{x_1, \dots, x_N\}$	Set
$[x_1, \dots, x_N]$	Row vector
$[\boldsymbol{A}; \boldsymbol{B}]$	Concatenation of two matrices (or vectors) along the first dimension (vertical concatenation)
$[\boldsymbol{A}, \boldsymbol{B}]$	Concatenation of two matrices (or vectors) along the second dimension (horizontal concatenation)
$[x_i]_{i=1}^N$	Element-wise definition of a vector in $\mathbb{R}^N$ whose i-th element is x_i
$\{x_i\}_{i=1}^N$	Element-wise definition of a set with N elements
$\mathbf{1}_N$	A N-dimensional vector of all ones
$\mathbf{0}_N$	A N-dimensional vector of all zeros
$\boldsymbol{I}_N$	The identity matrix of size N
$\mathbf{0}_{\boxtimes}$	A matrix of all zeros with compatible dimensions

$\mathbb{N}, \mathbb{R}, \mathbb{R}_+$	Natural numbers, real numbers and positive real numbers
z^{-k}	Time shift operator, $z^{-k} f(t) = f(t-k)$
$\succeq, \succ, \preceq, \prec$	Conic inequalities, if $\boldsymbol{x}, \boldsymbol{y} \in C$ where C is a cone then $\boldsymbol{x} \succeq \boldsymbol{y} \Leftrightarrow \boldsymbol{x} - \boldsymbol{y} \in C$ and $\boldsymbol{x} \succ \boldsymbol{y} \Leftrightarrow \boldsymbol{x} - \boldsymbol{y} \in \mathrm{int}(C)$ where $\mathrm{int}(C)$ is the interior of C If $\boldsymbol{x} \in \mathbb{R}^N$ and no cone is specified the inequalities are implicitly with respect to the nonnegative orthant $\mathbb{R}_+^N \cup \{\mathbf{0}_N\}$ and positive orthant $\mathbb{R}_+^N$ respectively, i.e. element-wise inequalities, $\boldsymbol{x} \succeq \boldsymbol{y} \Leftrightarrow x_i \geq y_i$ for $i = 1, \ldots, N$ If $\boldsymbol{x} \in \mathbb{R}^{N \times N}$ and no cone is specified the inequalities are implicitly with respect to the cone of positive semidefinite and positive definite matrices respectively
$\lVert\boldsymbol{x}\rVert_p$	Vector p-norm, $\lVert\boldsymbol{x}\rVert_p = \left(\sum_{i=1}^N \lvert x_i\rvert^p\right)^{\frac{1}{p}}$ for $\boldsymbol{x} \in \mathbb{R}^N$
$\lVert\boldsymbol{X}\rVert_F$	Frobenius norm, $\lVert\boldsymbol{X}\rVert_F = \left(\sum_{i,j} X_{ij}^2\right)^{\frac{1}{2}}$
$\lVert\boldsymbol{X}\rVert_2$	Operator or spectral norm, largest singular vector of $\boldsymbol{X}$
$\lVert\boldsymbol{X}\rVert_*$	Nuclear or trace norm, sum of singular vectors
$\frac{\partial}{\partial x}$	Partial derivate with respect to x
$\frac{\partial}{\partial \boldsymbol{x}}$	Gradient with respect to $\boldsymbol{x}$
∂	Subgradient
$\partial_{\boldsymbol{x}}$	Subgradient with respect to $\boldsymbol{x}$

Contents

Introduction 1

The main topics of this thesis are well described by its title "nonlinear system identification using structured kernel based models". This can be broken down into three main components,

1. nonlinear system identification,
2. kernel based models and
3. structure.

The central theme is system identification, which describes the process of obtaining a model based on measured data. Access to a model is crucial in many situations. One of the main applications is in control, regardless whether the control is manual or automatic. Another important use case for models is in analyzing and understanding a system. System identification for linear systems is a well-established field with a broad selection of methods as well as a deep understanding of their properties and limitations. However, most real systems show nonlinear behavior, which cannot be captured by linear models. As such, the class of nonlinear systems is much larger than that of linear systems. Yet, the field of nonlinear system identification is still in its infancy. Even certain subclasses of the full class of nonlinear systems with attractive properties such as systems with smooth nonlinearities still contain a vast amount of complicated behaviors. Most classic techniques in nonlinear system identification are basically a form of function estimation or regression using a mathematical model. The limitation of this approach is that most of these models do not relate in any way to the system that they ought to represent. This has two major drawbacks. First of all, it is difficult to incorporate any form of prior knowledge into the model. An example for prior knowledge could be the applicability of a physical law for part of system or information on its stability. Even though some effect might be well understood, this knowledge cannot be provided to the model, but the model has to rediscover it from the data. This is a waste of resources and results in suboptimal models as the information contained in the data could have been used for further refining the model. Second, once a model has been estimated, no or only very limited information on the system it represents can be extracted. Whereas in linear systems, one can connect the frequency response or other parameters like time constants to physical concepts, there are no such equivalents for most nonlinear modeling techniques. A large part of this thesis is therefore devoted to providing some known tools from linear identification in a nonlinear context.

All methods proposed in this thesis are derived from kernel based models. In particular the core formulation is using least squares support vector machines (LS-SVMs) [Suykens, Van Gestel, et al., 2002; Suykens et al., 2010] which have been shown to be a powerful technique for nonlinear regression and beyond. One main advantage of this methodology is its versatility, which is evident from its many applications besides regression, such as classification, unsupervised and semi-supervised learning and dimensionality reduction. A key aspect for this success is the formulation in a primal-dual framework and the choice of a least squares loss. The latter greatly simplifies the derivation of models and allows concentrating on the model formulation. The former provides an ideal environment to incorporate additional structure as model representations can be specified very explicitly in the primal. The derivation of a form suitable for numerical estimation is often straightforwardly solved by stating its dual.

A major contribution to the success of support vector techniques in general is their reliance on convex optimization. This assures that global solutions to the formalized optimization problems can be found in an efficient manner. This thesis profits even more from the field of convex optimization as several recently proposed powerful heuristics can be tailored to system identification problems.

1.1 Challenges

Challenges tackled in this thesis all relate to the identification of nonlinear systems employing kernel based models using a LS-SVM core and the complications arising in this context.

Nonlinear behavior poses complex problems as it contains a vast amount of effects compared to linear systems. Due to this large space of potentially relevant models, it is hard to find suitable model representations. Furthermore, the parametrization of these models quickly gives rise to nonlinear optimization problems, which are prone to local minima and therefore suboptimal solutions. The challenge is therefore selecting good model structures that on the one hand allow the representation of nonlinear dynamics and on the other hand are formulated in a fashion that admits an efficient numerical solution.

Large amounts of data are often accessible for problems in system identification. For many systems data can be acquired with relatively large sampling rates, providing a wealth of quantitative information. As averaging techniques are usually not suitable for nonlinear behavior, other techniques have to be considered. This is especially important as the complexity of the employed kernel based techniques scales cubically in the number of data samples. Therefore efficient techniques are necessary that allow utilizing the wealth of available data.

Incorporating prior information is important to come up with the best possible model by combining the prior knowledge with the information contained in the data. However, coming from a purely data driven approach it is not always straightforward how prior knowledge can be incorporated into the estimation problem. For every kind of prior knowledge, one has to look anew how to facilitate this information to acquire an improved model. Often the resulting estimation problems are more complicated and either cannot be solved exactly or at least require additional effort to be solved. Therefore next to the modeling challenge encountered when incorporating prior knowledge one regularly obtains further complications in numerical problem solving.

Model representations are crucial for any identification technique. Without a suitable model representation, the model cannot be utilized. Two key aspects are model structure and model parametrization. Kernel based techniques in a primal-dual setting have the advantage that they usually start off with a model parametrization that allows a straightforward integration of model structure. The model structure is any information that affects the model itself, i.e. a different model structure will in general give rise to a model generating different predictions. However, changing the parametrization of the model does not change the model itself but might merely be easier to work with for particular tasks. The initial model parametrization often suffers from two drawbacks. First, the models are given in a parametric form, which for many popular choices of the kernel function is unsuitable for solution due to the very high dimensionality of the problem. Second, the parametric model description may contain additional constraints on the model behavior which are not embodied in the model equation itself. These constraints are an integral part of the model structure as they dictate part of the model behavior. In classical kernel based models these complexities can be countered by switching to the nonparametric kernel based parametrization. This parametrization has the advantage that all information on the model structure is embedded in a single predictive equation. However, the derivation of the kernel based parametrization is only straightforward as long as the regularization term is quadratic. In this thesis, model representations in the presence of a nonquadratic regularization term have to be derived.

Numerical solution is essential for the applicability of any practical method. Besides the basic problem of handling complexities resulting from large data sets, more fundamental problems are encountered when relying on recent regularization techniques. The current trade-off is between (i) ease of implementation, (ii) numerical precision and (iii) rate of convergence. Each of these aspects is important to come up with a method that can be used in practice. The relative importance for a particular application can vary, though.

1.2 Objectives

The objective to advance nonlinear system identification based on least squares support vector machines can be divided into several key components.

Extension to more model classes The first objective is to extend the basic formulation of LS-SVMs to cover more classes of systems. The ones to be implemented are multiple output systems, time varying systems and systems with more complex noise structures.

Improving model performance The second objective is to incorporate prior information, thus improving the model performance. In practice the systems to be identified are rarely complete black boxes. Therefore exploring means to facilitate this information is important.

Convex formulations The third objective is to retain as much convexity from the basic formulation as possible. The addition of structure as mandated by the two prior objectives often results in nonconvex estimation problems. Hence, the goal is to find relaxations or approximations that allow the recovery of good solutions based on convex optimization techniques.

Validation on realistic data The last objective is to validate the proposed methods on realistic data. All models contain approximations and simplifications, which need to be verified on representative data. This allows an analysis of strengths as well as weaknesses of a particular approach.

1.3 Overview of chapters

The thesis is structured into two parts. The first part gives a brief introduction to the theoretical background required for this thesis. The original work can be found in Part II, which starts with Chapter 5. A short chapter by chapter overview is given in the following.

Chapters 2–4 Chapter 2 briefly summarizes key concepts in the area of system identification, e.g. parametric vs. nonparametric models and white box vs. black box modeling. The following chapter outlines some fundamental concepts of convex optimization that will be utilized later on in the text. The last chapter of Part I finally introduces least squares support vector machines and a few related techniques crucial for the remainder of this thesis.

Chapter 5 outlines partially linear systems, which are a particular type of nonlinear systems. These models combine a linear-in-parameters parametric model with a nonparametric model. Their advantage lies in situations in which good parametric models already exist. The chapter extends the classical formulation with a novel constraint that decouples the estimation of the two model parts. This removes an ambiguity which otherwise can result in large variabilities of the individual model estimates.

Chapter 6 extends the classical LS-SVM formulation for regression to models with multiple related outputs. This is achieved by introducing an advanced regularization scheme based on the nuclear norm. The main complications tackled in this chapter are the derivation of the dual nonparametric kernel based model as well as the expression of the predictive model in terms of the dual solution. Furthermore, some important properties of the underlying optimization problem are studied as well as methods for its numerical solution.

Chapter 7 presents the identification of a class of structured nonlinear systems, called Wiener-Hammerstein systems. These systems consist of two linear dynamical blocks at the input and output respectively, which sandwich a static nonlinearity. A convex relaxation scheme for their estimation within a kernel based framework is presented. This estimation scheme is then adapted for large data sets. After a discussion of projection schemes for recovering the original model class from its relaxation, the results from the previous chapter are applied for an improved relaxation. Finally, the proposed methods are compared on a benchmark data set.

Chapter 8 augments the basic nonlinear model given by LS-SVM with a linear parametric noise model. The use of a noise model is necessary in case the model residuals are correlated and can improve the prediction performance in these cases. The chapter proposes a relaxation scheme similar to that in Chapter 7 to jointly estimate the nonlinear system dynamics along with the linear noise model. Special attention is given to the projection onto the original model class as two independent estimates for the noise model are obtained.

Chapter 9 studies the sensitivity of LS-SVM based models with respect to unstructured perturbations. The analysis employs a worst case approximation and is based on a second order cone programming problem. This change of the regularization requires some changes to the derivation of the dual problem and the predictive model. To control the numerical complexity, the robustified model is cast back into least squares form.

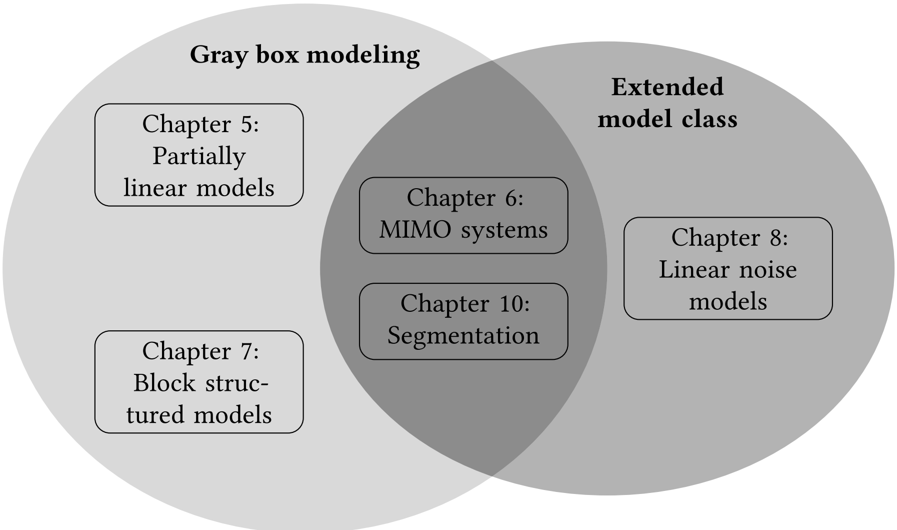

Figure 1.1: Clusters of chapters with respect to system identification topics

Based on the derived formulations, the sensitivity of simple models with respect to input variables and employed kernel functions is studied.

Chapter 10 presents a method for the offline segmentation of data generated by a nonlinear systems with abrupt changes of system dynamics. The estimation is once more based on a convex relaxation and uses advanced regularization. As in the previous chapters using nonquadratic penalties, the work to obtain a finite-dimensional kernel based model representation and the corresponding predictive equation is presented. Due to the time dependent nature of the model, the model selection is considered explicitly. Furthermore a scheme for a more efficient numerical solution is presented.

Finally the thesis is concluded in Chapter 11.

1.4 Guide through the chapters

The thesis covers different aspects of related problems. Two main points of entry can be identified. The first way chapters can be selected is based on the problem they are solving. From this point of view, the chapters can be grouped as shown in Figure 1.1. There is one cluster of chapters studying gray

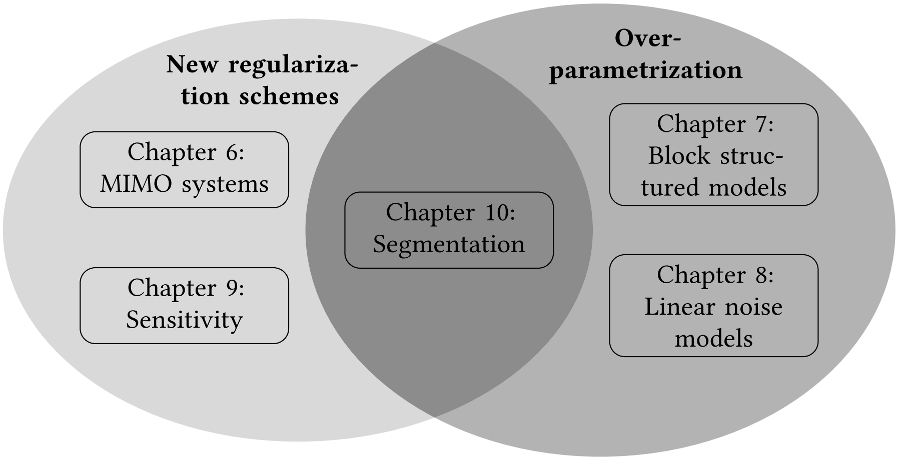

Figure 1.2: Clusters of chapters with respect to techniques for deriving convex approximation for nonconvex problems.

box models. Within this cluster the chapters can be read almost independently and the selection can be determined by the interest of the reader. In general Chapter 5 is a good entry point as it gives an overview of the common methodology, while requiring relatively few mathematical derivations. Section 7.4 is based on ideas more thoroughly discussed in Chapter 6, however otherwise Chapter 7 can be read independently. Moreover the mathematics used in Chapter 6 is quite similar to that in Chapter 10, but the presentation slightly different proving access from a different direction.

The second cluster of chapters depicted in Figure 1.1 contains approaches extending the model class. As can be seen in the figure, Chapters 6 and 10 can be attributed to both clusters. Although belonging to the other cluster, it is advisable to read Chapter 7 before Chapter 8 as the employed methodology is much more thoroughly presented in the former.

A second approach for selecting chapters of interest and a suitable order of reading is from a methodological point of view. Besides grouping the chapters according to their modeling goal, it can be interesting to cluster them based on the employed fundamental ideas. Such an arrangement is shown in Figure 1.2. Chapter 5 is intentionally left out of this representation, as the two main concepts for convex approximation used in this thesis are not exploited in this chapter. The remaining chapters revolve around the idea of overparametrization – the introduction of independent variables to model bilinear terms – and the technique of using convex norms as surrogates for nonconvex functions.

The concept of overparametrization and many ideas revolving about its integration with kernel based models are most thoroughly discussed in Chapter 7. Hence, for the cluster on overparametrization this should be the first chapter to read. In Chapter 8 the same idea is applied to a different problem. The main benefit from a methodological point of view is a more detailed numerical analysis of the attained convex approximation versus the true global optimum. Chapter 10 uses the idea of overparametrization in a more extreme setting. It does not relax a bilinear product but introduces new model parameters at each time instant. This only succeeds as the idea of overparametrization is combined with a suitably crafted regularization scheme. For exactly this reason Chapter 10 can be attributed to both clusters. Within the cluster on regularization schemes, Chapter 9 provides a straightforward introduction to nonquadratic regularization terms and the resulting complications for obtaining predictive models. With the most level of detail the topic is discussed in Chapter 6. Chapter 10 can be considered complementary as it treats a mathematically very similar problem but takes a slightly different approach of presenting them.

1.5 Contributions of this work

The main contributions of this work are summarized in the following.

Wiener-Hammerstein identification Wiener-Hammerstein systems are structured systems which consist of linear dynamical blocks and a single static nonlinear function that captures all nonlinearity. The prior knowledge about the system structure is used to improve the model performance. The contributions of this thesis are: (i) the extension of Hammerstein identification as proposed by Goethals et al. [2005b] to Wiener-Hammerstein systems, (ii) an improved methodology to recover the original model class by a new projection scheme, (iii) an extension to handle large data sets and (iv) a thorough evaluation on a large benchmark data set.

- Falck, T., Pelckmans, K., Suykens, J. A. K., and De Moor, B. (July 2009). “Identification of Wiener-Hammerstein Systems using LS-SVMs”. In: *Proceedings of the 15th IFAC Symposium on System Identification.* (Saint-Malo, France, July 6–8, 2009), pp. 820–825,

- Falck, T., Dreesen, P., De Brabanter, K., Pelckmans, K., De Moor, B., and Suykens, J. A. K. (Nov. 2012). "Least-Squares Support Vector Machines for the Identification of Wiener-Hammerstein Systems". In: *Control Engineering Practice* 20(11), pp. 1165–1174,
- Goethals, I., Pelckmans, K., Falck, T., Suykens, J. A. K., and De Moor, B. (2010). "NARX Identification of Hammerstein Systems using Least-Squares Support Vector Machines". In: *Block-oriented Nonlinear System Identification.* Ed. by F. Giri and E.-W. Bai. Vol. 404. Lecture notes in control and information sciences. Springer. Chap. 15, pp. 241–256.

Partially linear systems Partially linear systems combine parametric and nonparametric models. This allows incorporating prior information and yields models with improved performance. The novel contribution is an orthogonality constraint which simplifies the model estimation. It ensures a significantly reduced variability of the obtained parametric model estimate compared to existing techniques.

- Falck, T., Signoretto, M., Suykens, J. A. K., and De Moor, B. (2010). *A two stage algorithm for kernel based partially linear modeling with orthogonality constraints.* Tech. rep. 10-03. ESAT-SISTA, K.U. Leuven.

Parametric noise models For an accurate prediction of a system output, it is often necessary to model the noise structure as well as the system itself. The contribution of this thesis is a convex approach to jointly estimate a linear parametric noise model along with a nonlinear model for the system.

- Falck, T., Suykens, J. A. K., and De Moor, B. (Dec. 2010). "Linear Parametric Noise Models for Least Squares Support Vector Machines". In: *Proceedings of the 49th IEEE Conference on Decision and Control.* (Atlanta, GA, USA, Dec. 15–17, 2010), pp. 6389–6394.

Nonquadratic regularization Recent advances in convex optimization provide powerful heuristics for convex relaxations. This thesis picks up several of these approximations to improve system identification related problems. The main contributions in this context are (i) the derivation of dual, finite-dimensional, kernel based optimization problems, (ii) model representations in terms of the kernel and the dual model parameters and (iii) approaches for the numerical solution of the resulting formulations. This is done for several application areas.

Multiple output systems Based on nuclear norms, the basic LS-SVM model is extended to handle systems with more than one output. The use of the advanced regularization scheme enables information transfer from one system output to another.

Wiener-Hammerstein systems Wiener-Hammerstein systems as introduced before are cast into a particular form of multiple output systems. In this case, an intermediate signal takes the role of the multiple system outputs. This yields a more accurate representation of the original model class and therefore improves the convex relaxation proposed earlier.

- Falck, T., Suykens, J. A. K., Schoukens, J., and De Moor, B. (Dec. 2010). "Nuclear Norm Regularization for Overparametrized Hammerstein Systems". In: *Proceedings of the 49th IEEE Conference on Decision and Control.* (Atlanta, GA, USA, Dec. 15–17, 2010), pp. 7202–7207.

Time varying systems By using sum-of-norms regularization, it is possible to connect groups of variables. This enables linking time dependent parameters of a system. This results in a problem formulation that allows the detection of points in a time series at which the underlying system changes its dynamics.

- Falck, T., Ohlsson, H., Ljung, L., Suykens, J. A. K., and De Moor, B. (Aug. 2011). "Segmentation of time series from nonlinear dynamical systems". In: *Proceedings of the 18th IFAC World Congress.* (Milan, Italy, Aug. 28–11, 2011), pp. 13209–13214.

Sensitivity of kernel based models The use of unsquared ℓ_2-norms instead of their squared counterparts in standard LS-SVMs allows the application of results from robust linear modeling. Based on these results, LS-SVM derived models can be analyzed with respect to their sensitivity towards the selection of the kernel function and their input variables.

- Falck, T., Suykens, J. A. K., and De Moor, B. (Dec. 2009). "Robustness analysis for Least Squares Kernel Based Regression: an Optimization Approach". In: *Proceedings of the 48th IEEE Conference on Decision and Control.* (Shanghai, China, Dec. 16–18, 2009), pp. 6774–6779.

First principle information Prior information on systems is often given in terms of physical relations. These are usually formulated in terms of differential equations. The possibility to use analytic derivatives of the model during its estimation allows information provided in terms of differential equations to be fused with measured data.

- Mehrkanoon, S., Falck, T., and Suykens, J. A. K. (July 2012b). "Parameter Estimation for Time Varying Dynamical Systems using Least Squares Support Vector Machines". In: *Proceedings of the 16th IFAC Symposium on System Identification.* (Brussels, Belgium, July 11–13, 2012), pp. 1300–1305,
- Mehrkanoon, S., Falck, T., and Suykens, J. A. K. (Sept. 2012a). "Approximate Solutions to Ordinary Differential Equations Using Least Squares Support Vector Machines". In: *IEEE Transactions on Neural Networks and Learning Systems* 23(9), pp. 1356–1367.

Evaluation of benchmark data A time series prediction benchmark problem contained three data sets from unknown sources. Based on a manual analysis of these time series very competitive results could be obtained. These results are based on the combination of several extensions of LS-SVMs.

- Espinoza, M., Falck, T., Suykens, J. A. K., and De Moor, B. (Sept. 2008). "Time Series Prediction using LS-SVMs". In: *Proceedings of the European Symposium on Time Series Prediction.* (Porvoo, Finland, Sept. 17–19, 2008), pp. 159–168.

Applications outside of system identification Besides work in the context of system identification, similar algorithmic problems can be found in other domains. On several occasions the technical expertise acquired in this thesis was contributed to other problems.

- Yu, S., Falck, T., Daemen, A., Tranchevent, L.-C., Suykens, J. A. K., De Moor, B., and Moreau, Y. (2010). "L2-norm multiple kernel learning and its application to biomedical data fusion". In: *BMC Bioinformatics* 11(309), pp. 1–53,
- Ojeda, F., Falck, T., De Moor, B., and Suykens, J. A. K. (July 2010). "Polynomial componentwise LS-SVM: fast variable selection using low rank updates". In: *Proceedings of the International Joint Conference on Neural Networks 2010.* (Barcelona, Spain, July 18–23, 2010), pp. 3291–3297,

- Van Herpe, T., Mesotten, D., Falck, T., De Moor, B., and Van den Berghe, G. T. (Feb. 2010). “LOGIC-Insulin Algorithm for Blood Glucose Control in the ICU: a pilot test”. At: Third International Conference on Advanced Technologies & Treatments for Diabetes (Basel, Switzerland, Feb. 10–13, 2010).

Part I

Foundations

System identification

2

Many engineering applications rely on the concept of a *system* as shown in Figure 2.1. To be useful in an engineering context one needs a *model* for the system. Once a model has been obtained it can be used for different purposes, such as analysis, prediction or control. This thesis is about constructing such a model for a particular class of systems within a certain framework. The goals of this chapter are to

- explain the class of considered systems and place it into a context,
- highlight the key concepts that are required to understand the properties of the chosen framework and finally
- introduce the theory that is later on needed in the thesis.

Some distinctions that will be explained later on are those between white box and black box models, linear versus nonlinear systems and parametric and nonparametric techniques.

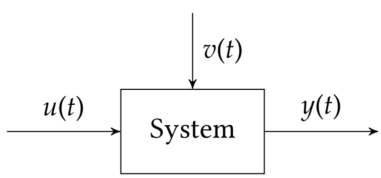

Figure 2.1: Dynamic system with input $u(t)$ and output $y(t)$ where t denotes time. The system is subject to a disturbance $v(t)$.

2.1 System properties

The block structure shown in Figure 2.1 is very general and can be used to represent many phenomena, depending on the precise definitions of the system, input, output and disturbance. The term *system* is actually imprecise because, in this thesis, it always refers to a *dynamical system*. The main characteristic of a dynamical system is that it has a memory. As such its output y at time t_0 in general depends on its input $u(t)$ for t from $-\infty$ to t_0.

In the scope of this thesis only lumped systems are considered in contrast to distributed systems. Whereas lumped systems can be described by a finite set of parameters and often can be modeled with ordinary differential equations, distributed systems have an infinite number of parameters and are usually described by partial differential equations. Throughout this thesis it is assumed that input $u(t)$, output $v(t)$ and disturbance are real valued. The case of discrete or complex valued variables is not considered.

In most chapters the presentation is targeted to systems with a single input $u(t)$ and a single output $y(t)$ (SISO systems), although the generalization to multiple input and single output (MISO) is usually straightforward. An exception in this respect is Chapter 6 which explicitly considers multiple input multiple output (MIMO) systems. With the exception of Chapter 10 it is assumed everywhere that the system is time invariant, i.e. that the system itself does not depend on the time t. All presented material implicitly assumes that the time variable t is discrete and uniformly sampled. Strictly speaking this is a property of the model and not of the system however.

The most important classification for systems relevant to this thesis is the distinction between linear and nonlinear systems. The primary goal is to construct models for nonlinear systems but in almost all cases there is some relation to linear systems. Comprehensive information relevant for linear systems can be found in [Kailath, 1980; Oppenheim et al., 1997], a good reference on nonlinear systems and their theory is [Khalil, 2002].

2.2 Prior information

Apart from the system properties outlined in the previous section, which are mostly governed by physics, a crucial point for choosing the modeling technique is the information available on the system. On the extreme sides of the spectrum are white box modeling and black box modeling. The term white box modeling is used in case the system is modeled using physical insight and based on physical laws. This approach is limited to problems where the physics are well understood. It often requires a large amount of domain specific expert knowledge and can be very time consuming if the system is complex. White box modeling often results in systems of differential equations. Depending on the application these might need to be discretized later on.

Black box modeling on the other side of the spectrum does not need any physical insight about the system, instead it tries to infer all information from measured data. This data-driven approach to modeling is what is usually referred to by the term *system identification*. For this to be feasible it of course has to be possible to take measurements of the system. Depending on the application, taking measurements might be expensive or time consuming and often is both. A historical overview of system identification is given by Gevers [2006] and some relations to earlier work in statistics and econometrics are presented by Deistler [2002]. A good overview of the key aspects of system identification is given by Ljung [2010], for comprehensive information the main references are [Söderström and Stoica, 1989; Ljung, 1999].

In between white box models and black box models there is a whole spectrum of so-called gray box models. Depending on the particular shade of gray, these might be physical models for which some parameters are unknown and have to be estimated from data. A much darker shade of gray would be structural information. In both cases the deviation from a pure color can introduce new problems. In case of black box models for example it is not always straightforward how prior knowledge about a system can be exploited.

In the overview paper on system identification by Ljung [2010] several core concepts are defined. These will be introduced in the following and related to different parts of this thesis.

- The first concept is the *model*, which is defined as "a relationship between observed quantities". In this thesis each chapter will describe a methodology to establish such a relationship.

- The next concept is that of a *true description*. This is a useful tool to prove statistical properties of a certain model, but will not be used further in this text.

- A more important concept in the context of this work is *information*, which on the one hand is described as the prior information and on the other hand is about the information contained in the data. Prior information has been introduced in this section and will play a major role in this thesis. For example in Chapters 5 and 7 prior structural information is used to estimate dark gray models. The information contained in the data is not explicitly addressed in this thesis and always assumed to be rich enough to carry out the estimation.

- A closely related concept is the *model class*. The choice of the model class is strongly influenced by prior information. Often considered model classes for linear as well as nonlinear systems are introduced in the next section. In this thesis the model class is either enriched as in Chapters 6 and 8 or restricted as in Chapters 5 and 7 depending on prior information.

- Having defined a particular model class, the next concept is *estimation*. The estimation of a model that explains given data is the key problem addressed in all chapters. Estimating a model often relies on solving optimization problems. In this thesis the focus is on convex problems for which some aspects are introduced in Chapter 3.

- Strongly related to estimation is the concept of *complexity*. The complexity of a model describes its versatility to explain different behaviors. One way to control the complexity of a model heavily used within this thesis is regularization. Regularization is a key concept in least squares support vector machines, the framework used for modeling throughout this thesis and described in Chapter 4. In Chapters 6 and 10 new complexity measures based on improved regularization schemes are considered.

- The estimation step is usually followed by *validation*. This step ensures that the model does not only fit the data that it was estimated on, but also generalizes to new data. According to Occam's razor [Rasmussen and Ghahramani, 2001] less complex models usually generalize better.

- Finally the last core concept according to Ljung [2010] is *model fit*. The model fit quantifies how well a model fits a given data set. In this work usually a simple least squares criterion is used, but one might benefit from an application specific choice [Gevers and Ljung, 1986; Gevers, 2005]. In Chapters 5, 9 & 10 model fit is compromised to better accommodate other objectives.

2.3 Model representation

To construct a model for the system shown in Figure 2.1 a representation needs to be chosen. The most popular ways to represent a system in an engineering setting are state space models, the behavioral approach and the representation of systems as filters. Among these three, the behavioral approach [Willems, 2007] is a particular case as, in contrast to most other representations, it does not consider inputs and outputs. Therefore, strictly speaking, it does not correspond to the structure in Figure 2.1. It rather models the interaction between variables and is particularly well suited for white box modeling and consequently will not be considered further. In the following subsections the remaining two model representations will be briefly introduced, namely models in state-space and in polynomial form.

2.3.1 State-space models

A representation that gained immense popularity in the control community due to the work of Kalman [1960b,a] is the state space representation. For a linear time invariant system with n-dimensional input $\boldsymbol{u}(t) \in \mathbb{R}^n$ and m-dimensional output $\boldsymbol{y}(t) \in \mathbb{R}^m$, a state space model [Ljung, 1999, Eq. 4.84] can be stated as

$$\boldsymbol{x}(t+1) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{w}(t), \tag{2.1a}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) + \boldsymbol{v}(t), \tag{2.1b}$$

where $\boldsymbol{x} \in \mathbb{R}^d$ is the d-dimensional state of the system. The matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ are of compatible dimensions and describe the dynamics of the system. The term $\boldsymbol{w}(t) \in \mathbb{R}^d$ is called process noise, whereas $\boldsymbol{v}(t) \in \mathbb{R}^m$ is called measurement noise. The noise terms are usually characterized through their covariance matrices $\boldsymbol{R}_1 = \mathcal{E}\{\boldsymbol{w}(t)\boldsymbol{w}(t)^T\}$, $\boldsymbol{R}_2 = \mathcal{E}\{\boldsymbol{v}(t)\boldsymbol{v}(t)^T\}$ and $\boldsymbol{R}_3 = \mathcal{E}\{\boldsymbol{v}(t)\boldsymbol{w}(t)^T\}$. Nonlinear versions can be stated as

$$\boldsymbol{x}(t+1) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{w}(t)), \tag{2.2a}$$

$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{v}(t)), \tag{2.2b}$$

with $\boldsymbol{f} : \mathbb{R}^{2d+n} \to \mathbb{R}^d$ and $\boldsymbol{g} : \mathbb{R}^{d+2m} \to \mathbb{R}^m$. The state-space representation is very popular for many applications as states can often be associated with physical quantities. Also this representation handles MIMO systems in a very natural fashion.

2.3.2 Polynomial or difference equation models

Whereas the memory of a system for state-space models is given by the state, one can also model a system as a *filter*. The filter takes past values of the output $y(t)$ and input $u(t)$ and relates them to new outputs. In this case the memory of the system is contained in the past values of the output. Therefore an alternative form to model a linear time invariant system is given by combining past values of its input $u(t)$ and its output $y(t)$ as in

$$y(t) = \sum_{k=0}^{q} b_k u(t-k) - \sum_{k=1}^{p} a_k y(t-k) + e(t) \tag{2.3}$$

where the a_k's and b_k's are coefficients that define the model while p and q are the model orders. The term $e(t)$ represents noise and can be characterized by its probability density function. Introducing the time shift operator $\mathscr{z}$ defined as $\mathscr{z}^{-1} f(t) = f(t-1)$ where f is an arbitrary function of time, the equation can be rewritten as

$$\left(1 + \sum_{k=1}^{p} a_k \mathscr{z}^{-k}\right) y(t) = \left(\sum_{k=0}^{q} b_k \mathscr{z}^{-k}\right) u(t) + e(t). \tag{2.4}$$

Defining two polynomials in $\mathscr{z}$, $A(\mathscr{z}) = 1 + \sum_{k=1}^{p} a_k \mathscr{z}^{-k}$ and $B(\mathscr{z}) = \sum_{k=0}^{q} b_k \mathscr{z}^{-k}$, the model equation can be further simplified to $A(\mathscr{z})y(t) = B(\mathscr{z})u(t) + e(t)$. Note that the model is completely determined by the polynomials $A(\mathscr{z})$ and $B(\mathscr{z})$. This particular model structure is called autoregressive model with exogenous input (ARX). The description is valid only as long as the noise process $e(t)$ is independent. In case the noise is correlated, more complicated model structures have been proposed. These can all be unified in a general polynomial model structure [Ljung, 1999, Eq. 4.33], visually represented in Figure 2.2,

$$A(\mathscr{z})y(t) = \frac{B(\mathscr{z})}{F(\mathscr{z})} u(t) + \frac{C(\mathscr{z})}{D(\mathscr{z})} e(t), \tag{2.5}$$

where $A(\mathscr{z})$, $B(\mathscr{z})$, $C(\mathscr{z})$, $D(\mathscr{z})$ and $F(\mathscr{z})$ are polynomials of the form introduced above. With the exception of $B(\mathscr{z})$ all of these polynomials are monic. Depending on which polynomials differ from unity, these structures are given different names. The simplest and most common model structures are *Finite Impulse Response* (FIR) and *AutoRegressive with eXogenous input* (ARX). Some others are *AutoRegressive Moving Average with eXogenous input* (ARMAX), *Box-Jenkins* (BJ) and *Output Error* (OE). Table 2.1 lists the nonunity polynomials for these model structures.

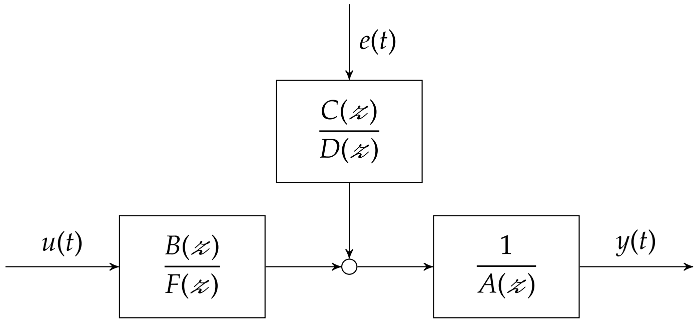

Figure 2.2: General structure of a linear time invariant system in polynomial form.

Table 2.1: Model structures for linear dynamic time invariant systems in polynomial form as in (2.5) and Figure 2.2 [Ljung, 1999, Table 4.1]. Polynomials that are not mentioned are equal to 1.

MODEL STRUCTURE	NONUNITY POLYNOMIALS
FIR	$B(\mathscr{z})$
ARX	$A(\mathscr{z}), B(\mathscr{z})$
ARMAX	$A(\mathscr{z}), B(\mathscr{z}), C(\mathscr{z})$
BJ	$B(\mathscr{z}), F(\mathscr{z}), C(\mathscr{z}), D(\mathscr{z})$
OE	$B(\mathscr{z}), F(\mathscr{z})$

The extension to nonlinear models of the polynomial model structures is not as straightforward as it is for state space models. To generalize the polynomial model structure, one defines a regressor vector $\varrho(t)$ that contains all elements needed to compute $y(t)$ and a parameter vector $\boldsymbol{\theta}$ that contains all model parameters. For the ARX model defined by (2.3) these are $\varrho(t) = [y(t-1), \dots, y(t-p), u(t), \dots, u(t-q)]^T$ and $\boldsymbol{\theta} = [a_1, \dots, a_p, b_0, \dots, b_q]^T$ such that it can be written as

$$y(t) = \boldsymbol{\theta}^T \varrho(t) + e(t). \tag{2.6}$$

The transition from linear to nonlinear systems is then achieved by replacing the linear function $\boldsymbol{\theta}^T \varrho(t)$ by a nonlinear one $f(\varrho(t))$ where $f : \mathbb{R}^{p+q+1} \to \mathbb{R}$ such that

$$y(t) = f(\varrho(t)) + e(t). \tag{2.7}$$

Table 2.2: Model structures for nonlinear dynamic time invariant systems specified as $y(t) = f(\varrho(t)) + e(t)$ [Sjoberg et al., 1995]. The table lists the variables that are present in the regression vector $\varrho(t)$.

MODEL STRUCTURE	VARIABLES ALLOWED IN REGRESSOR VECTOR
NFIR	$u(t)$
NARX	$u(t)$, $y(t)$
NARMAX	$u(t)$, $y(t)$, $\epsilon(t)$
NBJ	$u(t)$, $\hat{y}(t)$, $\epsilon(t)$, $\epsilon_u(t)$
NOE	$u(t)$, $\hat{y}_u(t)$

By Takens' theorem most nonlinear systems can be represented in this way under mild conditions [Takens, 1981; Kantz and Schreiber, 2003].

The nonlinear model in (2.7) is the nonlinear generalization of the ARX model shown in (2.3) and accordingly denoted as NARX. To obtain a generalization of the general model structure in (2.5) to nonlinear models, the regressor vector ϱ needs to be extended with variables beyond past input and output measurements. To reach this goal one also considers the one-step-ahead predictor

$$\hat{y}(t) = f(\varrho(t)) \tag{2.8}$$

and additionally a simulation predictor $\hat{y}_u(t)$. The difference between the one-step-ahead predictor $\hat{y}(t)$ and the simulation predictor $\hat{y}_u(t)$ is that the regressor vector ϱ for the former contains measured values for y while for the latter one these are replaced by their previously obtained predictions $\hat{y}_u$. Using these predictions one can further define the prediction error

$$\epsilon(t) = y(t) - \hat{y}(t) \tag{2.9}$$

and the prediction error in simulation mode $\epsilon_u(t) = y(t) - \hat{y}_u(t)$ respectively. Using these definitions Sjoberg et al. [1995] classify nonlinear models in a fashion corresponding to the linear polynomial models. Depending on which variables out of $u(t)$, $y(t)$, $\hat{y}(t)$, $\hat{y}_u(t)$, $\epsilon(t)$ and $\epsilon_u(t)$ are included in the regression vector ϱ, the nonlinear model structures are named in analogy to their linear counterparts. The model structures and their regression variables are summarized in Table 2.2.

2.4 Model parametrization and estimation

The last section introduced different model classes but did not touch the problem of estimating a model from data. A very natural approach is to define an optimization problem that tries to maximize a model fit subject to a model class. Therefore one needs to choose a model class, a parametrization for that model class and the model fit. To simplify the presentation, only linear models are considered in the beginning. Since polynomial models are most relevant for this thesis, they are considered first. Note that the coefficients of the polynomials completely characterize a model. Therefore one can collect these coefficients in a parameter vector $\boldsymbol{\theta}$. Such models are also called parametric models, as they are described in terms of much less parameters than the number of measurement data. On the other hand there are nonparametric models for which the number of parameters is in the same order of magnitude as the number of data. An example for nonparametric models are frequency domain models. These models are made up by frequency response functions. Their estimation considers each frequency value as one, often independent, parameter. Note that polynomial models and frequency response functions can be related via the Fourier transform.

No matter how a model is parametrized, for each choice of the parameters $\boldsymbol{\theta}$ one can compute the estimate $\hat{y}(t, \boldsymbol{\theta})$ at time t. Then one can estimate a model with

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}, \varepsilon_t} \sum_{t=1}^{N} V(\epsilon_t) \quad \text{subject to} \quad \varepsilon(t) = y(t) - \hat{y}(t, \boldsymbol{\theta}). \tag{2.10}$$

Solving this optimization problem estimates the model parameters $\boldsymbol{\theta}$ given a dataset $\{u(t), y(t)\}_{t=1}^{N}$. Here the function $V : \mathbb{R} \rightarrow \mathbb{R}_+$ is a loss function penalizing prediction errors. This general scheme is called prediction error framework and was introduced by Åström and Bohlin [1965]. Depending on the assumptions on the noise term $e(t)$, the model structure and the loss function V, the solution of (2.10) yields the maximum likelihood estimate. Solving the optimization problem is highly nontrivial except for particular choices of model structure and loss function. A special case are FIR and ARX models and the least squares loss $V(\epsilon) = \epsilon^2$ for which the estimation problem can be solved using least squares. More information, mostly in the context of linear systems in parametric form, can for example be found in [Ljung, 1999; Söderström and Stoica, 1989]. For models specified in the frequency domain the main reference is [Pintelon and Schoukens, 2001]. Estimation of nonlinear systems within this framework is discussed in [Sjoberg et al., 1995; Juditsky et al., 1995; Nelles, 2001].

For state space models a natural parametrization are the coefficients of the system matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$. However such a parametrization gives rise to potentially very difficult nonconvex optimization problems. Among other things, their solutions are not unique and only defined up to a similarity transform on the state. The state of a system and its evolution are very powerful tools to look at system dynamics. Therefore this description is very popular for example in control. Also system properties like stability, observability and controllability can be directly connected to and checked upon its state space description. It is also the key element in a Kalman filter [Kalman, 1960a] which allows the online reconstruction of the system state. Initially state space descriptions were derived from impulse responses or Markov parameters, their generalization to MIMO systems, if they were not obtained from first principles modeling. This is known as realization theory and was pioneered by Ho and Kalman [1966] in the deterministic setting and Akaike [1974] in the stochastic one.

A relatively recent approach to identify state space models without the need for an intermediate model or the direct measurement of Markov parameters is subspace identification. In contrast to (2.10) it does not start from an optimization problem but relies on a combination of system theoretic insights and linear algebra. The idea is to factor suitably defined matrices of input and output measurements in a way that allows the reconstruction of a state sequence or the extended observability matrix. From either one of these, the parameter matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ can be straightforwardly estimated. In case of a reconstructed state sequence for example one can apply least squares to the set of equations in (2.1) to obtain these estimates. The first comprehensive monograph on this topic is [Van Overschee and De Moor, 1996] while a more recent presentation incorporating additional material has been published by Katayama [2005].

A noteworthy parallel between state space models and the modeling methodology considered further on during this thesis are its hybrid nature when considering its parametrization. While the final state space model is parametric, the entire estimation process is nonparametric. The models that will be considered later on start from an implicit parametric description and yield nonparametric models in the end. Another similarity is that both approaches strongly rely on linear algebra and were made possible due to advances in other scientific fields. In case of subspace identification these were numerical linear algebra and a deeper understanding of system theory while for the kernel based models considered from now on, the main influences are convex optimization and machine learning. The crucial concepts for both will be outlined in the following two introductory chapters.

3 Convex optimization

Optimization is at the heart of every system identification problem as it involves fitting some parameters to measured data. Depending on the cost function, i.e. the optimization criterion and the structure of the model, the optimization problem can be nonlinear. In general one can use first order methods like conjugate gradient or second order techniques like the Newton method to solve these problems [Nocedal and Wright, 2006; Bertsekas, 1999]. A common problem in nonlinear optimization is the presence of local optima which may be found instead of the global optimum. Therefore an important class of optimization problems are convex optimization problems [Boyd and Vandenberghe, 2004; Nesterov, 2004]. For these every local optimum is also globally optimal and for certain classes at least one can be found efficiently. Furthermore, for a large class of convex problems called strictly convex, the optimum is unique.

The most widely solved convex problems are linear programs (LPs) that are popular since the 1950s due to the simplex algorithm by Dantzig in 1947 [Dantzig, 1963]. The simplex algorithm only applies to LPs, hence other convex problems lagged behind for a long time. This started to change with the introduction of a new algorithm known as *interior point method* for LPs by Karmarkar [1984]. This algorithm has been extended to large class of convex problems by Nesterov and Nemirovskii [1994]. Based on these ideas efficient and robust algorithms in the form of general purpose solvers have been developed. These are one of the main reasons for the success of convex optimization in the last two decades. In this thesis they have also been employed for various

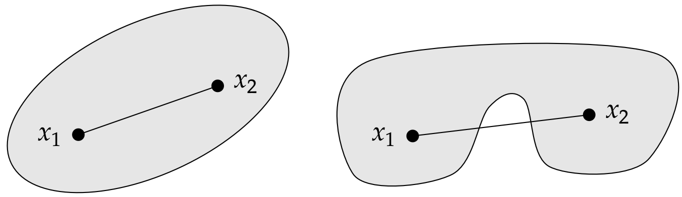

Figure 3.1: Illustration of a convex set (left panel) and a nonconvex set (right panel).

problems. However for more complex convex problems like *second order cone programs* (SOCP) and *semidefinite programs* (SDPs) general purpose software does not always scale well with the number of unknowns. Therefore one section of this chapter is dedicated to a recently popular approach using first order gradient techniques. A recurring problem throughout this thesis is that the problems that have to be solved are intrinsically nonconvex. Instead of relying on nonlinear optimization software, one goal of this thesis is to explore how far one can get by considering convex approximations and relaxations. A short overview of possible techniques to come up with similar convex problems is given in Section 3.5.

3.1 Basic definitions and notation

The canonical form of an optimization problem is

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & g_i(\boldsymbol{x}) \leq 0, \quad i = 1, \dots, M. \end{aligned} \tag{3.1}$$

Now if both $f(\boldsymbol{x})$ as well as all $g_i(\boldsymbol{x})$'s are convex the optimization problem is said to be convex. To verify that a set C is convex one has to check that the line segment between two arbitrary points $\boldsymbol{x}_1, \boldsymbol{x}_2 \in C$ is entirely contained in C. Formally

$$\boldsymbol{x}_1 + \gamma(\boldsymbol{x}_2 - \boldsymbol{x}_1) \in C \tag{3.2}$$

has to hold for all $0 \leq \gamma \leq 1$ and $\boldsymbol{x}_1, \boldsymbol{x}_2 \in C$. Examples for simple convex and nonconvex sets are shown in Figure 3.1. Some convex sets are

- hyperplanes $\{\boldsymbol{x} \mid \boldsymbol{a}^T \boldsymbol{x} + b = 0\}$,

- halfspaces $\{\boldsymbol{x} \mid \boldsymbol{a}^T\boldsymbol{x} + b \leq 0\}$ and
- norm balls, for the norm $\|\cdot\|_p$ with $p \geq 1$ the corresponding norm ball of size r is $\{\boldsymbol{x} \mid \|\boldsymbol{x}\|_p \leq r\}$.

A special type of convex sets are convex cones. A set is a cone if for every point $\boldsymbol{x} \in C$ all points $\gamma\boldsymbol{x}$ for $\gamma \geq 0$ are also in the set C. A compact condition for C being a convex cone is

$$\gamma_1\boldsymbol{x}_1 + \gamma_2\boldsymbol{x}_2 \in C \tag{3.3}$$

for all $\gamma_1, \gamma_2 \geq 0$ and $\boldsymbol{x}_1, \boldsymbol{x}_2 \in C$. Examples for convex cones are

- the nonnegative orthant of dimension d, $\mathbb{R}^d_+ = \{\boldsymbol{x} \mid \boldsymbol{x} \in \mathbb{R}^d, x_i \geq 0 \text{ for } i = 1, \dots, d\}$,
- norm cones, for the norm $\|\cdot\|_p$ with $p \geq 0$ the corresponding norm cone is $K_p = \{(\boldsymbol{x}, t) \mid \|\boldsymbol{x}\|_p \leq t\}$,
- especially the second order cone, also called Lorentz or ice-cream cone, that corresponds to the Euclidian or L_2-norm and
- the semidefinite cone, $S^d_+ = \{\boldsymbol{X} \mid \boldsymbol{X} \in \mathbb{R}^{d\times d}, \boldsymbol{X} = \boldsymbol{X}^T, \boldsymbol{X} \succeq \boldsymbol{0}\}$.

A function $f : \mathbb{R}^d \to \mathbb{R}$ is called convex if the domain D of f is a convex set and

$$f(\gamma\boldsymbol{x}_1 + (1-\gamma)\boldsymbol{x}_2) \leq \gamma f(\boldsymbol{x}_1) + (1-\gamma)f(\boldsymbol{x}_2), \tag{3.4}$$

for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in D$ and $0 \leq \gamma \leq 1$. Geometrically that means that the function must be below the line segment between any two points of its graph. This is illustrated in Figure 3.2.

3.2 Convex problems

A constrained optimization problem in standard form is given by

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & \boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0}_{M_e} \\ & \boldsymbol{h}(\boldsymbol{x}) \leq \boldsymbol{0}_{M_i} \end{aligned} \tag{3.5}$$

where $\boldsymbol{x} \in \mathbb{R}^N$, $f : \mathbb{R}^N \to \mathbb{R}$, $\boldsymbol{g} : \mathbb{R}^N \to \mathbb{R}^{M_e}$ and $\boldsymbol{h} : \mathbb{R}^N \to \mathbb{R}^{M_i}$. The *Lagrangian* of this problem is a function that incorporates the objective function and the constraints. Based on it, several important results can be derived.

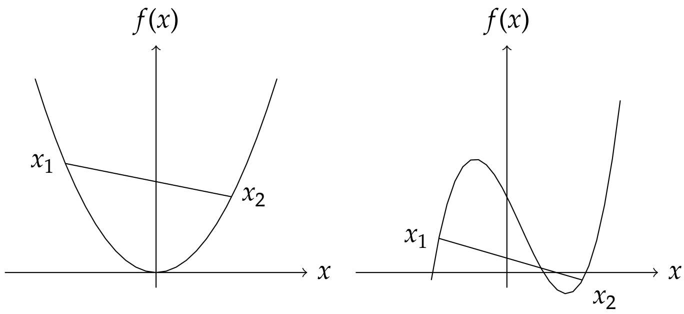

Figure 3.2: Illustration of a convex function (left panel) and a nonconvex function (right panel).

Table 3.1: Classification of convex optimization problems. The constraint set of all optimization problems can be complemented with linear equality constraints $\boldsymbol{Ax} = \boldsymbol{b}$. The matrix $\boldsymbol{P}$ is assumed to be positive semidefinite.

NAME	OBJECTIVE FUNCTION	CONSTRAINT SET
LP	$\boldsymbol{c}^T\boldsymbol{x} + d$	$\boldsymbol{Dx} \leq \boldsymbol{e}$
QP	$\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Px} + \boldsymbol{c}^T\boldsymbol{x} + d$	$\boldsymbol{Dx} \leq \boldsymbol{e}$
SOCP	$\frac{1}{2}\boldsymbol{x}^T\boldsymbol{Px} + \boldsymbol{c}^T\boldsymbol{x} + d$	$\boldsymbol{Dx} \leq \boldsymbol{e}$, $\lVert\boldsymbol{x}\rVert_2 \leq \boldsymbol{f}^T\boldsymbol{x} + g$
SDP	$\sum_{n=1}^{N} c_n x_n + d$	$\boldsymbol{Q}_0 + \sum_{n=1}^{N} \boldsymbol{Q}_n x_n \preceq 0$

These will be briefly summarized in this subsection as they are important for the remainder of this thesis. The Lagrangian is given by

$$\mathscr{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f(\boldsymbol{x}) + \boldsymbol{\nu}^T\boldsymbol{g}(\boldsymbol{x}) + \boldsymbol{\lambda}^T\boldsymbol{h}(\boldsymbol{x}) \tag{3.6}$$

with $\boldsymbol{\nu} \in \mathbb{R}^{M_e}$, $\boldsymbol{\lambda} \in \mathbb{R}^{M_i}$ and $\boldsymbol{\lambda} \succeq \boldsymbol{0}_{M_i}$. The most important concept following from this definition is the *Lagrange dual* — or just dual — function defined as

$$d(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\boldsymbol{x}} \mathscr{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}). \tag{3.7}$$

Note that the Lagrange dual function is the pointwise minimum of an affine function in $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$. Therefore it is always convex, even if the objective function or some of the constraints are not. An important property of the Lagrange dual function is that it provides a lower bound for the objective function. As this hold for any feasible $\boldsymbol{x}$ it also holds for the optimal one, hence one has a lower bound on the optimal value of the objective function $f(\boldsymbol{x})$, further on denoted as f^*. Based on the Lagrange dual function one can formulate the Lagrange dual problem — or simply dual problem or dual — as

$$\begin{aligned} &\max_{\boldsymbol{\lambda},\boldsymbol{\nu}} \quad d(\boldsymbol{\lambda},\boldsymbol{\nu}) \\ &\text{subject to} \quad \boldsymbol{\lambda} \geq \boldsymbol{0}_{M_i}. \end{aligned} \tag{3.8}$$

Denote the optimal value of the dual problem by d^*, then one can define the *duality gap* $f^* - d^*$. For primal problems (3.5) that are convex this quantity is often zero. A sufficient condition for it to be zero is Slater's condition [e.g. Boyd and Vandenberghe, 2004]. For problems that have a zero duality gap one says that strong duality holds. Among other things the duality gap is important for algorithms as it provides a powerful stopping criterion. If it is zero, one is guaranteed to have found optimal values for $\boldsymbol{x}$, $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$. In case strong duality holds, one can show a property known as *complementary slackness* which states that $\lambda_i^* h_i(\boldsymbol{x}^*) = 0$ for $i = 1, \dots, M_i$ where $\boldsymbol{x}^*$ and $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ denote primal and dual optimal solutions respectively.

While in some cases the derivation of dual functions is possible just by means of its definition (3.7), for problems with differentiable objective and differentiable constraints it is usually much easier to do so by considering the *Karush-Kuhn-Tucker* conditions for optimality, or shortly the KKT conditions. The KKT conditions combine the primal constraints, the dual constraints and complementary slackness with the condition that the gradient of the Lagrangian $\mathscr{L}$ in $\boldsymbol{x}$ has to vanish at an optimal solution $\boldsymbol{x}^*$. The complete set of conditions is given by

$$\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0}_{M_e}, \tag{3.9a}$$

$$\boldsymbol{h}(\boldsymbol{x}) \leq \boldsymbol{0}_{M_i}, \tag{3.9b}$$

$$\boldsymbol{\lambda} \geq \boldsymbol{0}_{M_i}, \tag{3.9c}$$

$$\lambda_i h_i(\boldsymbol{x}) = 0, \quad i = 1, \dots, M_i, \tag{3.9d}$$

$$\nabla f(\boldsymbol{x}) - \sum_{i=1}^{M_e} \nu_i \nabla g_i(\boldsymbol{x}) - \sum_{i=1}^{M_i} \lambda_i \nabla h_i(\boldsymbol{x}) = \boldsymbol{0}_N \tag{3.9e}$$

For a problem that has strong duality any optimal solution solution $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ satisfies the KKT conditions. If the problem is additionally convex then also the reverse holds and the KKT conditions provide a sufficient condition for optimality. Using these conditions it is often easier to derive a compact closed form expression for (3.8) than working with (3.7) directly. In a general setting the dual problem can be attractive as depending on the particular form of the problem it might be easier to solve than the corresponding primal. For the particular case discussed in this thesis it will be outlined in the next chapter that the dual problem is in many cases essential if a numerical solution should be obtained.

Up to now the considered inequalities in (3.5) and (3.8) were simply element-wise. In the process of this thesis the generalization to conic constraints will be important. Therefore consider that the primal problem (3.5) is augmented with conic constraints of the form $\boldsymbol{x} \in \mathcal{K}_i$ for $i = 1, \dots, M_c$. Then the corresponding Lagrangian is given by $\mathscr{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}, \boldsymbol{\xi}_i) = f(\boldsymbol{x}) - \boldsymbol{\nu}^T \boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{\lambda}^T \boldsymbol{h}(\boldsymbol{x}) - \sum_{i=1}^{M_c} \boldsymbol{\xi}_i^T \boldsymbol{x}$ with $\lambda \geq \mathbf{0}_{M_i}$ and $\boldsymbol{\xi}_i \in \mathcal{K}_i^*$, where $\mathcal{K}_i^*$ denotes the dual cone. Also the concept of complementary slackness is directly extendible to $\boldsymbol{\xi}_i^T \boldsymbol{x} = 0$ such that the KKT conditions (3.9) for a conic problem have to be extended with

$$\boldsymbol{x} \in \mathcal{K}_i, \quad i = 1, \dots, M_c \tag{3.10a}$$

$$\boldsymbol{\xi}_i \in \mathcal{K}_i^*, \quad i = 1, \dots, M_c \tag{3.10b}$$

$$\boldsymbol{\xi}_i^T \boldsymbol{x} = 0, \quad i = 1, \dots, M_c \tag{3.10c}$$

and (3.9e) needs to be modified to

$$\nabla f(\boldsymbol{x}) - \sum_{i=1}^{M_e} \nu_i \nabla g_i(\boldsymbol{x}) - \sum_{i=1}^{M_i} \lambda_i \nabla h_i(\boldsymbol{x}) + \sum_{i=1}^{M_c} \boldsymbol{\xi}_i = \mathbf{0}_N. \tag{3.10d}$$

3.3 Sparsity inducing norms

Recently there has been a lot of interest in sparse models. A model is sparse if many of its parameters are zero. On the one hand a sparse model is more suitable for interpretation than a model which results from the interaction of many influences. Another reason for their popularity is that one measure of complexity for models is given by counting the number of parameters. Then a well-known statement in learning and estimation, referred to as Occam's razor, is that a model with low complexity usually generalizes better than a very complex one. The goal is then finding a model that is as simple as possible but as complex as necessary (paraphrasing a quote by Einstein).

In addition to the interest in sparsity from a pure modeling and complexity point of view, there are plenty of other situations where sparsity is of interest. In large networks, e.g. gene networks, finding a sparse representation allows one to draw conclusions about relevant interactions. Also many engineering problems can be formulated in such a way that sparsity is of great benefit, basically in every situation where one starts with a very broad class of models and wants to eliminate a large fraction. Some applications of this idea can be found in Chapters 6 & 10.

Note however that counting the parameters is only one possible complexity measure. In all regularized model formulations the *effective* number of parameters [Hastie et al., 2009] will be much lower that the number of parameters.

3.3.1 ℓ_1-norm

The most natural measure of sparsity is simply counting nonzero elements. For a vector $x \in \mathbb{R}^N$ the number of its nonzero elements, its cardinality, is often written as $\|x\|_0$. This derives from the fact that the cardinality of a vector can be seen as the limit of the ℓ_p-norms and quasi-norms for $p \to 0$. Strictly speaking this is an abuse of notation because this "0-norm" does not satisfy the conditions for a norm. Nevertheless it is often used in literature. From an engineering point of view the 0-norm is not very useful as optimizing it is a combinatorial problem and therefore excessively expensive for problems with more than a few variables. As it can be relaxed into a convex problem, it is frequently used to motivate problem formulations based on these convex relaxations. The convex envelope, the smallest convex set containing another set, of the 0-norm is the ℓ_1-norm $\|\cdot\|_1$. Besides being the convex envelope of the cardinality, the ℓ_1-norm also works well in practice to obtain sparse solutions. Graphically this is due to its pointy shape as illustrated in Figure 3.3. ℓ_p-norms with $p < 1$ are even more pointed and would result in a better approximation of the cardinality and therefore sparser solutions but have the drawback that they cannot be implemented as a convex problem. Two papers with a huge impact on the application of convex optimization in a variety of fields are [Tibshirani, 1996] that introduced the LASSO to select variables in a regression problem and [Donoho, 2006] which introduced it for compressed sensing and started a whole new line of research.

On the computational side ℓ_1-regularized problems are very attractive as they can be solved as a QP problem when combined with a quadratic loss. Next to general purpose solvers which already are able to deliver good performance on many problems, for special cases like LASSO even more efficient special solvers like LARS [Efron et al., 2004] exist.

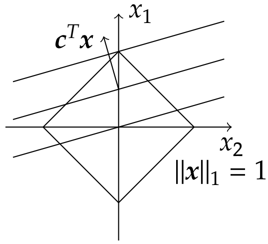

Figure 3.3: Level set of ℓ_1-norm and contour lines of a linear function.

3.3.2 Group ℓ_1-norm

An extension of ℓ_1-norms are so-called group ℓ_1-norms. Here the sparsity is not desired with respect to single variables but with respect to groups of variables, hence the name. The most popular way to construct such group norms is probably through the combination of ℓ_2- and ℓ_1-norms. Assume that a parameter vector $\boldsymbol{\theta} \in \mathbb{R}^N$ is partitioned as $\boldsymbol{\theta} = [\boldsymbol{\theta}_1^T, \dots, \boldsymbol{\theta_M}^T]^T$ with $\boldsymbol{\theta}_m \in \mathbb{R}^{N_m}$ and $N = \sum_m N_m$. Then one can construct a group ℓ_1-ℓ_2-regularization as $\sum_m \|\boldsymbol{\theta}_m\|_2$. Note that the ℓ_2-norms are always nonnegative and their summation is direct without an additional square. As the summation is over nonnegative quantities it corresponds to an ℓ_1-norm. In this fashion one can easily induce sparsity in groups of variables. Instead of the simple form of combining norms to create sparsity, one can also come up with more elaborate schemes like taking differences of variables as applied in Chapter 10.

In contrast to the ℓ_1-case the computational complexity of group ℓ_1-problems is in general higher. In most situations their solution requires solving SOCP problems which are considerably harder than simple QPs. Furthermore the effort to create specialized algorithms is higher. This also holds for their complexity.

3.3.3 Nuclear norm

A more recently considered form of sparsity does not relate to elements of a vector but to the rank of matrix. In many applications the solution of a problem can be written with additional structure and cast into matrix form. Then, next to the number of nonzero elements, another measure of complexity is the number of rank-1 outer products necessary to form the matrix. Penalizing the rank of a matrix has already been important in control theory for a long time [El Ghaoui and Gahinet, 1993]. A previously known relaxation for the rank function is the trace of a matrix [Mesbahi and Papavassilopoulos, 1997; Mesbahi, 1998]. However, this relaxation is only applicable under two assumptions. The first assumption is that the matrix has to be square as otherwise the trace is not defined. Besides the shape of the matrix it additionally has to be at least positive semidefinite (and thus symmetric) as only in that case the sum of all eigenvalues of the matrix is a summation of nonnegative values and therefore an incarnation of the ℓ_1-norm. In case any of the two assumptions is not satisfied, Fazel recently showed, that instead of the trace one can use the nuclear norm of the matrix. This norm is defined as the sum of the singular values of a matrix. As singular values are nonnegative by definition it is acting as a ℓ_1-norm on the singular values and hence imposing a low rank on the solution. Other names for this norm are trace norm or Schatten-p norm. In [Fazel et al., 2001; Fazel, 2002] it was shown that the nuclear norm is the convex envelope of the rank function and how it can be implemented as a SDP problem. Most applications of the nuclear norm so far are not in the control community but rather in the field of compressed sensing, especially matrix completion. The theoretic foundation in this field is developed in [Recht et al., 2010].

The nuclear norm, for a matrix $\boldsymbol{X} \in \mathbb{R}^{N \times M}$ is denoted by $\|\boldsymbol{X}\|_*$ and defined as

$$\|\boldsymbol{X}\|_* = \sum_{k=1}^{\min(N,M)} \sigma_k \tag{3.11}$$

where σ_k are the singular values of $\boldsymbol{X}$. As mentioned previously in the text it can be readily expressed as a problem that can be solved with a general purpose SDP solver. The relation was first given in [Fazel et al., 2001] and is based on the traces of suitably defined matrices as follows

$$\begin{aligned} \|\boldsymbol{X}\|_* = \min_{\boldsymbol{W}=\boldsymbol{W}^T, \boldsymbol{V}=\boldsymbol{V}^T} \quad & \frac{1}{2}\operatorname{tr}(\boldsymbol{W}) + \frac{1}{2}\operatorname{tr}(\boldsymbol{V}) \\ \text{subject to} \quad & \begin{bmatrix} \boldsymbol{W} & \boldsymbol{X} \\ \boldsymbol{X}^T & \boldsymbol{V} \end{bmatrix} \succeq 0 \end{aligned} \tag{3.12}$$

where $\boldsymbol{W} \in \mathbb{R}^{N \times N}$ and $\boldsymbol{V} \in \mathbb{R}^{M \times M}$ are positive semidefinite matrices. In many applications the dimensions of the matrix $\boldsymbol{X}$ are large. Furthermore this SDP based reformulation of the trace norm needs a large number of auxiliary variables to be implemented. Therefore the use of general purpose solvers is often not feasible. However, sparked through the large interest from the compressed sensing community, there are many efforts to come up with customized solvers for nuclear norm based problems. Some of the basic ideas of these algorithms are briefly outlined in the next section.

3.4 Algorithms

The adoption of advanced numerical optimization relies on the one hand on efficient algorithms like those outlined in this section. On the other hand a maybe even bigger impact for the adaption are tools that significantly simplify the formulation of (convex) optimization problem. These tools allow for a "natural" formulation of optimization problems and then automatically transform the problem into standard forms suitable for general purpose solvers. At least for academic purposes the packages CVX [Grant and Boyd, 2011] and YALMIP [Löfberg, 2004] have proven to be extremely convenient. These tools allow rapid development and successive testing of problem formulations and move the complexity from interfacing with software to the high-level modeling of the problem at hand.

3.4.1 Interior point methods

The presence of interior point solvers boosted the success of convex programming. It is also the most commonly implemented algorithm for general purpose solvers. The basic idea is to implement the constraints by an additional penalty in the objective function called (self-concordant) barrier function. The barrier function is taken from a family of functions which in the extreme case take on a value of zero inside the feasible region and of infinity when outside. Other elements of the family are smooth approximations of the extreme case. In the process of the optimization the barrier is adjusted such that it mimics the extreme case better and better. It has been shown that under certain conditions, for a sequence of tuning parameters, the solution at each iteration will converge to the solution of the optimization problem. Interior point algorithms can be implemented for the primal problem (3.5) but one can gain numerical stability by formulating an algorithm that solves the primal and the dual problem in a joint way. A historical overview as well as the key ingredients are nicely presented in a recent paper by Wright [2005].

Some popular general purpose solvers that rely on interior point methods are SDPT3 [Toh et al., 1999], CVXOPT [Dahl and Vandenberghe, 2011], MOSEK [Mosek, 2011] and CPLEX [IBM, 2010].

3.4.2 First order algorithms

As suggested by the name, first order algorithms only use gradient information. They recently regained popularity due to several advantages

- they often can be accelerated and then achieve convergence with $\mathcal{O}(1/k^2)$ [Nesterov, 2005; Beck and Teboulle, 2009],
- they have a very simple structure and only require basic algebra, which allows easy as well as efficient implementation on highly parallel structures like GPUs,
- they can be extended to some important nonsmooth optimization problems,
- the computation of regularization paths is relatively inexpensive as gradient methods can usually be warm-started efficiently and
- due to their simple form it is often straightforward to exploit structure of the optimization problem at hand.

In the scope of this thesis the relevant algorithms are called (accelerated) gradient projection which support nonsmooth optimization and can converge as $\|x_k - x^*\|_2 < ck^{-2}$ where $c > 0$ is a problem and initialization dependent constant, k the iteration count and x^* an optimal point. First it will be discussed under which conditions and how nonsmooth optimization problems can be tackled. After that it will be briefly outlined how the method can be accelerated to achieve the optimal rate of convergence.

Consider the optimization problem

$$\min_{x \in \mathbb{R}^N} \quad f(x) \quad \text{subject to} \quad x \in \mathcal{C} \tag{3.13}$$

where $f(x)$ is a smooth convex function and $\mathcal{C}$ is a convex set. The main requirement for the success of a gradient projection algorithm is that a projection operator $P_{\mathcal{C}} : \mathbb{R}^N \to \mathbb{R}^N$ onto $\mathcal{C}$ exists and the projection can be computed efficiently. Instead of assuming a convex constraint set one can generalize the algorithm to composite objective functions of the form $f(x) + g(x)$ where $f(x)$ is again a smooth convex function and $g(x)$ can be a nonsmooth function. In this setting a proximal operator takes the place of the projection in the simpler problem (3.13).

A problem of the form given by (3.13) can be solved with a straightforward extension of the steepest descent algorithm. For this aim define a gradient map $G_c(x) = \frac{1}{c}(x - P_{\mathcal{C}}(x - c\nabla f(x))$. Based on this a generic algorithm is as follows.

Algorithm 3.1 (Gradient projection method).

1. Set $k = 1$ and initialize $\boldsymbol{x}_0$.
2. For optimal convergence choose c_k
   a) as $1/L$ where L is a Lipschitz constant of ∇f or
   b) such that the inequality
   $$f(\boldsymbol{x}_{k-1} - c_k G_{c_k}(\boldsymbol{x}_{k-1})) \leq f(\boldsymbol{x}_{k-1}) - c_k \nabla f(\boldsymbol{x}_{k-1}) + \frac{c_k}{2} \|G_{c_k}(\boldsymbol{x}_{k-1})\|_2^2$$
   is satisfied, for example with a backtracking line search.
3. Then set $\boldsymbol{x}_k = \boldsymbol{x}_{k-1} - c_k G_{c_k}(\boldsymbol{x}_{k-1})$.
4. Stop if $\|G_{c_k}(\boldsymbol{x}_{k-1})\|_2$ is small enough.
5. Set $k := k + 1$ and repeat with 2.

This algorithm is guaranteed to converge, but its convergence is only $\mathcal{O}(1/k)$. An algorithm with the optimal convergence rate $\mathcal{O}(1/k^2)$ can be obtained by storing two sequences of gradients. A possible modification of the algorithm above, that achieves the optimal convergence rate, is given below.

Algorithm 3.2 (Accelerated gradient projection method).

1. Set $k = 1$ and initialize $\boldsymbol{x}_0$ and set $\boldsymbol{y}_0 = \boldsymbol{x}_0$.
2. For optimal convergence choose c_k
   a) as $1/L$ where L is a Lipschitz constant of ∇f or
   b) such that the inequality
   $$f(\boldsymbol{y}_{k-1} - c_k G_{c_k}(\boldsymbol{y}_{k-1})) \leq f(\boldsymbol{y}_{k-1}) - c_k \nabla f(\boldsymbol{y}_{k-1}) + \frac{c_k}{2} \|G_{c_k}(\boldsymbol{y}_{k-1})\|_2^2$$
   is satisfied, for example with a backtracking line search.
3. Then set $\boldsymbol{x}_k = \boldsymbol{y}_{k-1} - c_k G_{c_k}(\boldsymbol{y}_{k-1})$ and $\boldsymbol{y}_k = \boldsymbol{x}_k + \frac{k-1}{k+2}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$.
4. Stop if $\|G_{c_k}(\boldsymbol{y}_{k-1})\|_2$ is small enough.
5. Set $k := k + 1$ and repeat with 2.

With TFOCS [Becker et al., 2012] there is a MATLAB toolbox that allows the rapid development of first order algorithms by simple combination of basic building blocks. However in many cases cycle performance of first order algorithms is critical due to the typically large number of iterations needed for convergence. In these cases it might still be necessary to implement a custom solver exploiting and tailored to the specifics of the problem under consideration.

3.4.3 Related techniques

There are many other methods used for convex as well as general nonlinear optimization. In this subsection two of them are briefly introduced. The *method of multipliers* also known as *augmented Lagrangian method* allows the use of unconstrained solvers for constrained optimization problems in an iterative procedure. For problems with a large number of constraints, active set strategies can be beneficial by solving a series of subproblems with few constraints and terminating when all relevant ones are satisfied.

Method of multipliers

Most optimization routines are first developed for unconstrained optimization. One example for such developments are gradient techniques that can be extended by projecting the gradients as described in the last subsection. Another example that has already been discussed are interior point methods which are an extension of the Newton method. In contrast to these examples the method of multipliers is a straightforward and generic approach to extend any unconstrained optimization routine to handle constrained problems. Here, for clarity as well as brevity of presentation, it will only be considered for equality constraints. Consider the following optimization problem

$$\begin{aligned} &\min_{\boldsymbol{x}} \quad f(\boldsymbol{x}) \\ &\text{subject to} \quad \boldsymbol{h}(\boldsymbol{x}) = \mathbf{0}_M \end{aligned} \tag{3.14}$$

with $\boldsymbol{x} \in \mathbb{R}^N$, $f : \mathbb{R}^N \to \mathbb{R}$ and $\boldsymbol{h} : \mathbb{R}^N \to \mathbb{R}^M$. Then the idea used by the method of multipliers is to form an augmented Lagrangian

$$g_k(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T \boldsymbol{h}(\boldsymbol{x}) + c_k \|\boldsymbol{h}(\boldsymbol{x})\|_2^2 \tag{3.15}$$

for a series of $c_k \to \infty$ as the iteration k goes to infinity. One can show, under some conditions, that this procedure will converge to a local optimum of f that satisfies the equality constraints [Bertsekas, 1996, 1999; Nocedal and Wright, 2006].

In its most simple form, one initializes $\boldsymbol{\lambda}$ to an arbitrary $\boldsymbol{\lambda}_0$ and c_k to $c_0 > 0$. Then in every iteration first the problem $\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}} g_k(\boldsymbol{x}, \boldsymbol{\lambda}_k)$ is solved, and subsequently the multipliers are updated as $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + c_k \boldsymbol{h}(\boldsymbol{x}_k)$. Finally one chooses a c_{k+1} such that $c_{k+1} > c_k$ and iterates until a stopping criterion is satisfied. After convergence the solution will satisfy $\|\boldsymbol{h}(\boldsymbol{x})\|_2 \simeq 0$.

The basic outline of such an algorithm is given below. A more practical algorithm is for example described by Nocedal and Wright [2006, Alg. 17.4].

Algorithm 3.3 (Method of multipliers).

1. Initialize $c_1 > 0$, $\boldsymbol{x}_0 = \mathbf{0}_N$, $\boldsymbol{\lambda}_0 = \mathbf{0}_M$ and set $k = 1$.
2. Set $\boldsymbol{\lambda}_k := \boldsymbol{\lambda}_{k-1} + c_k \boldsymbol{h}(\boldsymbol{x}_{k-1})$.
3. Solve $\boldsymbol{x}_k = \min_{\boldsymbol{x}} \; g_k(\boldsymbol{x}, \boldsymbol{\lambda}_k)$.
4. Terminate if $\|\boldsymbol{h}(\boldsymbol{x}_k)\|_2$ is small enough.
5. Choose c_{k+1} such that $c_{k+1} > c_k$, increment k and go to (2).

Active set techniques

As the last method, also active set techniques are intended for constrained optimization problems. In contrast to the augmented Lagrangian method, here the goal is not the solution itself, but merely a more efficient computation. The idea of active set techniques is straightforward. Consider the optimization problem

$$\begin{aligned} &\min_{\boldsymbol{x}} \quad f(\boldsymbol{x}) \\ &\text{subject to} \quad g_k(\boldsymbol{x}) \leq 0, \quad k = 1, \dots, M \end{aligned} \tag{3.16}$$

with $\boldsymbol{x} \in \mathbb{R}^N$, $f : \mathbb{R}^N \to \mathbb{R}$ and $g_k : \mathbb{R}^N \to \mathbb{R}$ for $k = 1, \dots, M$. The use case for active set methods are problems with many constraints, i.e. $M \gg 1$. In that case it can be very time consuming or even impossible to solve (3.16) using a general purpose solver. One way to efficiently handle problems with many constraints is to first start by solving a simplified problem with a substantially reduced number of constraints. The obtained solution can then be substituted into the original set of constraints. Then the constraint that is violated the most is included in the set of active constraints and a new solution is found for this updated problem. It is often the case that only a small fraction of the constraints is active at the solution. In this situation the described iterative procedure can substantially improve the computational speed, especially if combined with a solver that can be warm started using the solution obtained in the previous iteration. The outline of a basic active set algorithm is given below.

Algorithm 3.4 (Active set algorithm).

1. Initialize $\mathcal{I}_1 = \{1\}$, $k = 1$.
2. Solve $\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x})$ subject to $g_l(\boldsymbol{x}) \leq 0$, $l \in \mathcal{I}_k$.
3. Select $l^* = \arg\max_{1 \leq l \leq M} g_l(\boldsymbol{x}_k)$.
4. If $g_{l^*}(\boldsymbol{x}_k) = 0$, stop.

5. Form $\mathcal{I}_{k+1} = \mathcal{I}_k \cap \{l^*\}$, set $k := k + 1$ and go to 2.

Remark 3.1. This method is not to be confused with active set techniques for quadratic or linear programming. These methods additionally exploit that at the optimal solution the active constraints have to hold with equality. Therefore at each iteration the current guess for the active inequality constraints is added to the problem as equality constraints. Then a problem with only equality constraints is solved and the guess for the set of active constraints is updated. More details can for example be found in [Nocedal and Wright, 2006].

3.5 Convex relaxations

Many optimization problems of practical interest are not convex. However, as motivated in this chapter, convex optimization problems have several theoretical as well as practical advantages, for instance that every local optimum is also globally optimal and that for several classes of convex problems many efficient solvers exist. Therefore there is strong interest to recast nonconvex problems as convex ones. The solutions of convex relaxations naturally do not necessarily coincide with those of the originally nonconvex problems. However, for some classes of problems one can compute a worst case bound of the distance that certain convex approximations can have to the globally optimal solution [Goemans and Williamson, 1995; Karger et al., 1998]. In other cases one can derive conditions under which the solution of the convex approximation condition will coincide with the original problem [Candès et al., 2006b].

3.5.1 Norms

One of the most widely studied classes of problems that can be relaxed using norms, are problems involving the cardinality of a vector, i.e. its nonzero elements. Also refer to Section 3.3 on sparsity. For example in feature selection one tries to estimate the least number of variables that explains the data, or in compressed sensing one tries to reconstruct a signal that is known to be sparse from as few measurements as possible. These problems are commonly stated as

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathbb{R}^N} \quad & \operatorname{card}(\boldsymbol{x}) \\ \text{subject to} \quad & \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\| \le \varepsilon \end{aligned} \tag{3.17}$$

where the cardinality is minimized subject to a bound on the misfit or

$$\begin{aligned} \min_{x\in\mathbb{R}^N} \quad & \|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}\| \\ \text{subject to} \quad & \operatorname{card}(\boldsymbol{x}) \leq C \end{aligned} \tag{3.18}$$

where the misfit is minimized with a bound on the cardinality. A powerful heuristic, that seems to work well in practice, is to replace the cardinality function by its convex envelope, which can be shown to be the ℓ_1-norm. In the area of compressed sensing it has been shown [Donoho, 2006; Candès et al., 2006b] that under certain assumptions on $\boldsymbol{A}$, known as the restricted isometry property (RIP), the relaxation will give rise to the exact solution with very high probability.

Another popular function that can often be used to capture the nonconvexity in a problem is the rank of a matrix. Similar to the cardinality problem the convex hull is again given by a norm, in this case the nuclear or trace norm. Also for the trace norm the compressed sensing community formulated a RIP condition under which the trace norm is a perfect surrogate for the rank function with very high probability [Recht et al., 2010].

An extension of the ℓ_1-heuristic used later in this thesis is its application to groups of variables and linear combinations. For example minimizing the cardinality of $\boldsymbol{z}$ where $z_i = \|\boldsymbol{P}_i\boldsymbol{x}\|$ for some linear combinations $\boldsymbol{P}_i$ of $\boldsymbol{x}$ are sometimes interesting in practice.

3.5.2 Overparametrization

Besides using norms to obtain relaxation, another capable approach is the introduction of new variables. This idea has first been applied by Shor [1987] for the solution of nonconvex quadratic programs. He reformulated $\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}$ where $\boldsymbol{Q}$ is indefinite as $\operatorname{tr}(\boldsymbol{Q}\boldsymbol{X})$ where $\boldsymbol{X}$ is obtained as the relaxation of the rank-1 matrix $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^T$ to a positive semidefinite matrix $\boldsymbol{X} \succeq 0$. In case the rank constraint on $\boldsymbol{X}$ is obeyed, then this reformulation is exact with an objective that is convex, however a constraint that is nonconvex. Then, by relaxing the constraint, a convex approximation is obtained.

In fact the same idea can be applied to a much larger range of problems. Whenever one faces bilinear forms $\boldsymbol{x}^T\boldsymbol{R}\boldsymbol{y} = \operatorname{tr}(\boldsymbol{R}\boldsymbol{y}\boldsymbol{x}^T)$ the rank-1 product can be replaced by a matrix variable $\boldsymbol{Y} = \boldsymbol{y}\boldsymbol{x}^T$ and the corresponding rank constraint. In the same manner as before a convex approximation of the bilinear form can be obtained by dropping the rank-1 constraint on $\boldsymbol{Y}$.

More advanced applications and theory on this topic, also referred to as SDP liftings, can be found for example in [Luo et al., 2010; Nesterov, 1998]. A generalization denoted as the *method of moments* is described in [Lasserre, 2001; Henrion and Lasserre, 2004]. The basic scheme described here will be used in most of the following chapters to obtain convex approximations for initially nonconvex problems. In some cases heuristics based on norms, as introduced in the last subsection, are considered to achieve tighter relaxations.

4 Least Squares Support Vector Machines

In the preceding chapter on system identification, one form of nonlinear system identification was reduced to the estimation of a function of suitably selected variables, c.f. (2.7). For this problem many methods have been proposed, like wavelets [Zhang and Benveniste, 1992], artificial neural networks [Narendra and Parthasarathy, 1990; Suykens et al., 1995; Haykin, 1998] and more recently kernel based techniques. This chapter will briefly discuss Least Squares Support Vector Machines (LS-SVMs) [Suykens, Van Gestel, et al., 2002; Suykens et al., 2010] which fall into the large class of kernel based learning methods. Other members of this class are for example Splines [Wahba, 1990], Gaussian processes [Rasmussen and Williams, 2006], Regularization Networks [Poggio and Girosi, 1990], Support Vector Machines (SVMs) [Vapnik, 1998] and Kriging [Krige, 1951]. Machine learning treats problems in many diverse fields like, but not limited to, supervised and unsupervised learning, regression and classification. Although LS-SVMs can be adapted to all of the mentioned settings, in the context of this thesis only the (supervised) regression case is of interest. Within the class of kernel based machine learning techniques there are two popular ways of presentation. On the one hand one can formulate optimization problems in a special class of function spaces, which allows a finite dimensional representation of the solution in terms of the estimation data. The key elements here are Reproducing Kernel Hilbert Spaces [Aronszajn, 1950] and representer theorems [Kimeldorf and Wahba, 1971]. On the other hand one can explicitly parametrize the model in a so-called primal representation and then use Lagrangian duality and convex optimization to obtain model representations in terms of the estimation data. This approach is based on Mercer's theorem [Mercer, 1909], which shows the existence of basis function expansions of positive definite functions. Many results can be obtained using either of these two approaches and the choice can be adapted to one's personal taste and background as well as the problem at hand. Classically LS-SVMs are formulated with primal and dual formulations based on convex optimization, which will also be used most of the times throughout this thesis. This chapter will introduce both formulations but concentrates on the primal-dual methodology. The objective in LS-SVMs is to minimize the squared residuals on a training set with a Tikhonov type of regularization on the estimated function. As such its solution is related to that of other methods such as Kriging and Regularization Networks which have the same objective. However from its formulation and interpretation it is more closely related to support vector machines. In both cases one starts from a linear regression model in a high dimensional space

$$f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}) + b. \tag{4.1}$$

The function $\boldsymbol{\varphi} : \mathbb{R}^D \to \mathbb{R}^{n_h}$ is denoted as "feature map" and defines the usually high dimensional "feature space". The parameters of the model are $\boldsymbol{w} \in \mathbb{R}^{n_h}$ and b. The important concept in support vector models is that the feature map does not have to be defined explicitly. It is sufficient to specify a positive definite and scalar valued kernel function. The relation between this kernel function and its feature map is then given by the afore mentioned Mercer's theorem. Given a set of training data $\mathcal{V}_1 = \{(\boldsymbol{x}_t, y_t)\}_{t=1}^N$ one can fit the model given by (4.1) to data using $y_t = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t$, where the e_t denotes the modeling residuals. In SVMs the residuals are subject to the ε-insensitive loss function. This loss function is convex and can be described by a linear objective function and linear inequality constraints. In combination with the Tikhonov regularization term the resulting optimization problem is a QP, which can be solved with general purpose optimization software. The complexity of kernel based problems usually scales with the number of data N. Therefore special solvers have been developed to accelerate the solution by exploiting the special structure of the problem, making the solution feasible even for larger values of N. In contrast to SVMs, LS-SVMs do not require specialized software for their solution as they only require the solution of linear systems of equations. This is due to a change of the loss function from ε-insensitive to the well-known least squares loss. Using this loss, the model can be imposed using equality constraints and the QP simplifies to a regularized least squares problem. In the following section the primal optimization problems for LS-SVMs and SVMs will be formally stated. Using Lagrange duality and the kernel trick their dual representations are derived. The subsequent section briefly discusses the formulation based on reproducing kernel Hilbert spaces. While SVMs have a loss function that induces sparsity, the solutions of LS-SVMs are usually dense. Therefore one section is dedicated to approaches to cope with large datasets by suitable approximations. The estimation of the parameters w and b characterizing the support vector models is a convex problem. However the estimation of a model with good generalization performance relies on the careful selection of kernel function and regularization constant. This problem is nonconvex and mostly tackled with some kind of validation technique which is briefly discussed in the last section of this chapter.

Note that the two proofs given in this chapter are blueprints for those in the remainder of this thesis. Most of the following proofs will share the same key ingredients like Lagrangian duality, the kernel trick and the general structure.

4.1 Primal and dual model representations

As suggested in the previous section, support vector techniques have two forms, the primal and the dual. While the primal looks like a parametric estimation problem and allows both easy interpretation and modification, in most cases it is unsuitable for solution. Therefore the dual description is derived which converts the parametric problem, that is often defined only implicitly, into a nonparametric formulation. The dual is explicitly defined and allows efficient solution but is itself not suitable for interpretation. An important result is that not only the estimation problem can be solved in the dual, but also the model itself can be expressed in terms of the dual solution. The first fact is actually true for a large class of optimization problems. The following two subsections will briefly derive two flavors of support vector machines, namely LS-SVMs using the least squares loss and ε-SVMs based on the ε-insensitive loss.

4.1.1 Least squares loss

Least squares support vector machines were first introduced by Suykens and Vandewalle [1999] for classification. Comprehensive information in the context of classification, regression and many others can be found in [Suykens, Van Gestel, et al., 2002; Suykens et al., 2010].

Table 4.1: Examples of positive definite kernel functions $K(\boldsymbol{x}, \boldsymbol{y})$ for $\boldsymbol{x}, \boldsymbol{y} \in R^D$.

KERNEL	DEFINITION
Linear	$\boldsymbol{x}^T \boldsymbol{y}$
Polynomial	$(\boldsymbol{x}^T \boldsymbol{y} + c)^d$, $c \geq 0$, $d \in \mathbb{N}$
Gaussian RBF	$\exp(-\|\boldsymbol{x} - \boldsymbol{y}\|_2^2 / \sigma^2)$, $\sigma > 0$

Based on the model given by (4.1), the least squares loss, Tikhonov regularization and training data $\mathcal{V}_1 = \{(\boldsymbol{x}_t, y_t)\}_{t=1}^N$ with $\boldsymbol{x}_t \in \mathbb{R}^D$ and $y_t \in \mathbb{R}$, the primal LS-SVM problem for regression can be stated as

$$\begin{aligned} \min_{\boldsymbol{w}, b, e_t} \quad & \frac{1}{2} \boldsymbol{w}^T \boldsymbol{w} + \frac{1}{2} \gamma \sum_{t=1}^{N} e_t^2 \\ \text{subject to} \quad & y_t = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t, \quad t = 1, \dots, N, \end{aligned} \tag{4.2}$$

where $\gamma \in \mathbb{R}$ is a positive regularization constant. As noted before, the feature map $\boldsymbol{\varphi}$ is not necessarily defined explicitly, but usually implicitly by a positive definite kernel function. Examples for popular kernel functions are given in Table 4.1. For some kernel functions, like the Gaussian RBF kernel, the corresponding feature map is infinite dimensional. The relation between kernel function and feature map is given by Mercer's theorem.

Definition 4.1 (Positive definite function). Let $S \subset \mathbb{R}^D$ and $K : S \times S \to \mathbb{R}$ be a symmetric function, i.e. $K(\boldsymbol{x}, \boldsymbol{y}) = K(\boldsymbol{y}, \boldsymbol{x})$ for all $\boldsymbol{x}, \boldsymbol{y} \in S$, that satisfies

$$\sum_{i,j=1}^{N} c_i c_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$$

for any $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N \in S$ and any $c_1, \dots, c_N \in \mathbb{R}$. Then K is said to be positive definite.

Theorem 4.1 (Mercer's theorem [Mercer, 1909]). *Let K be a positive definite function, further on denoted as kernel, then it can be expressed as*

$$K(\boldsymbol{x}, \boldsymbol{y}) = \sum_{n=1}^{\infty} \lambda_n \tilde{\varphi}_n(\boldsymbol{x}) \tilde{\varphi}_n(\boldsymbol{y}), \tag{4.3}$$

where $\lambda_n \geq 0$ and $\{\tilde{\varphi}_n\}_{n=1}^{\infty}$. The $\tilde{\varphi}_n : \mathbb{R}^D \to \mathbb{R}$ form an orthonormal basis. Here, λ_n is an eigenvalue and $\tilde{\varphi}_n$ the corresponding eigenfunction of K,

$$\int_S K(\boldsymbol{x}, \boldsymbol{y}) \tilde{\varphi}_n(\boldsymbol{x}) d\boldsymbol{x} = \lambda_n \tilde{\varphi}_n(\boldsymbol{y}). \tag{4.4}$$

Based on this relation and Lagrangian duality the implicitly defined and maybe infinite dimensional problem (4.2) can be converted to an explicitly defined problem in finite dimensions.

Lemma 4.2. *Given a positive definite kernel function K, that corresponds to the feature map $\boldsymbol{\varphi}$ via Mercer's theorem, the solution of the primal problem* (4.2) *can be obtained from the linear system*

$$\begin{bmatrix} \boldsymbol{\Omega} + \gamma^{-1}\boldsymbol{I}_N & \mathbf{1}_N \\ \mathbf{1}_N^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ 0 \end{bmatrix}, \tag{4.5}$$

where $\boldsymbol{y} = [y_1, \dots, y_N]^T$. The kernel matrix or Gram matrix $\boldsymbol{\Omega}$ is defined element-wise as $\Omega_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\varphi}(\boldsymbol{x}_i)^T \boldsymbol{\varphi}(\boldsymbol{x}_j)$ for $i, j = 1, \dots, N$. The dual variables $\boldsymbol{\alpha} \in \mathbb{R}^N$ are the Lagrange multipliers for the equality constraints in (4.2). *The relation between primal and dual variables is given by $\boldsymbol{w} = \sum_{t=1}^N \alpha_t \boldsymbol{\varphi}(\boldsymbol{x}_t)$.*

Proof. Introducing Lagrange multipliers α_t for the equality constraints of (4.2), its Lagrangian can be stated as

$$\mathscr{L}(\boldsymbol{w}, b, e_t, \alpha_t) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{t=1}^N e_t^2 - \sum_{t=1}^N \alpha_t(\boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t - y_t).$$

Computing the Karush-Kuhn-Tucker conditions for optimality one obtains the following relations

$$\begin{aligned} \mathbf{0}_{n_h} &= \frac{\partial \mathscr{L}}{\partial \boldsymbol{w}} = \boldsymbol{w} - \sum_{t=1}^N \alpha_t \boldsymbol{\varphi}(\boldsymbol{x}_t), \\ 0 &= \frac{\partial \mathscr{L}}{\partial b} = -\sum_{t=1}^N \alpha_t, \\ 0 &= \frac{\partial \mathscr{L}}{\partial e_t} = \gamma e_t - \alpha_t \text{ and} \\ 0 &= \frac{\partial \mathscr{L}}{\partial \alpha_t} = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t - y_t. \end{aligned}$$

The expansion of $\boldsymbol{w}$ in terms of the dual variables is directly obtained from the first KKT condition. Substituting this expansion into the last KKT condition, one can apply Mercer's theorem, in particular (4.3), backwards and replace $\boldsymbol{\varphi}(\boldsymbol{x}_i)^T\boldsymbol{\varphi}(\boldsymbol{x}_j) = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$. This is also known as the kernel trick. Note that for an explicitly known feature map Mercer's theorem is not necessary and $\boldsymbol{\varphi}$ can contain any linear independent functions. However, once $\boldsymbol{\varphi}$ is not explicitly defined and only implicitly induced by the kernel K, Mercer's theorem provides the rationale that some, albeit unknown, feature map exists. Furthermore substituting the relation $e_t = \gamma^{-1}\alpha_t$, obtained from the KKT condition for e_t, into the last KKT condition and writing it in matrix form gives the first block row of the dual linear system (4.5). The last row is simply the KKT condition for b written in vector form. □

Based on the expansion of $\boldsymbol{w}$ in terms of the dual variables, the predictor at a point $\boldsymbol{z}$ can be stated in the dual form as

$$\hat{y}(\boldsymbol{z}) = \sum_{t=1}^{N} \alpha_t K(\boldsymbol{x}_t, \boldsymbol{z}) + b. \tag{4.6}$$

Below a complete algorithm is given to estimate a LS-SVM model.

Algorithm 4.1 (Estimation of plain LS-SVM model)**.**

1. Select a regularization parameter γ and a kernel function K (and its parameters).
2. Compute the kernel matrix $\boldsymbol{\Omega}$ based on the training set $\mathcal{V}_1$.
3. Solve the dual linear system (4.5).

4.1.2 ε-insensitive loss

Support vector machines (SVMs) were initially introduced in the context of linear classification [Vapnik and Lerner, 1963]. They were later extended to, among others, the regression problem and most importantly nonlinear problems making use of the kernel trick. The ε-insensitive loss function was used for the extension to the regression problem. Extensive background on SVMs, their theory and extensions can be found in [Vapnik, 1998; Cristianini and Shawe-Taylor, 2000; Schölkopf and Smola, 2002; Steinwart and Christmann, 2008]. A short introduction is available in [Smola and Schölkopf, 2004].

The ε-insensitive loss is defined as $l_\varepsilon(x) = 0$ for $x \leq \varepsilon$ and $l_\varepsilon(x) = |x| - \varepsilon$ otherwise. The fact that this loss function has i) ℓ_1-characteristic and ii) a dead zone makes the solution of SVMs sparse in terms of the dual variables α_t. This is in contrast to LS-SVMs which usually give rise to non-sparse α_t. Based on this loss function, Tikhonov regularization for $\boldsymbol{w}$ and the training data $\mathcal{V}_1$

introduced previously, the primal ε-SVM can be formulated as follows

$$\begin{aligned} \min_{w,b,\xi_\pm} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{t=1}^{N}(\xi_t^+ + \xi_t^-) \\ \text{subject to} \quad & y_t - \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) - b \leq \varepsilon + \xi_t^-, \quad t = 1,\dots,N, \\ & \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b - y_t \leq \varepsilon + \xi_t^+, \quad t = 1,\dots,N, \\ & \xi_t^\pm \geq 0, \quad t = 1,\dots,N, \end{aligned} \tag{4.7}$$

where $C \in \mathbb{R}$ is a positive regularization constant and $\varepsilon > 0$ the width of the dead zone characterizing the loss l_ε. Comparing this formulation to the LS-SVM problem (4.2) one can see that modeling the ε-insensitive requires inequality constraints instead of equality constraints, which are easier to solve. Due to this, the dual problem in case of SVMs is the QP given in the following Lemma.

Lemma 4.3. *Given a kernel function K, the dual problem to* (4.7) *is given by*

$$\begin{aligned} \max_{\alpha_t} \quad & \sum_{t=1}^{N}\alpha_t y_t - \varepsilon\sum_{t=1}^{N}|\alpha_t| - \frac{1}{2}\boldsymbol{\alpha}^T\boldsymbol{\Omega}\boldsymbol{\alpha} \\ \text{subject to} \quad & -C \leq \alpha_t \leq C, \quad t = 1,\dots,N, \\ & \mathbf{1}_N^T\boldsymbol{\alpha} = 0, \end{aligned} \tag{4.8}$$

where $\boldsymbol{\alpha} = [\alpha_1,\dots,\alpha_N]^T$ is the difference of the Lagrange multipliers corresponding to the inequality constraints in (4.7). *The kernel matrix $\boldsymbol{\Omega}$ is defined element-wise as $\Omega_{ij} = K(\boldsymbol{x}_i,\boldsymbol{x}_j) = \boldsymbol{\varphi}(\boldsymbol{x}_i)^T\boldsymbol{\varphi}(\boldsymbol{x}_j)$ for $i,j = 1,\dots,N$. The relation between primal and dual variables is given by $\boldsymbol{w} = \sum_{t=1}^{N}\alpha_t\boldsymbol{\varphi}(\boldsymbol{x}_t)$.*

Proof. The Lagrangian for (4.7) can be written as

$$\begin{aligned} \mathscr{L}(\boldsymbol{w}, b, \xi_t^\pm, \alpha_t^\pm, \zeta_t^\pm) = {} & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{t=1}^{N}(\xi_t^+ + \xi_t^-) \\ & + \sum_{t=1}^{N}\alpha_t^+(y_t - \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) - b - \varepsilon - \xi_t^+) - \sum_{t=1}^{N}\xi_t^+\zeta_t^+ \\ & + \sum_{t=1}^{N}\alpha_t^-(\boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b - y_t - \varepsilon - \xi_t^-) - \sum_{t=1}^{N}\xi_t^-\zeta_t^- \end{aligned}$$

where $\alpha_t^\pm \geq 0$ are the Lagrange multipliers for the inequality constraints and $\zeta_t^\pm \geq 0$ are the multipliers for the positivity constraints, respectively. Taking

Table 4.2: Summary of important relations for LS-SVMs and ε-SVMs.

	LS-SVM		SVM
loss function	LS		ε-insensitive
primal problem	(4.2)		(4.7)
primal model		— (4.1) —	
dual problem	(4.5)		(4.8)
dual model		— (4.6) —	
optimization problem	linear system		QP

the conditions for optimality one obtains

$$
\begin{aligned}
\mathbf{0}_{n_h} &= \frac{\partial \mathscr{L}}{\partial \boldsymbol{w}} = \boldsymbol{w} - \sum_{t=1}^{N} (\alpha_t^+ - \alpha_t^-) \boldsymbol{\varphi}(\boldsymbol{x}_t), \\
0 &= \frac{\partial \mathscr{L}}{\partial b} = -\sum_{t=1}^{N} (\alpha_t^+ - \alpha_t^-), \\
0 &= \frac{\partial \mathscr{L}}{\partial \xi_t^\pm} = C - \alpha_t^\pm - \zeta_t^\pm, \quad t = 1, \dots, N.
\end{aligned}
$$

Introducing new variables $\alpha_t = \alpha_t^+ - \alpha_t^-$, the expansion for $\boldsymbol{w}$ is identical to the LS case and so is the constraint $\mathbf{1}_N^T \boldsymbol{\alpha} = 0$. From the optimality conditions for $\xi_t^\pm$ one obtains $\alpha_t^\pm = C - \zeta_t^\pm$. Combining this with the positivity constraints on $\alpha_t^\pm$ and $\zeta_t^\pm$ one obtains the relation $0 \le \alpha_t^\pm \le C$. Performing the change of variables introduced earlier this yields $-C \le \alpha_t \le C$. Substitution of all KKT conditions into the Lagrangian yields the dual objective function. Note that $|\alpha_t| = \alpha_t^+ + \alpha_t^-$. □

The one-step-ahead predictor in terms of the dual variables has the same form as in the least squares case (4.6). A difference is that for those t for which the inequality constraints of (4.7) are strictly satisfied, i.e. the data $(\boldsymbol{x}_t, y_t)$ is within the insensitivity zone, the corresponding Lagrange multipliers are zero. Therefore the SVM formulation has inherent sparsity. Table 4.2 briefly summarizes the key elements and formulations of SVMs and LS-SVMs.

A notable variation of ε-SVMs are ν-SVMs [Schölkopf et al., 2000]. Whereas in LS-SVMs one has to select only one parameter besides the kernel parameters, γ, for the regularization, in ε-SVMs the regularization parameter C must be selected along with the parameter ε of the cost function. In ν-SVMs the parameter ε is replaced by a variable ρ, the regularization constant C is fixed to N^{-1} and the cost function augmented by the term $-\nu\rho$. It can be shown that this automatically selects a value for ρ (or ε respectively). Therefore only ν has to be chosen by the user, reducing the number of parameters to be selected. Furthermore by selecting ν one can directly influence the sparsity of the model, i.e. the fraction of nonzero support vectors.

4.2 Estimation in reproducing kernel Hilbert spaces

Instead of adopting the optimization setting presented in the previous section, one can also rely on functional analysis to derive support vector models. Instead of optimization and Mercer's theorem, in this setting reproducing kernel Hilbert spaces (RKHSs) [Aronszajn, 1950] and representer theorems [Kimeldorf and Wahba, 1971; Schölkopf et al., 2001] are the crucial building blocks.

Given a positive definite kernel K and some domain $\mathcal{X}$ one can define functions as

$$f(z) = \sum_{i=1}^{N_f} \alpha_i K(\boldsymbol{x}_i, z) \quad \text{and} \quad g(z) = \sum_{j=1}^{N_g} \beta_j K(\boldsymbol{x}_j, z), \tag{4.9}$$

where $\alpha_i, \beta_j \in \mathbb{R}$, $N_f, N_g \in \mathbb{N}$ and $\boldsymbol{x}_i, \boldsymbol{x}_j$ are arbitrary points in $\mathcal{X}$. An inner product can be defined as

$$\langle f, g \rangle = \sum_{i=1}^{N_f} \sum_{j=1}^{N_g} \alpha_i \beta_j K(\boldsymbol{x}_i, \boldsymbol{x}_j). \tag{4.10}$$

Here it was used that a positive definite kernel is a representing kernel, i.e. $f(x) = \langle K(x, \cdot), f(\cdot) \rangle$. Completing the space given by (4.9) one obtains the RKHS $\mathcal{H}$ that corresponds to the kernel K. The inner product for $\mathcal{H}$ is given by (4.10) and the associated norm can be defined based on the inner product and is given by $\|f\| = \sqrt{\langle f, f \rangle}$.

Now given the empirical data $\mathcal{V}_1$, introduced in the previous section, one can formulate estimation problems in the RKHS $\mathcal{H}$, for example

$$\min_{f \in \mathcal{H}} \quad \sum_{t=1}^{N} c(f(\boldsymbol{x}_t) - y_t) + \lambda \|f\|^2 \tag{4.11}$$

where $c : \mathbb{R} \to \mathbb{R}_+ \cup \{0\}$ is a convex loss function and λ a positive regularization constant. For estimation problems of the form given by (4.11) it has been shown that the solution can be expressed as

$$f(\cdot) = \sum_{t=1}^{N} \alpha_t K(\boldsymbol{x}_t, \cdot). \tag{4.12}$$

This result is known as representer theorem and was first shown for $c(x) = x^2$ by Kimeldorf and Wahba [1971]. A more universal generalization has been proved in [Schölkopf et al., 2001]. Substituting the form of f given by (4.12) into (4.11) and using the least squares loss for c, one obtains

$$\min_{\alpha_t} \quad \sum_{t=1}^{N} \left(y_t - \sum_{n=1}^{N} \alpha_n K(\boldsymbol{x}_t, \boldsymbol{x}_n) \right)^2 + \lambda \sum_{t,n=1}^{N} \alpha_t \alpha_n K(\boldsymbol{x}_t, \boldsymbol{x}_n). \tag{4.13}$$

Constructing the kernel matrix $\boldsymbol{\Omega}$ in the same fashion as in the previous section, the solution to this now finite dimensional optimization is determined by the linear system

$$(\boldsymbol{\Omega} + \lambda \boldsymbol{I}_N)\boldsymbol{\alpha} = \boldsymbol{y}. \tag{4.14}$$

Therefore, for the least squares loss, both the model representation (4.12) and the estimation problem (4.14) are identical to the dual expressions for LS-SVMs (4.6) and (4.5) respectively, in case the bias term b is disregarded. In this formulation this approach is known as regularization network [Poggio and Girosi, 1990]. The representer theorem also holds for the ε-insensitive loss function and can be used to derive SVM models.

4.3 Handling of large data sets

The estimation of LS-SVMs on large data sets gives rise to two problems. Problem number one concerns the estimation of the model. Note that, as indicated in Algorithm 4.1, the dense kernel matrix has to be computed and stored. The memory requirements of the kernel matrix scale with $O(N^2)$ where N denotes the size of the training set $\mathcal{V}_1$. This limits the size of feasible problems. The second problem is related to model evaluations. As the solution of $\boldsymbol{\alpha}$ obtained by LS-SVMs is also dense, the evaluation of the model gets more and more costly as the size of the data set increases.

Problem one can be tackled with, possibly approximate, low rank factorizations of the kernel matrix. One approach based on the incomplete Cholesky decomposition has been proposed by Fine and Scheinberg [2002]. An alternative matrix factorization is the Nyström approximation [Williams and Seeger, 2001]. The latter will be discussed in the next subsection. A favorable property of the Nyström approximation is that it cannot only be used to obtain a low rank matrix factorization of the kernel matrix, but also to obtain a finite dimensional approximation of the feature map. Using this approximation one can then solve the estimation problem in the primal (4.2). Even though the solution is still not sparse in the dual variables $\boldsymbol{\alpha}$, it is sparse in the primal variable $\boldsymbol{w}$ however. Therefore this procedure also allows more efficient model evaluations. This approach is know as fixed-size LS-SVM and was initially proposed in [Suykens, Van Gestel, et al., 2002]. It is discussed in Subsection 4.3.3. For SVMs the sparsity of the model, and therefore the support vectors, follows directly from the dual optimization problem. In case of the Nyström approximation the support vectors have to be selected manually. A simple but efficient procedure to optimize this selection is briefly outlined in the last subsection. The procedure is based on the maximization of an entropy criterion and was proposed in the context of fixed-size LS-SVMs.

4.3.1 Nyström method

Given the data set $\mathcal{V}_1 = \{(\boldsymbol{x}_t, y_t)\}_{t=1}^N$, a finite number of eigenfunctions φ_n of the kernel can be obtained as described by Williams and Seeger [2001]. Therefore recall the integral equation (4.4) used to define the eigenfunctions of the kernel K in Mercer's theorem. Taking the underlying data distribution $p(\boldsymbol{x})$ into account it can be modified to

$$\int_S K(\boldsymbol{x}, \boldsymbol{y}) \tilde{\varphi}_n(\boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x} = \lambda_n \tilde{\varphi}_n(\boldsymbol{y}), \tag{4.15}$$

with solutions λ_n and $\tilde{\varphi}_n$, $n = 1, \dots, n_h$. Then, as performed in the kernel trick, one can define the elements of the feature map as $\varphi_n(\boldsymbol{x}) = \sqrt{\lambda_n}\tilde{\varphi}_n(\boldsymbol{x})$. To obtain a finite dimensional representation, the integral (4.15) can be approximated by means of its sample average. The eigenvalue decomposition of the kernel matrix $\boldsymbol{\Omega} = \boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^T$, with $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}_N$ and $\boldsymbol{\Sigma}$ diagonal, can then be used in order to compute a N-dimensional approximation of the feature map

$$\widehat{\boldsymbol{\varphi}}(\boldsymbol{z}) = \boldsymbol{\Sigma}^{-1}\boldsymbol{U}^T\boldsymbol{k}(\boldsymbol{z}) \tag{4.16}$$

where $\boldsymbol{k}(\boldsymbol{z}) = [K(\boldsymbol{x}_1, \boldsymbol{z}), \dots, K(\boldsymbol{x}_N, \boldsymbol{z})]^T$. As shown in [Suykens, Van Gestel, et al., 2002] this explicit finite-dimensional representation of the feature map $\widehat{\boldsymbol{\varphi}}$ can be used in the primal model formulation (4.2) to estimate $\boldsymbol{w}$ and c directly.

4.3.2 Approximation of the kernel matrix

The approximation of the feature map in (4.16) allows one to approximate the kernel matrix. The goal of the approximation is to obtain a low rank factorization for which the dual system (4.5) can be solved more efficiently. Being based on the Nyström approximation introduced above, the parameter that affects the rank of the decomposition is the dimension of the approximated feature map. This dimension is identical to the size of the sample used to approximate the feature map. Therefore one takes a subsample $\mathcal{V}_1^M$ of size M of $\mathcal{V}_1$ to compute the finite dimensional approximation of the feature map. In a second step the approximate feature map can be evaluated on the whole training set $\mathcal{V}_1$. This yields $\widehat{\boldsymbol{\Phi}} = [\widehat{\boldsymbol{\varphi}}(\boldsymbol{x}_1), \dots, \widehat{\boldsymbol{\varphi}}(\boldsymbol{x}_N)] \in \mathbb{R}^{M \times N}$ with the corresponding approximation of $\boldsymbol{\Omega}$ based on M samples given by $\widehat{\boldsymbol{\Omega}} = \widehat{\boldsymbol{\Phi}}^T \widehat{\boldsymbol{\Phi}}$. This low rank approximate factorization can then be used to efficiently solve (4.5). Note that the linear system can be solved in two steps with

$$b = \frac{\mathbf{1}_N^T \boldsymbol{A}^{-1} \boldsymbol{y}}{\mathbf{1}_N^T \boldsymbol{A}^{-1} \mathbf{1}_N} \quad \text{and} \tag{4.17a}$$

$$\boldsymbol{\alpha} = \boldsymbol{A}^{-1}(\boldsymbol{y} - b\mathbf{1}_N), \tag{4.17b}$$

with $\boldsymbol{A} = \widehat{\boldsymbol{\Omega}} + \gamma^{-1}\boldsymbol{I}_N$. Using the matrix inversion lemma and the low rank factorization of $\widehat{\boldsymbol{\Omega}}$, the inverse of $\boldsymbol{A}$ can be computed efficiently as $\boldsymbol{A}^{-1} = \gamma \boldsymbol{I}_N - \gamma \widehat{\boldsymbol{\Phi}}^T (\gamma^{-1}\boldsymbol{I}_M + \widehat{\boldsymbol{\Phi}}\widehat{\boldsymbol{\Phi}}^T)^{-1} \widehat{\boldsymbol{\Phi}}$. This reduces the complexity from one linear system of size $N + 1$ to three linear systems of size M and some matrix multiplications.

Algorithm 4.2 (Solution of LS-SVMs based on Nyström approximation of the kernel matrix)**.**

1. Pick a regularization parameter γ and a kernel function K (and its parameters).
2. Select a subset $\mathcal{V}_1^M$ of M samples with $\mathcal{V}_1^M \subset \mathcal{V}_1$.
3. Compute the kernel matrices $\boldsymbol{\Omega}_M \in \mathbb{R}^{M \times M}$ with $(\boldsymbol{\Omega}_M)_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{V}_1^M$ and $\boldsymbol{\Omega}_{M,N} \in \mathbb{R}^{M \times N}$ with $(\boldsymbol{\Omega}_{M,N})_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $\boldsymbol{x}_i \in \mathcal{V}_1^M$ and $\boldsymbol{x}_j \in \mathcal{V}_1$ respectively.
4. Construct the eigenvalue decomposition $\boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^T$ of $\boldsymbol{\Omega}_M$.
5. Evaluate the approximate feature map as $\widehat{\boldsymbol{\Phi}} = \boldsymbol{\Sigma}^{-1}\boldsymbol{U}^T\boldsymbol{\Omega}_{M,N}$.
6. Solve (4.17) to obtain $\boldsymbol{\alpha}$ and b.

The one-step-ahead predictor in this setting is given by

$$\hat{y}(z) = \boldsymbol{k}_M(z)^T \boldsymbol{\Omega}_M^{-1} \boldsymbol{\Omega}_{M,N} \boldsymbol{\alpha} + b, \tag{4.18}$$

with $k_M(z) = [K(x_1, z), \dots, K(x_M, z)]^T$ where $x_1, \dots, x_M$ are the samples forming $\mathcal{V}_1^M$.

4.3.3 Fixed size approach

The approach considering the matrix factorization of $\boldsymbol{\Omega}$ has two main disadvantages, i) the vector $\boldsymbol{\alpha}$ is of dimension N which makes the naive evaluation of (4.18) costly and ii) the estimation requires the solution of three linear systems and some matrix multiplications. Both disadvantages can be circumvented when using the approximate feature map $\widehat{\boldsymbol{\varphi}}$ in the primal problem (4.2) to estimate $\boldsymbol{w}$ and b directly. Using the same definition of $\widehat{\boldsymbol{\Phi}}$ as in the previous subsection, one can solve a finite dimensional approximation of the primal problem (4.2) based on $\widehat{\boldsymbol{\varphi}}$ by means of the linear system

$$\begin{bmatrix} \widehat{\boldsymbol{\Phi}}\widehat{\boldsymbol{\Phi}}^T + \gamma^{-1}\boldsymbol{I}_M & \widehat{\boldsymbol{\Phi}}\boldsymbol{1}_N \\ \boldsymbol{1}_N^T\widehat{\boldsymbol{\Phi}}^T & N \end{bmatrix} \begin{bmatrix} \boldsymbol{w} \\ b \end{bmatrix} = \begin{bmatrix} \widehat{\boldsymbol{\Phi}}\boldsymbol{y} \\ \boldsymbol{1}_N^T\boldsymbol{y} \end{bmatrix}. \tag{4.19}$$

The evaluation at a new point $\boldsymbol{z}$ is performed using the one-step-ahead predictor

$$\hat{y}(\boldsymbol{z}) = \boldsymbol{k}_M(\boldsymbol{z})^T \boldsymbol{U}\boldsymbol{\Sigma}^{-1}\boldsymbol{w} + b, \tag{4.20}$$

where $\boldsymbol{k}_M(\boldsymbol{z})$ is defined as in the last subsection. Note that the disadvantages mentioned at the beginning could — to a large extent — be resolved by an efficient implementation. However the fixed size approach described here is more straightforward as it does not need the dual formulation and special linear algebra to be exploited. Therefore it is often easier to adapt to modified primal problems. It will be the preferred method used in the following chapters.

The algorithm for this fixed size LS-SVM (FS-LS-SVM) procedure is almost identical to Algorithm 4.2 with the only difference in the model estimation step 6. It is reproduced here for later reference.

Algorithm 4.3 (Solution of LS-SVMs in the primal based on finite dimensional approximation of the feature map)**.**

1. Pick a regularization parameter γ and a kernel function K (and its parameters).
2. Select a subset $\mathcal{V}_1^M$ of M samples with $\mathcal{V}_1^M \subset \mathcal{V}_1$.
3. Compute the kernel matrices $\boldsymbol{\Omega}_M \in \mathbb{R}^{M\times M}$ with $(\boldsymbol{\Omega}_M)_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{V}_1^M$ and $\boldsymbol{\Omega}_{M,N} \in \mathbb{R}^{M\times N}$ with $(\boldsymbol{\Omega}_{M,N})_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $\boldsymbol{x}_i \in \mathcal{V}_1^M$ and $\boldsymbol{x}_j \in \mathcal{V}_1$ respectively.

Table 4.3: Summary of important relations of large scale implementations of LS-SVMs.

	LOW RANK LS-SVM	FS-LS-SVM
estimation problem	in the dual via (4.17)	in the primal via (4.19)
variables	$\boldsymbol{\alpha} \in \mathbb{R}^N$, b	$\boldsymbol{w} \in \mathbb{R}^M$, b
complexity	$3 \times O(M)$	$O(M+1)$
predictor	(4.18)	(4.20)
algorithm	4.2	4.3

4. Construct the eigenvalue decomposition $\boldsymbol{U\Sigma}^2\boldsymbol{U}^T$ of $\boldsymbol{\Omega}_M$.
5. Evaluate the approximate feature map as $\widehat{\boldsymbol{\Phi}} = \boldsymbol{\Sigma}^{-1}\boldsymbol{U}^T\boldsymbol{\Omega}_{M,N}$.
6. Solve (4.19) to obtain $\boldsymbol{w}$ and b.

4.3.4 Active selection of support vectors

In the previous two subsections one had to form a subset $\mathcal{V}_1^M$ of the whole training set $\mathcal{V}_1$ to reduce the computational complexity of the algorithms. Some straightforward methods for selecting a suitable subset of M samples are, i) deterministic subsampling by taking every $\sim N/M$-th sample and ii) stochastic subsampling by drawing M samples uniformly from $\mathcal{V}_1$. Both of these methods can work well in practice depending on the application at hand. In case the performance is not satisfactory, active selection of the support vectors, i.e. the samples in the subset $\mathcal{V}_1^M$, can give rise to models with better performance.

Whereas SVMs have an inherently active selection of their support vectors, through means of the ε-insensitive loss functions that induces sparsity, LS-SVMs have to rely on independent preprocessing steps. One method that works well and is computationally efficient is a form of entropy maximization [Girolami, 2002; Suykens, Van Gestel, et al., 2002]. In this case, the M support vectors are selected such that the quadratic Rényi entropy H_R is maximized. This entropy is given as

$$H_R = -\log \int p^2(\boldsymbol{x})d\boldsymbol{x},$$

with p the probability density of the selected support vectors. The quadratic entropy can be approximated on a finite subsample of size M [Girolami, 2002] by using

$$\int \hat{p}^2(\boldsymbol{x})d\boldsymbol{x} \simeq \frac{1}{M^2}\mathbf{1}_M^T\boldsymbol{\Omega}_M\mathbf{1}_M, \tag{4.21}$$

where $\boldsymbol{\Omega}_M$ is defined as before. The initial choices for the kernel function and an efficient bandwidth to be used in this procedure are addressed in detail in [De Brabanter et al., 2010]. The use of this active selection procedure can be quite important for large-scale problems. The optimality of the selection is related to the final accuracy that can be obtained for the modeling problem. By using an entropy maximization criterion, one can assure that the selected subsample is well spread over the entire data range, and it will not be concentrated in a certain area of the data set. The support vector selection is then an iterative procedure formalized in the following algorithm.

Algorithm 4.4 (Active support vector selection by maximization of Rényi entropy)**.**

1. Select a kernel function K and its parameters [De Brabanter et al., 2010].
2. Decompose $\mathcal{V}_1$ into a set $\mathcal{S}$ of size M and its complement $\mathcal{S}_C$ such that $\mathcal{V}_1 = \mathcal{S} + \mathcal{S}_C$.
3. Compute the approximate quadratic Rényi entropy $\widehat{H}_R$ of $\mathcal{S}$ via (4.21).
4. Pick one element of $\mathcal{S}_C$ and use it to replace another element of $\mathcal{S}$ and denote the new set by $\mathcal{S}^*$.
5. Compute the approximate quadratic Rényi entropy $\widehat{H}_R^*$ of $\mathcal{S}^*$ via (4.21).
6. If $H_R^* > H_R$ set $\mathcal{S} := \mathcal{S}^*$, $\mathcal{S}_C := \mathcal{V}_1 \setminus \mathcal{S}$ and $H_R := H_R^*$.
7. If not converged, go to 4.

4.4 Model selection

Two of the key concepts identified by Ljung, as outlined on page 20, are *validation* and *model fit*. Both of these terms are key elements for model selection. All models considered in this thesis are based on the LS-SVM model formulated in (4.2). However this formulation actually specifies a whole family of model classes. To fix a single model class that allows to estimate a model by solving (4.5), one has to choose i) a kernel function and its parameters, ii) the regularization parameter γ and iii) the structure of the regression vector $\boldsymbol{x}_t$. This section will briefly outline how these hyper-parameters can be selected. As the problem is nonconvex and in some cases not differentiable, the basic procedure is to define a criterion that assesses the quality of a candidate model. Then that criterion is used to rate a series of models and the best one is chosen.

One important element of the quality criterion for a particular model is the model fit. In the scope of this thesis only the root mean squared error (RMSE), defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{N'} \sum_{t=1}^{N'} (y_t - \hat{y}_t)^2}, \tag{4.22}$$

is used. Here it is important that the data used to evaluate $y_t - \hat{y}_t$ are usually not elements of the training set $\mathcal{V}_1$, which will be discussed in the next paragraph. In the context of system identification it is important how the predictions $\hat{y}_t$ are generated. One possible choice is to use the one-step-ahead predictors like (4.6) as specified in this and the following chapters. For all model structures containing past model outputs y_t such as in NARX models, it is possible to replace the measured quantities by their predictions. In case of NARX models, the measured values for y_t present in the regression vector $\boldsymbol{x}_t$ can be replaced by predictions $\hat{y}_t$. When this is done recursively for all u_t, then this is called simulation mode. As the computational complexity of evaluating the RMSE in simulation mode is higher than with one-step-ahead predictions and there often is little to no benefit, model selection is always performed based on one-step-ahead predictions throughout this thesis.

As mentioned in the last paragraph the RMSE should not be computed on the training set $\mathcal{V}_1$. Doing this would, especially for nonlinear models, lead to overfitting which in turn results in models that do not generalize well to unseen data. Popular ways to select models with good generalization performance are Bayesian inference [MacKay, 1999; Suykens, Van Gestel, et al., 2002] and validation techniques. For their simplicity only validation techniques are considered further on. Three basic validation techniques are i) using an independent validation set, ii) k-fold cross-validation and iii) leave-one-out cross-validation. For k-fold cross validation the whole data set $\mathcal{D}$ is partitioned into k subsets $\mathcal{D}_k$ of equal size. Then one estimates k models where for the m-th model the training set is $\mathcal{V}_1 := \bigcup_{n=1, n\neq m}^{k} \mathcal{D}_n$ and the RMSE is computed on $\mathcal{D}_m$. In a final step the RMSE values for all k models are combined to an overall score. Leave-one-out cross-validation is a special case of k-fold cross-validation in which one considers N subsets $\mathcal{D}_k$, each containing just a single point, for a data set $\mathcal{D}$ of size N. In system identification it is often relatively cheap to acquire large amounts of data. Since both cross-validation techniques require the estimation of multiple models, which increases the computational complexity, in this thesis a validation set is used throughout. A validation set is the easiest validation technique which simply partitions the data set $\mathcal{D}$ as $\mathcal{V}_1 \cup \mathcal{V}_2$. Then the RMSE is just evaluated on $\mathcal{V}_2$. More comprehensive information on the topic can for example be found in [Hastie et al., 2009].

Using a validation set in combination with the RMSE a given model can easily be tested for its (relative) quality. The candidate choices for the kernel, its parameters, the regularization constant and, if considered, the elements of the regression vector can be generated in many different ways. This includes global search algorithms like genetic algorithms or simulated annealing, direct search algorithms like the simplex method or coordinate search and a simple grid based search. As with the previous choices, also for the search method the simplest technique is generally used in this thesis, namely grid search. A rough outline for a model selection procedure based on a validation set is given below.

Algorithm 4.5 (Model selection for kernel based model)**.**

1. Choose a regression vector $\boldsymbol{x}_t$ and pick nominal values for the regularization parameters γ, the kernel function K (and its parameters).
2. Estimate a single model with Algorithm 4.1 on $\mathcal{V}_1$ a subset of all measured data $\mathcal{D}$ to obtain model parameter $\boldsymbol{\alpha}$, b.
3. Predict values using this model by evaluating (4.6) on a different subset $\mathcal{V}_2$ of the measured data $\mathcal{D}$.
4. Compute a cost function, e.g. the root mean squared error (RMSE) comparing the predicted with the measured values.
5. If stopping criterion is satisfied, stop, otherwise change one of the parameters and repeat from 2.

Part II

Original work

Partially linear models with orthogonality

5

The main goal of this work is to explore model structures for nonlinear systems based on Least Squares Support Vector Machines. The model structures should either incorporate prior knowledge about the system or extend the class of systems that can be represented. This chapter considers the case where a parametric model for the system already exists. The objective is to complement the existing model with a kernel based model while ensuring that the information in the parametric model is retained.

The general model structure is shown in Figure 5.1. Models of this class are usually referred to as semi-parametric models. To ensure that the identification problem is convex it is assumed that the parametric model is linear in the parameters. In this case the model class is known as partially linear models

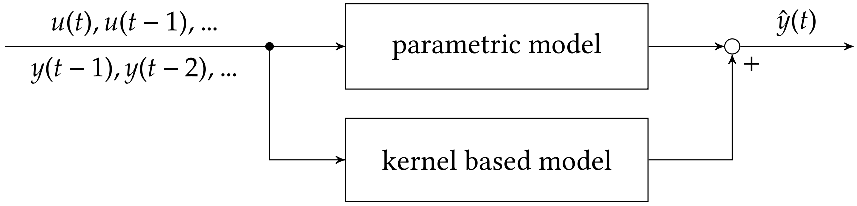

Figure 5.1: Partially linear model with parametric and kernel based part and ARX type regression variables.

[Härdle et al., 2000]. Their main use is to extend an existing parametric model to achieve improved prediction performance.

Kernel based models of this form have been considered for example in [Wahba, 1984; Speckman, 1988; Wahba, 1990; Espinoza et al., 2005a; Li et al., 2006; Xu and Chen, 2009]. All these formulations face the problem that parts of the studied system may be modeled by the parametric part as well as by the kernel based model. Thus an ambiguity arises and the estimate for the parametric part might loose the connection to an interpretation. There are different approaches to tackle this problem, ranging from neglecting the issue for highest prediction performance [Li et al., 2006; Xu and Chen, 2009] over constraints on the explanatory variables [Speckman, 1988; Espinoza et al., 2005a] to constraints on the admissible parametric and nonparametric model classes [Wahba, 1984, 1990].

The novel contribution of this chapter is to impose no restrictions on the model classes or the explanatory variables. Nevertheless the estimate for the parametric model part should be of the same quality as without the additional kernel based model. This is achieved by introducing a new constraint in the estimation problem that ensures empirical orthogonality of the predictions of the two model parts.

Structure of the chapter The next section will review prior work on kernel based partially linear systems with a focus on formulations in a primal-dual framework. Section 5.2 will analyze the estimate of the parametric model part in case necessary assumptions are not satisfied and introduces the orthogonality constraint which gives rise to improved estimates. In Section 5.3 reformulations of the estimation problem are considered. The first reformulation derives a regularization term that captures the orthogonality constraint such that the parametric model part and the kernel based model part can be estimated using separate estimation problems. The second reformulation modifies the kernel function such that an equivalent kernel function is obtained which embodies the orthogonality constraint. The partially linear modeling approach with orthogonality constraint is generalized from the least square loss used for LS-SVMs to the ε-insensitive loss used in SVMs in Section 5.4. Section 5.5 considers the partially linear models in a RKHS setting instead of the primal-dual framework as considered in the rest of this chapter. Finally Section 5.6 illustrates the effect of the orthogonality constraint on simulated as well as real world data.

5.1 Review of kernel based partially linear models

For LS-SVM based models, partially linear models were introduced by Espinoza et al. [2005a]. They are based on the following assumption.

Assumption 5.1. The explanatory variables $\boldsymbol{x} \in \mathbb{R}^D$ can be partitioned into $\boldsymbol{x}_a \in \mathbb{R}^{D_a}$ that are part of the parametric model and $\boldsymbol{x}_b \in \mathbb{R}^{D_b}$ that are part of the nonparametric model such that $\boldsymbol{x} = [\boldsymbol{x}_a^T, \boldsymbol{x}_b^T]^T$ and $D = D_a + D_b$. Furthermore the variables $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ are independent.

Then a predictive equation can be stated as

$$\hat{h}(\boldsymbol{x}) = \boldsymbol{\theta}^T \boldsymbol{\psi}(\boldsymbol{x}_a) + \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_b) + b, \tag{5.1}$$

where b, $\boldsymbol{w}$ are the parameters and $\boldsymbol{\varphi}$ is the feature map as introduced in Chapter 4 respectively. The parametric model is defined by M basis functions $\psi_i : \mathbb{R}^{D_a} \to \mathbb{R}$ compactly written as $\boldsymbol{\psi} = [\psi_1, \dots, \psi_M]^T$ and the parameter vector $\boldsymbol{\theta} \in \mathbb{R}^M$. The predictive equation in (5.1) is a generalization of [Espinoza et al., 2005a] as any linear-in-parameters parametric model can be used and not just linear models.

Given observation data $\{(\boldsymbol{x}_{a,t}, \boldsymbol{x}_{b,t}, y_t)\}_{t=1}^N$ a model given by the form in (5.1) can be estimated by solving the convex optimization problem

$$\begin{aligned} \min_{\boldsymbol{w},\boldsymbol{\theta},b,e_t} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{t=1}^N e_t^2 \\ \text{subject to} \quad & y_t = \boldsymbol{\theta}^T \boldsymbol{\psi}(\boldsymbol{x}_{a,t}) + \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_{b,t}) + b + e_t, \qquad t = 1, \dots, N, \end{aligned} \tag{5.2}$$

with regularization constant $\gamma > 0$. The optimal model representation for the kernel based model is

$$\hat{h}(\boldsymbol{z}) = \boldsymbol{\theta}^T \boldsymbol{\psi}(\boldsymbol{z}_a) + \sum_{t=1}^N \alpha_t K(\boldsymbol{x}_{b,t}, \boldsymbol{z}_b) + b \tag{5.3}$$

which follows from Lagrangian duality. The parameters α_t are Lagrange multipliers corresponding to the equality constraints in (5.2). They, as well as the other parameters $\boldsymbol{\theta}$ and b, can be obtained from the linear system

$$\begin{bmatrix} \boldsymbol{\Omega} + \gamma^{-1}\boldsymbol{I}_N & \boldsymbol{\Psi}^T & \boldsymbol{1}_N \\ \boldsymbol{\Psi} & \boldsymbol{0}_{\boxtimes} & \boldsymbol{0}_M \\ \boldsymbol{1}_N^T & \boldsymbol{0}_M^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\theta} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{0}_M \\ 0 \end{bmatrix}. \tag{5.4}$$

The matrix $\boldsymbol{\Psi} \in \mathbb{R}^{M\times N}$ is defined as $[\boldsymbol{\psi}(\boldsymbol{x}_{a,1}), \dots, \boldsymbol{\psi}(\boldsymbol{x}_{a,N})]$ and the elements of the kernel matrix $\boldsymbol{\Omega}$ are given by $\Omega_{ij} = K(\boldsymbol{x}_{b,i}, \boldsymbol{x}_{b,j})$ for $i, j = 1, \dots, N$.

5.2 Imposing orthogonality constraints

The need to partition the explanatory variables as in Assumption 5.1 is an unnecessary restriction. In the context of Splines or more general for RKHSs, less restrictive assumptions are known as will be discussed in Section 5.5. In this section the consequence of not partitioning the variables will be analyzed. Then a modification of the estimation problem is proposed that mitigates that disadvantage. Therefore consider (5.2) without partitioned variables

$$\begin{aligned}
&\min_{\boldsymbol{\theta},\boldsymbol{w},b,e_t} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma\sum_{t=1}^{N} e_t^2 \\
&\text{subject to} \quad y_t = \boldsymbol{\theta}^T\boldsymbol{\psi}(\boldsymbol{x}_t) + \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t, \quad t = 1, \dots, N,
\end{aligned} \tag{5.5}$$

where $\boldsymbol{\varphi} : \mathbb{R}^D \to \mathbb{R}^{n_h}$.

5.2.1 Parametric estimates under violated assumptions

The model ambiguity between the parametric and nonparametric parts can be seen from the estimates obtained for $\boldsymbol{\theta}$. To simplify the presentation it is assumed that the measurements y_t are zero mean and therefore b in (5.5) is equal to zero. Furthermore the matrix $\boldsymbol{\Psi}$ is supposed to be of full rank and defined as $\boldsymbol{\Psi} = [\boldsymbol{\psi}(\boldsymbol{x}_1), \dots, \boldsymbol{\psi}(\boldsymbol{x}_N)]$. Consequently the value for $\boldsymbol{\theta}$ obtained from (5.5) is

$$\hat{\boldsymbol{\theta}}_{PL} = (\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}(\boldsymbol{y} - \boldsymbol{\Phi}^T\hat{\boldsymbol{w}}), \tag{5.6}$$

where $\hat{\boldsymbol{w}}$ is the estimate following from $(\boldsymbol{\Phi}\boldsymbol{P}_{\Psi}^{\perp}\boldsymbol{\Phi}^T + \gamma^{-1}\boldsymbol{I}_N)\boldsymbol{w} = \boldsymbol{\Phi}\boldsymbol{P}_{\Psi}^{\perp}\boldsymbol{y}$ and $\boldsymbol{P}_{\Psi}^{\perp} = \boldsymbol{I}_N - \boldsymbol{\Psi}^T(\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}$ is the projector onto the nullspace of $\boldsymbol{\Psi}$.

Under the assumption that the variables in (5.2) are partitioned such that $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ are independent, the term $\boldsymbol{\Phi}\boldsymbol{\Psi}^T$ goes to zero as the number of samples N approaches infinity [Espinoza et al., 2005a]. For a finite amount of samples this does not hold even if the restrictive partitioning of the variables is carried out. Note that in the limit one obtains $\hat{\boldsymbol{\theta}}_{PL} \to \hat{\boldsymbol{\theta}}_{OLS} = (\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\boldsymbol{y}$, the ordinary least squares (OLS) estimate.

Therefore in the case that $\boldsymbol{\Phi}\boldsymbol{\Psi}^T$ is empirically nonzero, the estimate $\hat{\boldsymbol{\theta}}_{PL}$ depends not only on the data, but also the kernel K and the regularization constant γ controlling the complexity of the nonparametric model. This is a manifestation of the model ambiguity mentioned at the beginning.

Remark 5.1 (Ordinary least squares). It is important to note that while the partially linear structure is able to improve the prediction performance of the composite model, it is however not able to improve the estimate of the parametric part. Assume that the variables are partitioned and the sample size is large enough such that the partial linear estimate $\hat{\boldsymbol{\theta}}_{PL}$ coincides with the OLS solution $\hat{\boldsymbol{\theta}}_{OLS}$. Therefore the estimate for the parametric model part will be given by the ordinary least squares estimate (unless a different overall cost function is used).

In the ideal setting the data is generated by a true underlying system of the form

$$y_t = \boldsymbol{\theta}_0^T \psi(\boldsymbol{x}_t) + \rho(\boldsymbol{x}_t) + \varepsilon_t,$$

where $\rho(\cdot)$ is the part of the underlying system that cannot be captured by the parametric model. Then for a noise term ε_t that is zero mean, Gaussian white noise with finite variance and independent of $\boldsymbol{x}_t$, the OLS estimate is the best linear unbiased estimate (BLUE). It is given by

$$\hat{\boldsymbol{\theta}}_{OLS} = \boldsymbol{\theta}_0 + (\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}(\boldsymbol{\varepsilon} + \boldsymbol{\rho}),$$

with $\boldsymbol{\varepsilon} = [\varepsilon_1, \dots, \varepsilon_N]^T$ and $\boldsymbol{\rho} = [\rho(\boldsymbol{x}_1), \dots, \rho(\boldsymbol{x}_N)]^T$. Note that the noise contribution goes to zero for increasing sample sizes. What remains is the contribution that depends only on the fraction of the true underlying system that is not captured by the parametric model.

Remark 5.2 (Modeling of residuals). In view of the last remark it would seem beneficial to start with the OLS estimate $\hat{\boldsymbol{\theta}}_{OLS}$ and compute the residuals $r_t = y_t - \boldsymbol{\theta}^T \psi(\boldsymbol{x}_t)$. However modeling these residuals r_t with a nonparametric model gives rise to low prediction performances as will be illustrated in the experimental section.

5.2.2 Imposing orthogonality

While a new model should have a comparable predictive performance as existing partially linear models, the estimate of the parametric model given by (5.6) should also always be as good as the OLS estimate. This should hold for finite sample sizes as well as when the independence assumption is not satisfied. To achieve this goal, the term $(\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\boldsymbol{\Phi}^T\hat{\boldsymbol{w}}$ has to be zero which follows from comparing (5.6) with the OLS estimate $\hat{\boldsymbol{\theta}}_{OLS} = (\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\boldsymbol{y}$. This holds if and only if

$$\boldsymbol{\Psi}(\boldsymbol{\Phi}^T\hat{\boldsymbol{w}} + \mathbf{1}_N\hat{b}) = \mathbf{0}_M \tag{5.7}$$

is satisfied. Here the assumption that the output signal y_t is zero mean and therefore the bias term b is zero has been dropped. To obtain an improved partially linear model this relation is imposed as an additional constraint in problem (5.5). This can be interpreted as (empirical) orthogonality of the parametric and nonparametric models. Another interpretation is that the nonparametric model is constrained such that its predictions are uncorrelated with the predictions of any function with the chosen parametrization. For a given data set $\{(x_t, y_t)\}_{t=1}^N$ the modified optimization problem is then

$$\begin{aligned} \min_{\boldsymbol{\theta}, \boldsymbol{w}, b, e_t} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{t=1}^N e_t^2 \\ \text{subject to} \quad & y_t = \boldsymbol{\theta}^T\boldsymbol{\psi}(x_t) + \boldsymbol{w}^T\boldsymbol{\varphi}(x_t) + b + e_t, \quad t = 1, \dots, N, \\ & \sum_{t=1}^N \psi_i(x_t)(\boldsymbol{w}^T\boldsymbol{\varphi}(x_t) + b) = 0, \quad i = 1, \dots, M. \end{aligned} \tag{5.8}$$

Note that for kernel based models the feature map $\boldsymbol{\varphi}(x)$ is only defined implicitly by a kernel function K at the dual level. The basis functions $\boldsymbol{\psi}(x)$ however belong to the (existing) parametric model and need to be explicitly defined.

5.2.3 Dual problem: model representation and estimation

As mentioned earlier in Chapter 4 for some kernels like the RBF kernel, the feature map $\boldsymbol{\varphi}$ is not only defined implicitly, but furthermore infinite dimensional. Thus it is in general not possible to directly solve the primal problem in (5.8). Therefore the problem has to be solved in the dual which is formalized in the following Lemma.

Lemma 5.1. *The solution to* (5.8) *is given by the linear system*

$$\begin{bmatrix} \boldsymbol{\Omega} + \gamma^{-1}\boldsymbol{I}_N & \boldsymbol{\Omega}\boldsymbol{\Psi}^T & \boldsymbol{1}_N & \boldsymbol{\Psi}^T \\ \boldsymbol{\Psi}\boldsymbol{\Omega} & \boldsymbol{\Psi}\boldsymbol{\Omega}\boldsymbol{\Psi}^T & \boldsymbol{\Psi}\boldsymbol{1}_N & \boldsymbol{0}_\boxtimes \\ \boldsymbol{1}_N^T & \boldsymbol{1}_N^T\boldsymbol{\Psi}^T & 0 & \boldsymbol{0}_M^T \\ \boldsymbol{\Psi} & \boldsymbol{0}_\boxtimes & \boldsymbol{0}_M & \boldsymbol{0}_\boxtimes \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \\ b \\ \boldsymbol{\theta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{0}_M \\ 0 \\ \boldsymbol{0}_M \end{bmatrix}. \tag{5.9}$$

The variables $\boldsymbol{\beta} \in \mathbb{R}^M$ are the Lagrange multipliers for the newly introduced orthogonality constraints in (5.8) *and can be used to express the dual predictive model as*

$$\hat{h}(z) = \sum_{t=1}^N \eta_t K(x_t, z) + \boldsymbol{\theta}^T\boldsymbol{\psi}(z) + b$$

with $\eta_t = \alpha_t + \boldsymbol{\beta}^T\boldsymbol{\psi}(x_t)$.

Proof. The Lagrangian for (5.8) is

$$\mathscr{L}(\boldsymbol{w}, \boldsymbol{\theta}, b, e_t, \alpha_t, \beta_i) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma\sum_{t=1}^{N} e_t^2$$
$$-\sum_{t=1}^{N}\alpha_t(\boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t)+\boldsymbol{\theta}^T\boldsymbol{\psi}(\boldsymbol{x}_t)+b+e_t-y_t)-\sum_{i=1}^{M}\beta_i\sum_{t=1}^{N}\psi_i(\boldsymbol{x}_t)(\boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t)+b).$$

The corresponding KKT conditions for optimality are

$$\begin{aligned}
\mathbf{0} &= \frac{\partial\mathscr{L}}{\partial\boldsymbol{w}} = \boldsymbol{w} - \sum_{t=1}^{N}(\alpha_t + \boldsymbol{\beta}^T\boldsymbol{\psi}(\boldsymbol{x}_t))\boldsymbol{\varphi}(\boldsymbol{x}_t),\\
\mathbf{0} &= \frac{\partial\mathscr{L}}{\partial\boldsymbol{\theta}} = -\sum_{t=1}^{N}\alpha_t\boldsymbol{\psi}(\boldsymbol{x}_t),\\
0 &= \frac{\partial\mathscr{L}}{\partial b} = -\left(\sum_{t=1}^{N}\alpha_t + \boldsymbol{\beta}^T\boldsymbol{\psi}(\boldsymbol{x}_t)\right),\\
0 &= \frac{\partial\mathscr{L}}{\partial e_t} = \gamma e_t - \alpha_t, \quad t = 1, \dots, N,\\
0 &= \frac{\partial\mathscr{L}}{\partial\alpha_t} = y_t - \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) - \boldsymbol{\theta}^T\boldsymbol{\psi}(\boldsymbol{x}_t) - b - e_t, \quad t = 1 \dots, N,\\
0 &= \frac{\partial\mathscr{L}}{\partial\beta_i} = -\sum_{t=1}^{N}\psi_i(\boldsymbol{x}_t)(\boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b), \quad i = 1, \dots, M.
\end{aligned}$$

Rewriting the conditions for $\boldsymbol{w}$ and e_t and substituting them into $\partial\mathscr{L}/\partial\alpha_t$ yields

$$y_t = \sum_{k=1}^{N}(\alpha_k + \boldsymbol{\beta}^T\boldsymbol{\psi}(\boldsymbol{x}_k))K(\boldsymbol{x}_k, \boldsymbol{x}_t) + \boldsymbol{\theta}^T\boldsymbol{\psi}(\boldsymbol{x}_t) + b + \frac{1}{\gamma}\alpha_t.$$

Substitution of $\boldsymbol{w}$ into $\partial\mathscr{L}/\partial\beta_i$ yields

$$\sum_{t=1}^{N}\psi_i(\boldsymbol{x}_t)\sum_{k=1}^{N}(\alpha_k + \boldsymbol{\beta}^T\boldsymbol{\psi}(\boldsymbol{x}_k))K(\boldsymbol{x}_k, \boldsymbol{x}_t) + b = 0.$$

Here, the inner products $\boldsymbol{\varphi}(\boldsymbol{x}_k)^T\boldsymbol{\varphi}(\boldsymbol{x}_t)$ were replaced by a positive definite kernel $K(\boldsymbol{x}_k, \boldsymbol{x}_t)$ using the kernel trick. The rewritten KKT conditions $\partial\mathscr{L}/\partial\alpha_t$ and $\partial\mathscr{L}/\partial\beta_i$ together with $\partial\mathscr{L}/\partial b$ and $\partial\mathscr{L}/\partial\boldsymbol{\theta}$ correspond to the dual system (5.9). □

5.3 Improved estimation schemes and representations

The optimization problem given in (5.8) and its dual in (5.9) allow for the flexible estimation of partially linear models in a kernel based setting. Rewriting parts of the problem, alternative formulations can be obtained. One reformulation allows the estimation of the parametric model part to be carried out separately from the kernel based part and is based on a special regularization matrix for the kernel based model. The second reformulation derives an equivalent kernel that embeds the information of the orthogonality constraint into the kernel, which can in turn be used in the classical partially linear setting (5.5).

5.3.1 Separation principle

As detailed in Section 5.2, the goal of the orthogonality constraint is to obtain the OLS solution for the parametric model part. However, it is not possible to use the corresponding residuals to estimate the kernel based model as mentioned in Remark 5.2. The following lemma describes how the orthogonality constraint imposed on the kernel based model can be transformed into a special regularization matrix. This special regularization then allows the kernel based model to be estimated from the residuals in a separate estimation step..

Lemma 5.2. *Under the assumption that $\boldsymbol{\Psi}$ has full row rank, the estimation of the model can be performed in two stages as follows:*

1. *The parameter vector $\boldsymbol{\theta}$ can be estimated using ordinary least squares $\hat{\boldsymbol{\theta}}_{OLS} = (\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\boldsymbol{y}$.*
2. *The kernel based model can then be identified by solving the reduced linear system*

$$\begin{bmatrix} \boldsymbol{\Omega} + \gamma^{-1}\boldsymbol{P}_{\Psi}^{\perp} & \mathbf{1}_N \\ \mathbf{1}_N^T & \mathbf{0}_{\boxtimes} \end{bmatrix} \begin{bmatrix} \boldsymbol{\eta} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{r} \\ 0 \end{bmatrix} \tag{5.11}$$

with $\boldsymbol{r} = \boldsymbol{P}_{\Psi}^{\perp}\boldsymbol{y} = \boldsymbol{y} - \boldsymbol{\Psi}^T\hat{\boldsymbol{\theta}}_{OLS}$ and $\boldsymbol{P}_{\Psi}^{\perp} = \boldsymbol{I}_N - \boldsymbol{\Psi}^T(\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}$ as in Subsection 5.2.1.

Proof. Let $\mathfrak{R}_1$ to $\mathfrak{R}_4$ denote the block rows of (5.9). Isolating $\boldsymbol{\Omega}\boldsymbol{\alpha}$ in $\mathfrak{R}_1$ and substituting this into $\mathfrak{R}_2$ yields $\boldsymbol{\Psi}\boldsymbol{y} = \boldsymbol{\Psi}\boldsymbol{\Psi}^T\boldsymbol{\theta} + \gamma^{-1}\boldsymbol{\Psi}\boldsymbol{\alpha}$. Exploiting $\mathfrak{R}_4$ one obtains $\boldsymbol{\theta} = (\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\boldsymbol{y}$. Performing a change of variables $\boldsymbol{\eta} = \boldsymbol{\alpha} + \boldsymbol{\Psi}^T\boldsymbol{\beta}$

and substitution of this relation into $\Re_4$ yields $\boldsymbol{\beta} = (\boldsymbol{\Psi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\boldsymbol{\eta}$. Stating $\Re_3$ in terms of $\boldsymbol{\eta}$ yields $\mathbf{1}_N^T\boldsymbol{\eta} = 0$ and $\Re_1$ can be rewritten as $(\boldsymbol{\Omega} + \gamma^{-1}\boldsymbol{P}_{\Psi}^{\perp})\boldsymbol{\eta} + \mathbf{1}_N b = \boldsymbol{P}_{\Psi}^{\perp}\boldsymbol{y}$. □

Note that $\Re_2$ in terms of $\boldsymbol{\eta}$ reads $\boldsymbol{\Psi}(\boldsymbol{\Omega}\boldsymbol{\eta} + \mathbf{1}_N b) = \mathbf{0}_M$. This nicely illustrates that the correlation between the kernel based model $\boldsymbol{\Omega}\boldsymbol{\eta} + \mathbf{1}_N b$ and the basis functions $\boldsymbol{\Psi}$ is zero on the estimation data.

5.3.2 Equivalent kernel

In various situations it can be easier to change the kernel function than to change the estimation problem. In that case the information of the orthogonality constraint can be included in a modified kernel function. For simplicity of presentation the bias term b is set to zero for the remainder of this section. This is no limitation since a static offset can be included in the parametric basis. In that case the orthogonality constraint will ensure that b is chosen as zero.

Lemma 5.3. *Under the assumption that $\boldsymbol{\Psi}\boldsymbol{\Omega}\boldsymbol{\Psi}^T$ is invertible, an equivalent feature map $\boldsymbol{\varphi}_{eq}$ that embeds the orthogonality constraint is given by $\boldsymbol{\varphi}_{eq}(\cdot) = \boldsymbol{P}_{\Phi\Psi}^{\perp}\boldsymbol{\varphi}(\cdot)$. The projector $\boldsymbol{P}_{\Phi\Psi}^{\perp}$ is defined as*

$$\boldsymbol{P}_{\Phi\Psi}^{\perp} = \boldsymbol{I}_N - \boldsymbol{\Phi}\boldsymbol{\Psi}^T(\boldsymbol{\Psi}\boldsymbol{\Phi}^T\boldsymbol{\Phi}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\boldsymbol{\Phi}^T$$

with $\boldsymbol{\Phi} = [\boldsymbol{\varphi}(x_1), \dots, \boldsymbol{\varphi}(x_N)]$. Application of the kernel trick to the equivalent feature map yields the equivalent kernel function

$$K_{eq}(x, y) = K(x, y) - \boldsymbol{k}(x)^T\boldsymbol{\Psi}^T(\boldsymbol{\Psi}\boldsymbol{\Omega}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\boldsymbol{k}(y) \tag{5.12}$$

with $\boldsymbol{k}(z) = [K(x_1, z), \dots, K(x_N, z)]^T$ and where $\boldsymbol{\Omega} = \boldsymbol{\Phi}^T\boldsymbol{\Phi}$ is the original kernel matrix.

Proof. Substitution of the expansion for $\boldsymbol{w}$ obtained from the KKT conditions in the proof of Lemma 5.1 into the KKT condition for $\boldsymbol{\beta}$ gives $\boldsymbol{\Psi}\boldsymbol{\Phi}^T\boldsymbol{\Phi}(\boldsymbol{\alpha} + \boldsymbol{\Psi}\boldsymbol{\beta}) = \mathbf{0}_M$. Then, assuming invertibility, the Lagrange multipliers $\boldsymbol{\beta}$ can be expressed as $\boldsymbol{\beta} = -(\boldsymbol{\Psi}\boldsymbol{\Omega}\boldsymbol{\Psi}^T)^{-1}\boldsymbol{\Psi}\boldsymbol{\Omega}\boldsymbol{\alpha}$. Using this solution in the expansion of $\boldsymbol{w}$ yields $\boldsymbol{w} = \boldsymbol{P}_{\Phi\Psi}^{\perp}\boldsymbol{\Phi}\boldsymbol{\alpha}$. Reading this column-wise for the columns of $\boldsymbol{\Phi}$, the equivalent feature map can be extracted. The equivalent kernel follows directly from $K_{eq}(x, y) = \boldsymbol{\varphi}_{eq}(x)^T\boldsymbol{\varphi}_{eq}(y) = \boldsymbol{\varphi}(x)^T\boldsymbol{P}_{\Phi\Psi}^{\perp}\boldsymbol{\varphi}(y)$. □

Corollary 5.4. *Using the equivalent feature map, problem* (5.8) *can be solved equivalently as*

$$\begin{aligned}
&\min_{\boldsymbol{\theta},\boldsymbol{w},e_t} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma\sum_{t=1}^{N} e_t^2 \\
&\text{subject to} \quad y_t = \boldsymbol{\theta}^T\boldsymbol{\psi}(\boldsymbol{x}_t) + \boldsymbol{w}^T\boldsymbol{\varphi}_{eq}(\boldsymbol{x}_t) + e_t, \qquad t = 1,\dots,N.
\end{aligned}$$

In the kernel based form, the dual variables can be obtained from the linear system

$$\begin{bmatrix} \boldsymbol{\Omega}_{eq} + \gamma^{-1}\boldsymbol{I}_N & \boldsymbol{\Psi}^T \\ \boldsymbol{\Psi} & \mathbf{0}_{\boxtimes} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\theta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ \mathbf{0}_M \end{bmatrix},$$

where $(\boldsymbol{\Omega}_{eq})_{ij} = K_{eq}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ *for* $i, j = 1, \dots, N$. *This model can be evaluated at a new point* z *through*

$$\hat{h}(z) = \sum_{t=1}^{N} \alpha_t K_{eq}(\boldsymbol{x}_t, z) + \boldsymbol{\theta}^T\boldsymbol{\psi}(z). \tag{5.13}$$

5.4 Extension to different loss functions

The orthogonality constraint for partially linear models can be generalized from the least square loss to other (convex) loss functions. For example the orthogonality constraint can be integrated into a SVM [Vapnik, 1998] with ε-insensitive loss. The corresponding primal formulation is

$$\begin{aligned}
&\min_{\boldsymbol{\theta},\boldsymbol{w},b,\xi_t} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{t=1}^{N}\xi_t \\
&\text{subject to} \quad |y_t - \boldsymbol{\theta}^T\boldsymbol{\psi}(\boldsymbol{x}_t) - \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) - b| \le \varepsilon + \xi_t, \quad t = 1,\dots,N, \\
&\qquad\qquad\quad \xi_t \ge 0, \quad t = 1,\dots,N, \\
&\qquad\qquad\quad \sum_{t=1}^{N}\psi_i(\boldsymbol{x}_t)(\boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b) = 0, \quad i = 1,\dots,M.
\end{aligned} \tag{5.14}$$

The Lagrange dual can be obtained by solving the KKT system

$$\begin{aligned}
&\max_{\alpha_t,\beta_i} \quad \boldsymbol{y}^T\boldsymbol{\alpha} - \frac{1}{2}(\boldsymbol{\alpha} + \boldsymbol{\Psi}^T\boldsymbol{\beta})^T\boldsymbol{\Omega}(\boldsymbol{\alpha} + \boldsymbol{\Psi}^T\boldsymbol{\beta}) - \varepsilon\mathbf{1}^T|\boldsymbol{\alpha}| \\
&\text{subject to} \quad -C \le \alpha_t \le C, \quad t = 1,\dots,N, \\
&\qquad\qquad\quad \boldsymbol{\Psi}\boldsymbol{\alpha} = \mathbf{0}_M, \\
&\qquad\qquad\quad \mathbf{1}_N^T(\boldsymbol{\alpha} + \boldsymbol{\Psi}^T\boldsymbol{\beta}) = 0.
\end{aligned} \tag{5.15}$$

where $|\boldsymbol{\alpha}|$ is understood as the element-wise absolute value. The predictive model is identical to the one stated in Lemma 5.1. The primal variables b and θ_i can for example be obtained from an interior point solver and correspond to the dual variables of the equality constraints in (5.15).

Extensions to Huber's robust loss function and other convex loss functions can be derived in a similar fashion.

5.5 Equivalent RKHS approach

So far, the orthogonality constraint was presented in an optimization setting. In addition to the primal-dual framework, optimization in function spaces is a popular technique employed in machine learning. As introduced in Chapter 4, support vector techniques can also be formulated in reproducing kernel Hilbert spaces. The earliest results on partially linear models in RKHSs however consider smoothing splines [Wahba, 1990, Ch. 1].

5.5.1 Partially linear models in RKHSs

In general, for partially linear models with a nonparametric model in a RKHS one assumes that two spaces $\mathcal{F}$ and $\mathcal{G}$ are given. The mandatory condition on these spaces, which generalizes Assumption 5.1, is:

Assumption 5.2. One can form the direct sum of the two spaces $\mathcal{F} \oplus \mathcal{G} = \mathcal{H}$ such that for any $f \in \mathcal{F}$ and $g \in \mathcal{G}$ the function $f + g$ admits a unique decomposition.

In the following the space $\mathcal{G}$ is a RKHS induced by a kernel K, while the space $\mathcal{F}$ can be expressed with a finite number of M basis functions $\psi_i : \mathbb{R}^D \to \mathbb{R}$, $i = 1, \dots, M$. Given observational data $\{(\boldsymbol{x}_t, y_t)\}_{t=1}^N$ with $\boldsymbol{x}_t \in \mathbb{R}^D$ and $y_t \in \mathbb{R}$, a function $h \in \mathcal{H}$ can be estimated from

$$\min_{h \in \mathcal{H}} \quad \sum_{t=1}^{N} (y_t - h(\boldsymbol{x}_t))^2 + \lambda \|\mathscr{P}_{\mathcal{G}} h\|_{\mathcal{H}}^2 \tag{5.16}$$

where $\mathscr{P}_{\mathcal{G}}$ is the projector onto the reproducing kernel Hilbert Space (RKHS) induced by K and $\lambda > 0$ a regularization constant. From a representer theorem [Schölkopf et al., 2001] it follows that the solution to this problem can be expressed as

$$\hat{h}(\boldsymbol{z}) = \sum_{i=1}^{M} \theta_i \psi_i(\boldsymbol{z}) + \sum_{t=1}^{N} \alpha_t K(\boldsymbol{x}_t, \boldsymbol{z}) \tag{5.17}$$

for coefficients $\theta_i, \alpha_t \in \mathbb{R}$. Here the estimation problem (5.16) corresponds to (5.2) in the primal-dual setting. Due to the generalization of Assumption 5.1 to Assumption 5.2 it is however closer to (5.5). Note that the modified assumption imposes conditions on $\boldsymbol{\varphi}$ and $\boldsymbol{\psi}$ which will become clear later on. Another difference can be seen by comparing the predictive models (5.17) and (5.3), notably that the RKHS formulation considers no bias term. This distinction however is cosmetic as the bias term could be included in the parametric basis. The model parameters can be estimated from (5.4) when accounting for the missing bias term by eliminating the last column and the last row from the system. The derivation in a functional setting is slightly more involved and given by Wahba [1990, Eqs. (1.3.16) & (1.3.17)].

5.5.2 Empirical orthogonality in RKHSs

To derive results similar to the ones in the primal-dual setting, Assumption 5.2 on the existence of a direct sum decomposition between the spaces $\mathcal{F}$ and $\mathcal{G}$ is dropped. It is replaced by an explicit construction of a suitable subspace $\mathcal{G}_{eq}$ of $\mathcal{G}$ that allows forming the direct sum between $\mathcal{F}$ and $\mathcal{G}_{eq}$.

The result is an application of a lemma which states that a closed subspace of an RKHS is itself an RKHS.

Lemma 5.5 (Kernel of a Closed Subspace [Berlinet and Thomas-Agnan, 2004, Theorem 11]). *Let $\mathcal{V}$ be a closed subspace of a RKHS $\mathcal{G}$. Then $\mathcal{V}$ is an RKHS and its kernel K' is given by*

$$K'(\boldsymbol{x}, \boldsymbol{y}) = \left[\mathscr{P}_{\mathcal{V}} K(\cdot, \boldsymbol{y})\right](\boldsymbol{x}) \tag{5.18}$$

where $\mathscr{P}_{\mathcal{V}}$ denotes the projection operator onto $\mathcal{V}$.

Deriving a suitable subspace is again based on an orthogonality criterion formulated upon the empirical data. As the orthogonality holds only on the estimation data, the model parts need to be sampled, which is achieved by the sampling operator $\mathscr{S} : \mathcal{L}_2 \to \mathbb{R}^N$, defined entry-wise as $\mathscr{S}h := [h(\boldsymbol{x}_t)]_{t=1}^N$. Then the linear operator $\mathscr{C} : \mathcal{L}_2 \to \mathbb{R}^M$ defined as $\mathscr{C}h = \boldsymbol{\Psi}\mathscr{S}h$ computes the correlation between the parametric basis functions and an arbitrary function in $\mathcal{L}_2$, both sampled on the empirical data. For technical reasons one has to assume that $h \in \mathcal{L}_2$ where $\mathcal{L}_2$ is the space of square integrable functions. Note that it is implicitly assumed that $\mathcal{F}, \mathcal{G} \subset \mathcal{L}_2$ which, for $\mathcal{G}$, is true under mild conditions on the kernel K [De Vito et al., 2004].

Now the original RKHS can be decomposed as

$$\mathcal{G} = \mathrm{range}(\mathscr{C}^*) \oplus \mathrm{null}(\mathscr{C}),$$

where $\mathscr{C}^* : \mathbb{R}^M \to \mathcal{G}$ is an adjoint operator of $\mathscr{C}$ and defined as $\mathscr{C}^*\boldsymbol{\beta} := \boldsymbol{k}(\cdot)^T \boldsymbol{\Psi}^T \boldsymbol{\beta}$, where $\boldsymbol{\beta} \in \mathbb{R}^M$ and $\boldsymbol{k}(\cdot)$ is defined as in Lemma 5.3. Note that range($\mathscr{C}^*$) is a finite dimensional subspace of $\mathcal{G}$ and null($\mathscr{C}$) is a closed subspace. Then by Lemma 5.5 null($\mathscr{C}$) is a RKHS and its kernel function is solely given in terms of g and of $\mathscr{P}_{\text{null}(\mathscr{C})}$, the projector onto the nullspace of $\mathscr{C}$. Therefore $\mathscr{P}_{\text{null}(\mathscr{C})} : \mathcal{G} \to \mathcal{G}$ is given by (see e.g. [Luenberger, 1998, Theorem 1, Chapter 6.9])

$$\mathscr{P}_{\text{null}(\mathscr{C})} g = (\mathscr{I} - \mathscr{C}^* \left(\mathscr{C}\mathscr{C}^*\right)^{-1} \mathscr{C}) g = g - \boldsymbol{k}(\cdot)^T \boldsymbol{\Psi}^T \boldsymbol{\Omega}_{\Psi}^{-1} \boldsymbol{\Psi} (\mathscr{S} g),$$

where $\boldsymbol{\Omega}_{\Psi} := \mathscr{C}\mathscr{C}^* = \boldsymbol{\Psi}\boldsymbol{\Omega}\boldsymbol{\Psi}^T$ is assumed to be a full-rank matrix as in Lemma 5.3 and $\mathscr{I}$ denotes the identity operator. Then by equation (5.18) the reproducing kernel of null($\mathscr{C}$), denoted by K_{eq}, is given by $K_{eq}(\boldsymbol{x}, \boldsymbol{y}) = \left[\mathscr{P}_{\text{null}(\mathscr{C})} K(\cdot, \boldsymbol{y})\right](\boldsymbol{x})$. Finally one can denote null($\mathscr{C}$) as $\mathcal{G}_{eq}$ and has constructed a suitable subspace of the original RKHS such that the following holds.

Proposition 5.6. *For any $f_1 \in \mathcal{F}$ and any $g_1 \in \mathcal{G}_{eq}$ the function $h = f_1 + g_1$ admits a unique decomposition under the assumption that $\boldsymbol{\Psi}$ has full rank.*

Proof. Suppose that there are $f_2 \in \mathcal{F}$ and $g_2 \in \mathcal{G}_{eq}$ with $f_1 \neq f_2$ such that $h = f_2 + g_2$. Note that f_1 and f_2 can be parametrized as $f_1(\boldsymbol{x}) = \boldsymbol{\psi}(\boldsymbol{x})^T \boldsymbol{\theta}_1$ and $f_2(\boldsymbol{x}) = \boldsymbol{\psi}(\boldsymbol{x})^T \boldsymbol{\theta}_2$ respectively. As $f_1 \neq f_2$ also $\boldsymbol{\theta}_1 \neq \boldsymbol{\theta}_2$. By assumption one needs to have $(f_1 + g_1)(\boldsymbol{x}) = (f_2 + g_2)(\boldsymbol{x})$ for all $\boldsymbol{x} \in \mathbb{R}^D$. In particular this has to hold if $\mathscr{C}$ is applied to both sides. By construction one has $\mathscr{C} g_1 = \mathscr{C} g_2 = \mathbf{0}_M$. Therefore $\boldsymbol{\Psi}\mathscr{S} f_1 = \boldsymbol{\Psi}\boldsymbol{\Psi}^T \boldsymbol{\theta}_1 = \boldsymbol{\Psi}\boldsymbol{\Psi}^T \boldsymbol{\theta}_2 = \boldsymbol{\Psi}\mathscr{S} f_2$. As $\boldsymbol{\Psi}$ has full rank it follows $\boldsymbol{\theta}_1 = \boldsymbol{\theta}_2$ which is a contradiction. □

It now follows from Proposition 5.6 that one can write

$$\mathcal{H} := \mathcal{F} \oplus \mathcal{G}_{eq}.$$

Then for any $h_1 = \boldsymbol{\theta}_1^T \boldsymbol{\psi} + g_1$ and $h_2 = \boldsymbol{\theta}_2^T \boldsymbol{\psi} + g_2$ one can define the inner product $\langle h_1, h_2 \rangle := \boldsymbol{\theta}_1^T \boldsymbol{\theta}_2 + \langle g_1, g_2 \rangle_{K_{eq}}$ that turns $\mathcal{G}_{eq}$ and $\mathcal{F}$ into orthogonal complements. The corresponding norm is $\|h\| := \sqrt{\langle h, h \rangle}$. Denoting by $\mathscr{P}_{\mathcal{G}_{eq}}$ the projection onto $\mathcal{G}_{eq}$ by the representer theorem, the solution of (5.16) is given by (5.17) where $\boldsymbol{\theta} \in \mathbb{R}^M$ and $\boldsymbol{\alpha} \in \mathbb{R}^N$ can be found in closed form by solving a system of linear equations similar to (5.4). Since in the current setting one has $\boldsymbol{\Psi}\boldsymbol{\Omega}_{eq}\boldsymbol{\alpha} = \mathbf{0}_M$, similar arguments as in Lemma 5.2 can be used to established that $\boldsymbol{\theta}$ is given by $\boldsymbol{\theta}_{OLS}$. Once the parametric part has been computed, one can also find the nonparametric part starting from the residuals $\boldsymbol{r} = \boldsymbol{y} - \boldsymbol{\Psi}^T \boldsymbol{\theta}_{OLS}$ and solve $(\boldsymbol{\Omega}_{eq} + \lambda \boldsymbol{I})\boldsymbol{\alpha} = \boldsymbol{r}$. The final model (for out-of-sample prediction) is given by (5.17) in terms of $\boldsymbol{\alpha}$, $\boldsymbol{\theta}$ and the equivalent kernel function K_{eq} that is given in terms of the original kernel K, the training data $\{\boldsymbol{x}_t\}_{t=1}^N$ and the parametric basis functions $\{\psi_i\}_{i=1}^N$.

5.6 Experiments

To validate the modified model structure it is compared to existing ones on a simulation example as well as a real life data set. For simplicity and interpretability the considered parametric model class for both examples is a linear ARX model. The different model structures that are evaluated are

PAR purely parametric models

SVM, LS-SVM purely kernel based black box models,

RES (LS-SVM) parametric model with kernel based model for residuals (cf. Remark 5.2),

PL (LS-SVM) partially linear model without orthogonality constraints (cf. Eq. (5.5)) and

OPL (LS-SVM), OPL (SVM) partially linear models with orthogonality constraints (cf. Eqs. (5.8) and (5.14)).

5.6.1 Experimental setup

For all experiments the parametric part is taken to be linear, i.e. $\boldsymbol{\psi}(\boldsymbol{x}) = \boldsymbol{x}$, while the nonparametric contribution is modeled with a RBF kernel. The tuning parameters of the nonlinear models (kernel parameters and regularization constant) are selected according to one-step ahead prediction performance on a validation set using grid search.

Remark 5.3 (Necessity to employ orthogonality constraints). It has been shown that if the Gaussian kernel is used in combination with a linear model, the corresponding spaces allow unique decompositions [Ha Quang et al., 2009]. Therefore the experimental conditions satisfy the conditions for classical partially linear models as given in Subsection 5.5.1. Although given in an RKHS setting, it also applies to the primal-dual framework, i.e. a partitioning of the regression variable is not needed. Therefore the addition of the orthogonality constraint is, at least theoretically, not necessary and the PL (LS-SVM) model structure satisfies its assumptions. However the examples indicates that in case of small but realistic sample sizes (> 1000 samples) an improvement by using the constraint can be observed. This is due to the finite subsample size as outlined in Subsection 5.2.1.

In addition to the root mean squared error (RMSE) on independent data the parametric estimates are compared. In case of the simulation data this is done with a metric for ARMA systems while for the real world data, the estimated frequency response functions are plotted.

Remark 5.4 (Metric for ARMA systems [Martin, 2000]). This metric compares the spectra of linear systems and, as such, is independent of their parametrization. Elaborate arguments why this metric is better in terms of system-theoretic properties than e.g. comparing AR coefficients are given in the reference. For two stable AR systems S_A, S_B with poles p_i^A, p_j^B, $i = 1, \dots, P_A, j = 1, \dots, P_B$ the metric can be computed as

$$d(S_A, S_B) = \ln\left(\frac{\prod_{i=1}^{P_A} \prod_{j=1}^{P_B} |1 - p_i^A p_j^{B^*}|^2}{\prod_{i,j=1}^{P_A} (1 - p_i^A p_j^{A^*}) \prod_{i,j=1}^{P_B} (1 - p_i^B p_j^{B^*})} \right)$$

where x^* denotes the complex conjugate of x. The generalization to stable ARX systems is given by Martin [2000]. A drawback of this metric is that it applies equal weight to all frequencies, which might not be intended in some situations.

5.6.2 Toy example

To illustrate the behavior of the different model classes, consider a simple static one dimensional toy example $y_t = x_t + x_t^2 + \sin(x_t) + \cos(x_t) + e_t$. Both input and noise are normally distributed with mean zero. The standard deviation for the inputs is 1 and for the noise 0.1 respectively. The size of the estimation set for each model is 100 samples. The validation set used to select the tuning parameters is of size 500 and so is the independent test set for evaluating the prediction performance. To obtain insight into the stability of the estimates 1000 realizations of the data are tested. For this example the parametric model is defined as $\psi(x) = [1, x, x^2]^T$ and the nonparametric model is based on a RBF kernel. Several properties of the model structures are compared in Table 5.1.

As motivated in the introduction, the prediction performance of the partially linear models should be better than a purely parametric and a purely nonparametric model. For this example PL slightly outperforms OPL and

RES but all satisfy the objective. While the good performance of RES is most likely due to the rather simple structure of the problem, the advantage of PL is due to the higher flexibility of this model structure. An indication for this flexibility is also the large fraction of the estimate contributed by the nonparametric part. Instead of 11% for OPL and RES, it is is on average at 76% for PL with a very large variability compared to the others.

The second objective is to obtain an estimate for the parametric part that is as good as the OLS one. Analyzing the mean is of limited usefulness in this example as $x + x^2$ and the nonparametric part $\sin(x) + \cos(x)$ are highly correlated. Therefore the comparison is focused on the variance. As can be seen in Table 5.1, it is many times larger for PL than for OPL. The histograms depicted in Figure 5.2 for the parameter values show that, for this example, there seem to be at least two distinct decompositions that result in a similar predictive performance in case of PL.

This observation is consistent with many other simulations, showing a high variability for PL models. Only very few examples exhibited a smaller variability for PL than for OPL. However, in these cases both models resulted in stable estimates.

5.6.3 Mass-Spring-Damper system

To further evaluate the proposed method, it is tested on a simple mass spring damper system with two masses and springs with cubic nonlinearities as depicted in Figure 5.3. The system is described as

$$
\begin{aligned}
m_1\ddot{s}_1 &= -L_1 f_1(s_1) + L_2 f_2(s_2 - s_1) - C_1\dot{s}_1 + C_2(\dot{s}_2 - \dot{s}_1),\\
m_2\ddot{s}_2 &= -L_2 f_2(s_2 - s_1) - C_2(\dot{s}_2 - \dot{s}_1) + F_u,
\end{aligned}
$$

$y = s_2$, $f_1(x) = x + 0.02x^3$ and $f_2(x) = x + 0.005x^3$. The states s_1 and s_2 are the displacement of mass 1 and mass 2 from rest, respectively. The constants are chosen as $m_1 = 5\,\mathrm{kg}$, $m_2 = 0.1\,\mathrm{kg}$, $L_1 = 1.5\,\mathrm{N/m}$, $L_2 = 0.5\,\mathrm{N/m}$, $C_1 = 0.3\,\mathrm{Ns/m}$ and $C_2 = 0.05\,\mathrm{Ns/m}$. The output y is sampled with 4 Hz and the excitation force F_u is given by a Gaussian white noise process u_t. The equations are simulated using MATLAB's `ode45`, a zero order hold stage for the input signal and zero initial conditions. For each realization of the input signal and every amplitude value, 4000 samples are generated. The first 400 samples of each data set are discarded to exclude transient behavior. To compare measurements with different amounts of nonlinearity, excitation amplitudes of $\{0.1, 0.3, 0.5, 0.75, 1, 1.25, 1.5, 2\}$ are tested. This accounts to different fractions

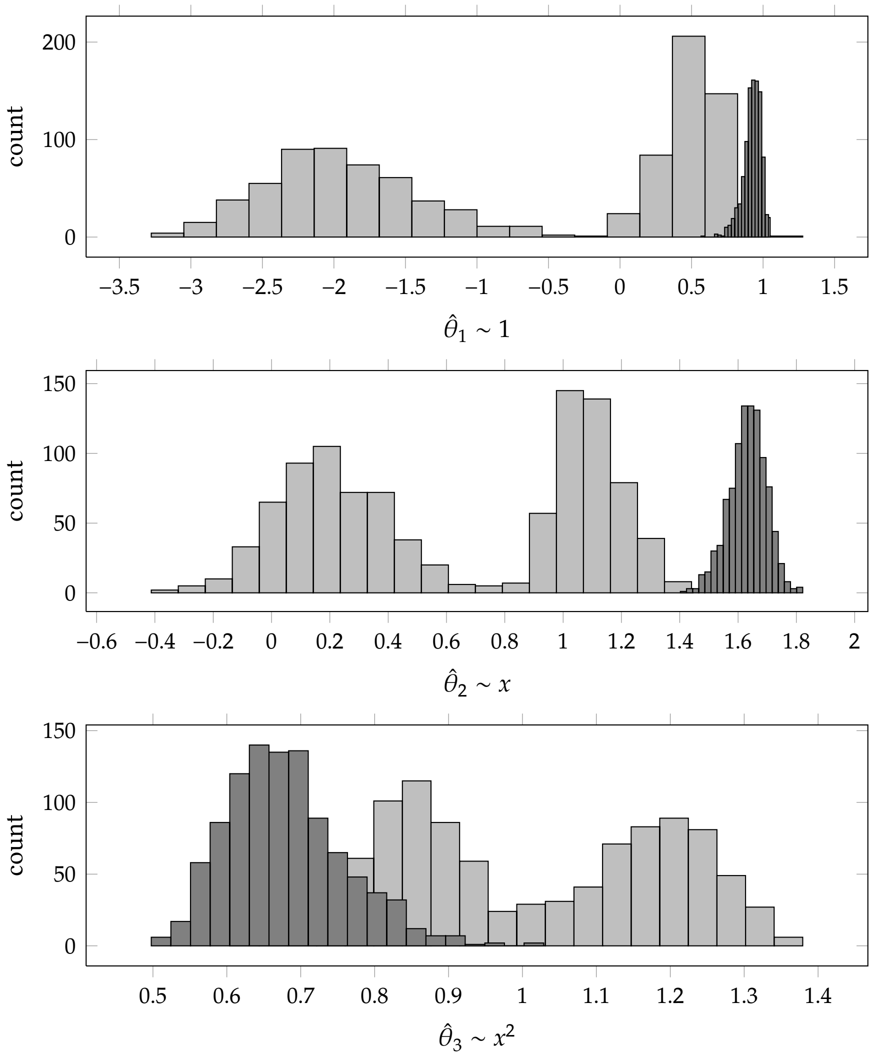

Figure 5.2: Histograms of parameter estimates for the toy example over 1000 realizations of the data. The light gray histograms are for PL and the dark gray ones are for OPL respectively.

Table 5.1: Comparison of model structures for the toy example. All reported values are mean values averaged over 1000 realizations of the data with the standard deviation given between parenthesis. Predictive performance (perf) is given as relative RMSE: RMSE($\hat{y}_t - y_t$)/RMSE(y_t). "% nonpar" is the percentage of the model output contributed by the nonparametric part (RMSE($\hat{y}_{\text{nonpar},t}$)/RMSE($\hat{y}_t$)). The remaining columns are the parameter estimates.

MODEL	PERF	% NONPAR	$\hat{\theta}_1$ (~ 1)	$\hat{\theta}_2$ ($\sim x$)	$\hat{\theta}_3$ ($\sim x^2$)
PAR	0.12	0	0.92	1.63	0.68
	(0.02)	(0)	(0.06)	(0.06)	(0.08)
LS-SVM	0.07	100	–	–	–
	(0.03)	(0)			
RES	0.05	11	0.92	1.63	0.68
	(0.01)	(2)	(0.06)	(0.06)	(0.08)
PL	0.04	74	0.77	0.63	1.02
	(0.18)	(46)	(1.29)	(0.48)	(0.18)
OPL (LS-SVM)	0.05	11	0.92	1.63	0.68
	(0.01)	(2)	(0.06)	(0.06)	(0.08)

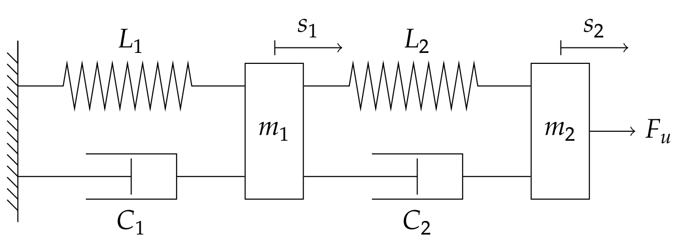

Figure 5.3: Mass-spring-damper system with nonlinear springs.

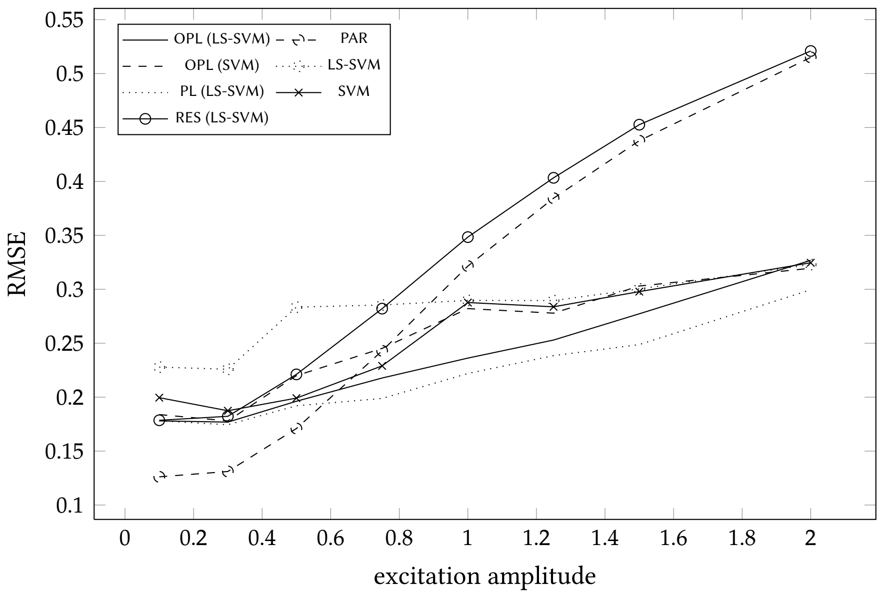

Figure 5.4: Prediction performance of different model structures for the mass spring damper system on an independent test set as a function of excitation amplitude. The root mean squared error for simulated outputs averaged over ten input realizations is being reported.

of the system not being captured by the parametric (linear) model. For each amplitude the estimation is performed for ten realizations of the input signal. All data sets are split in three parts of equal size for model estimation, model validation and test.

Figure 5.4 depicts the simulation performance for different model structures. The model orders are as follows $p = q = 13$ for PAR and $p = q = 10$ for all other models, where the input variable $\boldsymbol{x}_t$ has NARX structure with $\boldsymbol{x}_t = [y_{t-1}, \dots, y_{t-p}, u_{t-1}, \dots, u_{t-q}]^T$. For a fair comparison the input sequence u_t as well as the output sequence y_t are normalized to unit variance for each amplitude. As it can be expected the linear model is best for small amplitudes but degrades quickly as the amplitude increases. The nonlinear black box models LS-SVM and SVM are the best models for “large” amplitudes. Modeling the residuals is clearly inferior to all other approaches as it is never better than the linear model. Finally the partially linear structures give improved prediction performance for a number of amplitudes. Imposing orthogonality constraints yields slightly worse predictive performance than using a partially

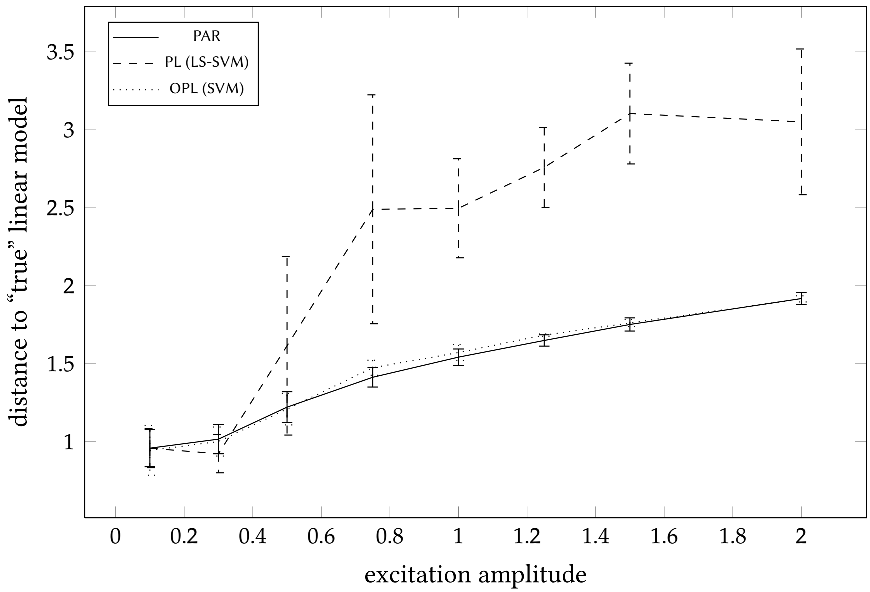

Figure 5.5: Distance of the estimated parametric submodel to the "true" linear model for the mass-spring damper system for several model structures as a function of the excitation amplitude. The models PAR, OPL (LS-SVM) and RES (LS-SVM) have identical distances. Error bars indicate the standard deviation for ten realizations of the input signal. The comparison is based on the ARMA metric proposed in [Martin, 2000].

linear structure without constraints.

Setting $f_1(x) = f_2(x) = x$, i.e. linearizing the springs, a "true" linear system can be obtained. Using MATLAB's `c2d` a discrete time transfer function is computed from the continuous time system described above. The "true" linear system is of order four while the identified models are of orders thirteen and ten. Therefore the ARMA metric described earlier is used to compare the estimated with the "true" model. The result is shown in Figure 5.5. It can be seen that for small amplitudes the partially linear model is close to the linear estimate. Yet for larger amplitudes it quickly diverges away from the linear estimate and not only has a larger distance from the "true" linear model but the estimates also have a larger variance. While both partially linear structures move away from the "true" linear model as the excitation amplitude increases, the models with orthogonality constraints degrade more gracefully than those without.

In conclusion, both partial linear structures yield better predictive performance than a parametric or nonparametric model alone. The unique advantage of using orthogonality constraints is that the estimate for parametric part has a much smaller variance and also is much closer to the “true” underlying model as seen in Figure 5.5.

5.6.4 Wiener-Hammerstein benchmark data

To test on a real life data set, the data of the Wiener-Hammerstein benchmark [Schoukens et al., 2009] at SYSID2009 is considered. As it serves as a test problem, only a small subset of the 188,000 measured samples is taken. For model estimation and validation 2,000 samples are used each. The prediction performance is computed on 5,000 independent samples.

The results are shown in Table 5.2, the reported values are for outputs normalized to unit variance. It can be seen that roughly the same behavior as with the simulated data is present. The partially linear models improve over the purely linear and purely nonlinear models. Using orthogonality constraints results in a slight degradation of prediction performance as observed in the previous section. In tests with small estimation sample sizes the prediction performance of OPL (LS-SVM) is better than PL (LS-SVM) up to a sample size of about 200. Figure 5.6 shows transfer functions derived from the parametric model parts. The estimation is done for 20 subsamples of size 300. One can observe that the OPL structures are much closer to the reference spectrum given by the Best Linear Approximation than the PL model. Also its variability is much smaller.

5.7 Conclusions

This chapter reviewed existing classical partially linear models in Section 5.1 in a primal-dual setting and in Subsection 5.5.1 in a RKHS setting respectively. The analysis of the existing model structures in Section 5.2 led to the introduction of a novel orthogonality constraint. This constraint ensures that, if the model classes of the parametric model part and the kernel model overlap, the kernel model is constrained such that the parametric model is not affected. This is also ensured for finite sample sizes for which otherwise even for disjoint model classes, the parametric model part cannot be estimated properly. This was illustrated on numerical examples. Both the classical as well as the modified partially linear model class have a better predictive performance

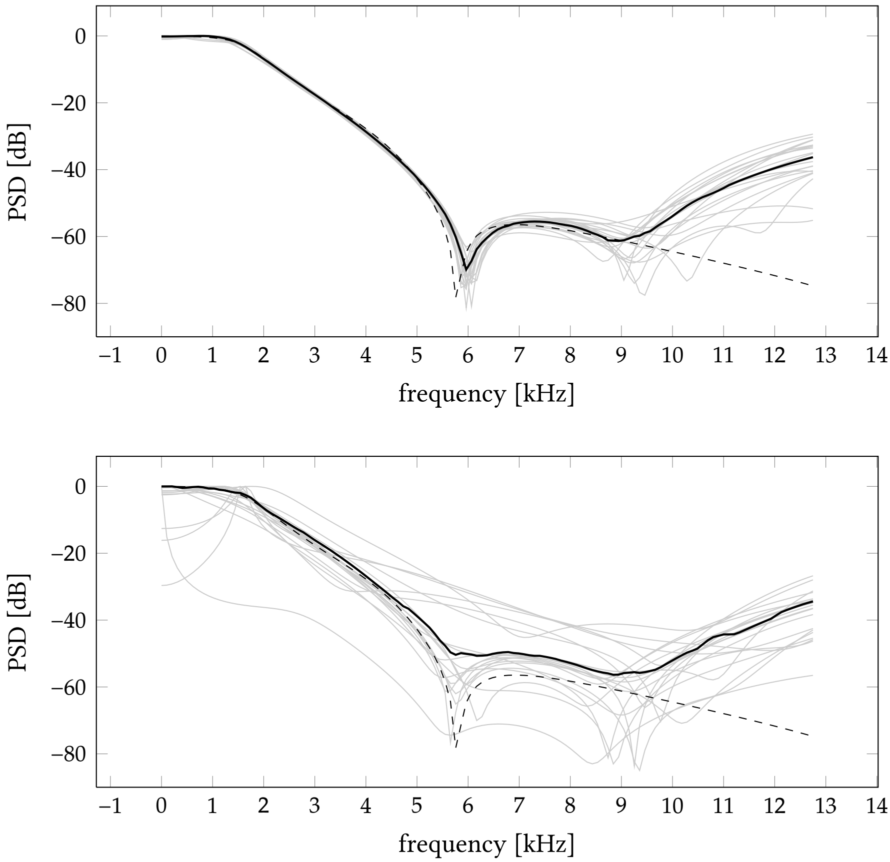

Figure 5.6: Estimated transfer functions for the proposed model with orthogonality constraints (OPL (LS-SVM), top) and classical partially linear models (PL (LS-SVM), bottom). The estimates are obtained for 20 different training samples of size of 300 taken out of the Wiener-Hammerstein benchmark dataset. Each light gray line corresponds to one realization, the thick black line indicates the median value while the dashed line gives the Best Linear Approximation [Schoukens et al., 2003] estimated on the full dataset as reference. The excitation signal was subject to low-pass filtering with a cut-off frequency of 10 kHz.

Table 5.2: Performance of different model structures on a subsample of the Wiener-Hammerstein benchmark dataset. The performance is reported as root mean square error for simulation values.

MODEL	TRAINING DATA	VALIDATION DATA	TESTING DATA
PAR	0.164	0.168	0.230
LS-SVM	0.067	0.135	0.183
SVM	0.053	0.079	0.139
RES (LS-SVM)	0.162	0.155	0.222
PL (LS-SVM)	0.037	0.068	0.133
OPL (LS-SVM)	0.091	0.094	0.166
OPL (SVM)	0.051	0.071	0.140

compared to purely parametric and to purely kernel based models. However the estimates for the parametric part exhibit large variations and deviate significantly from the OLS estimate for the classical partial linear models. In contrast to this, the estimates of the proposed partial linear model structures have a small variability and show a good agreement with existing estimates. In case the model with the best predictive performance is sought, then the classical partially linear model structure is best. However in this case the user has be aware of the effect this can have on the parametric estimate and must be extremely cautious when attaching it to any kind of interpretation.

Additionally the results were considered in the context of RKHSs which offered insights on how the new model class is related to the old one. For the primal-dual formulation an alternative estimation scheme was derived that allows estimating both model parts separately from each other. For better integration with existing approaches the orthogonality constraint has been embedded in an equivalent kernel function. Finally it has been demonstrated that the extension to some alternative loss functions is straightforward on the example of SVMs.

6 Modeling systems with multiple outputs

The motivation for the research presented in this chapter is twofold. On the one hand, it picks up a recently popular way of regularization, specifically nuclear norm regularization, and applies it to kernel based modeling in a primal-dual framework. On the other hand there is an interest from the application point of view, where it is beneficial when knowledge about one system, or part of one system, can be used to improve the estimate of a similar system, or another part of the same system. The next paragraphs should give a rough overview of this chapter, starting with motivation as well as means. A larger part will discuss the challenges identified so far and provide some insight to what extent they have been solved. In general, the results presented in this chapter are only preliminary and in many areas incomplete. Therefore the intention is mostly to report the current status of this line of research and provide some personal opinion on which areas will need most attention and what would be the benefit if progress can be made.

6.1 Introduction

6.1.1 Possible applications

The application brought up in the first paragraph is to share information about similar systems. To illustrate the intention and its potential use, three hypothetical examples will be given.

1. Consider a state space system and the objective is to predict the future state trajectory from past data. For the sake of simplicity, assume that the system has only one input and that the system is observable from each of its multiple observed outputs. Then it is obvious that one has access to several sets of measurements that allow the reconstruction of the system dynamics. In the linear case, the measurements will be different linear combinations of the system state, probably subject to some measurement noise. For this linear case this knowledge, that the different measured signals relate to the same system, can be readily exploited by, for example, subspace identification. For the quite general setting of this thesis, nonlinear systems with nonparametric kernel based models, such methods are not generally known.

2. Another example where one has access to multiple sets of data that relate to basically the same dynamical system can be found in other chapters of this thesis, namely those making use of overparametrization to overcome nonconvexity. The most straightforward example is that of a Hammerstein system. Consider a static nonlinearity $f(\cdot)$ driving a linear system that can be modeled by a finite impulse response system $H(z) = \sum_{k=0}^{q} b_k z^{-k}$. Now modeling the output y_t as a function of the input u_t, yields

$$y_t = \sum_{k=0}^{q} b_k f(u_{t-k}), \tag{6.1}$$

where noise is neglected for the sake of simplicity. It is straightforward to realize that this accounts to measuring a linear combination of the common nonlinear system f.

3. As last motivating example, consider the following real world scenario. To distribute power in current electricity networks it is of paramount importance to have accurate knowledge of power demands and sources. Only then it can be distributed correctly in the network. To facilitate this and make it more effective, there is also interest to forecast demands as well as available power at a given time in the future. This is even more important nowadays as power is traded on international markets and – especially renewable – energy sources are highly volatile over time. Looking at the demand side, one usually has measurements at substations distributed around the transmission network. Concentrating on a very simple situation one has substations located in (i)

purely residential areas, where power consumption is concentrated on mornings, evenings and weekends, (i) purely office like environments that consume most energy between 9-to-5 and only during workdays and finally (iii) purely industrial areas, which have very regular profiles and may for example operate on two shifts. While substations in each of these hypothetical areas will have their specific profiles, there are chances that a residential profile in village A is very similar to a residential profile in village B. In fact it has been demonstrated [Alzate et al., 2009] that for the Belgian power distribution network one can identify a number of distinct profiles with which almost all of the substations can be described. Again there is the situation where one has access to data for which a large number of measurements can possibly be explained by a much smaller number of dominant behaviors.

6.1.2 Technical approach and theoretic setting

From a purely optimization point of view, basically all of the above problems can be transformed into a formulation where the objective is to minimize the number of independent components. Assuming that one has a suitable and rich enough basis to describe those problems, one can focus on linear independence. One of the key ideas in kernel based modeling, especially in a primal-dual setting, is that an inherently nonlinear problem can be solved using linear means by projection into a high dimensional space. Under the assumption that this is also feasible once more than one system is considered at the same time, one can focus on minimizing a number of linear independent components. The advantage of the primal-dual framework is that models are explicitly parametrized in the primal. Hence, a problem which aims at modeling M systems will usually be described by M sets of parameters, which will conveniently be denoted by w_m, $m = 1, \dots, M$ for the remainder of this subsection. Now the number of independent models in the set $\{w_m\}_{m=1}^M$ is equal to rank(W) where $W = [w_1, \dots, w_M]$ is the concatenation of the individual model parameters.

Summarizing the above, it turns out that the problem can be treated as an optimization problem involving the rank of a suitably defined matrix of parameters. Rank based optimization is a highly nonconvex problem, but due to its many applications has gained significant interest in the last decade. On the one side there are approaches that tackle the complexity directly on the optimization side with techniques like optimization on manifolds [Absil et al., 2008]. On the other side there is a growing community that looks into a specific convex relaxation of the rank problem. In the context of system identification the nuclear, or trace norm, was proposed by Fazel as the convex envelope of the rank function [Fazel et al., 2001; Fazel, 2002]. Fazel also gave a transformation into SDP form suitable for general purpose solvers. This has been picked up in different areas, the most prominent probably being compressed sensing. In this context special purpose algorithms have been derived, mostly based on recent advances in first-order optimization. These improved algorithms are capable of handling much larger problems as the memory demand is much smaller and it is often possible to exploit problem specific structure to speed up computations. On the downside, gradient based algorithms need fairly results.

In the context of learning functional relations from data the work of Argyriou et al. [2008, 2009] derives many fundamental results.

6.1.3 General setting and identified difficulties

In the context of this chapter, the goal is to find joint kernel based models for groups of models, where it is assumed that all systems can be described as a linear combination of a few prototype models. Two additional technical design choices are that the approach should rely on (i) convex optimization as well as (ii) primal-dual model descriptions. As will be seen in the next section, the problem formulation in this setting is rather straightforward once the nuclear norm is chosen. The encountered challenges in a primal-dual framework are threefold. The first challenge in kernel based models is that the primal formulation is only implicitly defined and possibly infinite dimensional. Therefore for numerical solution, the problem has to be transformed into an explicitly defined, finite dimensional problem. In the conventional setting outlined in Chapter 4 the derivation is straightforward mainly due to the presence of a simple quadratic regularization term. However, in this case the process involves several new approaches. The second problem is that once the problem has been transformed into a finite dimensional problem, the relation to the original model equations is lost. Although a numerical solution to the optimization problem can be obtained, there is no access to the model parameters without linking the dual solution variables with the primal ones used to describe the model. The last problem is then of more practical nature and concerns numerically efficient means to solve the optimization problem and establish the aforementioned link between primal and dual parameters.

The first problem of dualizing the optimization problem, transforming it into a finite dimensional and explicitly defined problem is solved in Section 6.4.

The derivation is different from the traditional one in Chapter 4 but quite general and also employed in similar form in Chapters 9 and 10.

The problem of linking the dual solution to the primal problem specification is particular to the setting in this thesis. In kernel based learning and system identification, optimization is a tool used to find good parameters for a specific model structure. In the kernel based setting the model is parametrized as a primal optimization problem and it is of crucial importance that values for the chosen parameters are obtained, as otherwise the model cannot be evaluated. In Section 6.5 such a relation is established. However the derived connection has several limitations: i) in general another optimization problem has to solved to establish a link between primal variables and dual solution, ii) there are multiple ways to compute the relation, iii) none of these approaches is elegant and maybe most importantly iv) no intuition or insight into the problem is gained in the process. Therefore it is my personal opinion that another link can still be found that not only connects the two representations but also gains additional information in the process.

The third challenge in the methodology introduced in this chapter is of more practical matter. The necessary optimization problems are very expensive to solve. The current situation for suitable algorithms offers two compromise solutions. On the one hand there is the possibility to treat the resulting problems as SDPs and use general purpose solvers for such problems. On the positive side this has the advantage that the implementation is straightforward and requires only little effort. Also the results usually have a very high numerical precision which is important for linking primal and dual solutions as discussed in the previous paragraph. On the downside the available general purpose solvers are slow and only capable of solving problems with relatively few optimization variables.

The latter is a problem especially in the context of this chapter because collections of models are being optimized in a joint approach. Therefore the number of free parameters is the number of parameters of a single model times the number of models. As a consequence, the computational demand that would already be substantial for a single model is multiplied by the number of models considered. The situation for kernel based models is even worse. One advantage of classical kernel based models is that the number of variables scales with the number of available data and not with the number of model parameters, which can go to infinity for certain choices of the kernel function. In the present context this has a severe consequence, whereas a single system might be associated with N data samples, M systems will usually have N data samples each, resulting in a total of $M \cdot N$ data samples.

Now recall that a kernel matrix is square in the number of data. Hence the available memory of the employed computing hardware is limiting the number of models or data that can be analyzed in practice. For better intuition consider the availability of 1 GB for storing the kernel matrix alone, not exploiting symmetry and assuming double precision representation of the matrix elements, then the product $M \cdot N$ can be 11,000. While this may seem plenty, consider that for interesting situations the number of individual models M can easily range between 10 and 100. This limits the processable data per model quite drastically.

As an alternative to general purpose SDP solvers, one can consider first-order optimization techniques. These recently gained much interest and progressed quickly due to the demands in the compressed sensing community. The advantages of these algorithms are that their complexity per iteration is low, they can handle large problems, allow structure in the problem to be exploited, can be warm-started and are relatively straightforward to implement. Apart from these attractive characteristics they also feature several important disadvantages. No general purpose solvers exist that could be employed, but the optimization algorithm needs to be implemented by the user, the number of iterations until convergence is usually very high and the absolute numerical performance is in general worse than that of general purpose interior-point algorithms. Especially this last point makes their application very difficult for the material presented in this chapter as the link between primal and dual solutions is very sensitive to the quality of the results.

In conclusion, for the third problem there are two possible choices of algorithms that both have advantages and disadvantages relevant to the current setting. Which one to choose is basically an open problem as both choices do not give overall satisfactory results at the moment. Therefore either progress has to be made on an algorithmic level or the algorithmic challenges have to be solved by more clever approaches in the modeling. From a personal point of view, most potential is in making progress towards solving the second challenge identified in one of the paragraphs further above. Depending on the nature of the progress the need for extremely accurate solutions could be relaxed so that the relatively flexible and fast first-order algorithms can be employed.

6.1.4 Structure of chapter

The chapter is structured as follows. The following section will formally state the estimation problem. First in its conventional form directly derived from the basic LS-SVM model and finally proposing an improved version. Due to the more complex form of the optimization problem some of its properties are derived in Section 6.3. The transformation of the parametric primal model to the kernel based dual model is carried out in Section 6.4. In contrast to the straightforward derivation of the dual model representation with LS-SVMs, the more involved procedure necessary in this case is presented in Section 6.5. Following that, the basic formulation derived in the previous sections is extended to the case where each output variable can be modeled by different input variables. This derivation includes the possibility to account for different amounts of measurement data for each target variable, as for example with missing data. Extensions to other convex loss functions besides the squared loss function, like the ε-insensitive or the Huber loss [Huber and Ronchetti, 2009] can be derived in a straightforward fashion. However, this task is left as an exercise for the interested reader. The last extension discussed in Section 6.6 are overparametrized models, which will play an important role in the next chapters. The solution of the convex optimization problems is discussed in Section 6.7. This section considers both, general purpose semidefinite programming (SDP) solvers as well as approaches based on first order algorithms, especially gradient projection algorithms. For validation a numerical example is given in Section 6.8, before the chapter is concluded in the last section.

6.2 Formal problem formulation and motivation

The goal of this chapter is to jointly model multivariate time-series. These time-series can be generated by a single system with multiple measured output variables, multiple related systems or some other process. In this general setting one is given measurement data $\{(\boldsymbol{x}_t, \boldsymbol{y}_t)\}_{t=1}^N$ with regression variables $\boldsymbol{x}_t \in \mathbb{R}^D$ and target variables $\boldsymbol{y}_t \in \mathbb{R}^M$.

6.2.1 Choice of model structure

If one denotes the i-th target variable with a superscript (i) then a simple parametric model is given by

$$\hat{y}_t^{(i)} = f_i(\boldsymbol{x}_t) = \boldsymbol{w}_i^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b_i \tag{6.2}$$

where the components of $\boldsymbol{\varphi}(\cdot) : \mathbb{R}^D \to \mathbb{R}^{n_h}$ form a set of n_h linearly independent basis functions and $\boldsymbol{w}_i \in \mathbb{R}^{n_h}$ and $b_i \in \mathbb{R}$ are the model parameters for the

i-th target variable $y_t^{(i)} = (\boldsymbol{y}_t)_i$. In this case f_i is the nonlinear relation between the input variables and a single output variable.

The relation in (6.2) is a convenient one, but often will not map well to a system identification setting. In a typical system identification problem, the input variable $\boldsymbol{x}_t$ will be of ARX structure, i.e. $\boldsymbol{x}_t = [y_{t-1}, \dots, y_{t-p}, u_t, \dots, u_{t-q}]^T$. This brings forward a complication; in the situation analyzed here, there is more than one output variable y_t and possibly different relevant input variables u_t. Therefore two approaches can be considered:

1. One can define a joint regression vector $\boldsymbol{x}_t$ aggregating the information of input and output variables, like $\boldsymbol{x}_t = [y_t^{(1)}, \dots, y_{t-p}^{(1)}, \dots, y_t^{(M)}, \dots, y_{t-p}^{(M)}]^T$ where for simplicity exogenous variables have been neglected. This has the advantage that it fits the framework given by (6.2). However the dimensionality of $\boldsymbol{x}_t$ can get very large and a lot of irrelevant information is provided to the individual models.

2. The second option is to define a regression vector per output variable. In this setting one would define $\boldsymbol{x}_t^{(i)} = [y_{t-1}^{(i)}, \dots, y_{t-p}^{(i)}]$, again neglecting possible exogenous variables. The obvious advantage of this strategy is that the dimensionality of $\boldsymbol{x}_t$ is identical to the univariate modeling case. The drawback of this choice is that the model (6.2) has to be modified to

$$\hat{y}_t^{(i)} = f_i(\boldsymbol{x}_t) = \boldsymbol{w}_i^T \boldsymbol{\varphi}(\boldsymbol{x}_t^{(i)}) + b_i. \tag{6.3}$$

   This however will increase the numerical complexity of the problem as was already briefly mentioned in the introduction and will become evident further along in the text.

To keep the presentation as straightforward as possible, the initial derivation will be carried out for the first possibility with a unique regression vector $\boldsymbol{x}_t$. In Subsection 6.6.1 the generalization to different regression vectors per target variable is briefly presented.

6.2.2 Conventional estimation problem

The model (6.2) is chosen such that it mimics the primal formulation of LS-SVMs as introduced in Chapter 4. Therefore a basic estimation scheme would formulate M independent estimation problems such as

$$\begin{aligned}
&\min_{w_i, b_i, e_t^{(i)}} \quad \frac{1}{2}\eta w_i^T w_i + \frac{1}{2}\sum_{n=1}^{N}(e_t^{(i)})^2 \\
&\text{subject to} \\
&\quad y_t^{(i)} = w_i^T \varphi(x_t) + b_i + e_t^{(i)}, \quad t = 1, \dots, N,
\end{aligned} \tag{6.4}$$

for $i = 1, \dots, M$. The first potential to couple the estimation problem is through the model residuals $e_t^{(i)}$. Therefore let $\boldsymbol{W} = [w_1, \dots, w_M]$, $\boldsymbol{b} = [b_1, \dots, b_M]^T$ and $\boldsymbol{e}_t = [e_t^{(1)}, \dots, e_t^{(M)}]^T$, then the previous estimation problems can be reformulated as

$$\begin{aligned}
&\min_{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{e}_t} \quad \frac{1}{2}\eta \|\boldsymbol{W}\|_F^2 + \frac{1}{2}\sum_{t=1}^{N} \boldsymbol{e}_t^T \boldsymbol{T} \boldsymbol{e}_t \\
&\text{subject to} \\
&\quad \boldsymbol{y}_t = \boldsymbol{W}^T \varphi(x_t) + \boldsymbol{b} + \boldsymbol{e}_t, \qquad t = 1, \dots, N.
\end{aligned} \tag{6.5}$$

The matrix $\boldsymbol{T} \in \mathbb{R}^{M \times M}$ is a positive definite weighting matrix. One possible use case of this weighting is to establish different regularization values η for each one of the components, which corresponds to a diagonal weighting. For more complex cases with nonzero elements off the main diagonal, problem (6.5) cannot be decoupled to (6.4) anymore. It is straightforward to see that $\|\boldsymbol{W}\|_F^2 = \sum_{i=1}^{M} w_i^T w_i$ as well as that the constraint of (6.5) is a concatenation of the constraints of (6.4) for $i = 1, \dots, M$.

6.2.3 Improved estimation problem

In the introduction it has been motivated that in many practical problems the models for the M target variables will not be independent. In that case the matrix $\boldsymbol{W}$ will be rank deficient. Especially for a large number of variables one can assume that $\text{rank}(\boldsymbol{W})$ should be small. This belief or intuition can be translated into a mathematically viable optimization problem by employing the nuclear norm introduced in Section 3.3.3.

Assume that the parameter matrix $\boldsymbol{W}$ is generated by L independent components such that it can be written as $\boldsymbol{W} = \sum_{l=1}^{L} v_l u_l^T$, with $v_l^T v_k = u_l^T u_k = 0$ for $l \neq k$ and $u_l^T u_l = 1$ for $l = 1, \dots, L$. Then the nuclear norm of $\boldsymbol{W}$ can be expressed in terms of its independent components as $\|\boldsymbol{W}\|_* = \sum_{l=1}^{L} \|v_l\|$. Considering this as a regularization term allows for several interpretations:

1. The regularization applied to each component individually $\|\boldsymbol{v}_l\|$ is very similar to that of an individual model in (6.2) $\boldsymbol{w}_i^T\boldsymbol{w}_i$. The difference between the ℓ_2-norm on the one hand and the squared ℓ_2-norm on the hand is negligible as, for an individual model, one can be transformed into the other.

2. The sum over the individual components corresponds to an ℓ_1-norm over the components. This means that the optimization algorithm will try to find a solution where only a few components have a norm different from zero.

This motivates the change of the regularization term in (6.5) to the nuclear norm. This ensures that the optimal solution $\boldsymbol{W}^*$ for

$$\min_{\boldsymbol{W},\boldsymbol{b},\boldsymbol{e}_t} \quad \eta\|\boldsymbol{W}\|_* + \frac{1}{2}\sum_{t=1}^{N} \boldsymbol{e}_t^T \boldsymbol{T}\boldsymbol{e}_t \tag{6.6}$$

subject to

$$\boldsymbol{y}_t = \boldsymbol{W}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + \boldsymbol{b} + \boldsymbol{e}_t, \qquad t = 1,\dots,N,$$

will consist of only few dominant components.

6.3 Properties of parametric estimation problem

Most other chapters of this thesis consider problems with purely quadratic loss functions and regularizations. These have the advantage that they are very well understood. At the same time they have some convenient properties, such as uniqueness and strong duality being straightforward to guarantee. For the problem defined by (6.6) these can also be shown but require a little more argument. Additionally this section considers the choice of the regularization constant η. In ℓ_1-related optimization it is typical that starting at a specific value of the regularization constant, the solution will remain constant.

6.3.1 Uniqueness of the solution

To simplify the analysis note that the estimation of $\boldsymbol{b}$ in (6.5) and (6.6) is not subject to regularization. Therefore it coincides with estimating the mean of the data. Hence, it is possible to assume, without loss of generality, that the data is zero mean and concentrate on solving the problem without $\boldsymbol{b}$.

The least squares problem in (6.5) always has a unique solution as its objective is a strongly convex function [Boyd and Vandenberghe, 2004]. This follows from $\|W\|_F^2$ being strongly convex and the fact that the sum of a strongly convex function with a convex function is again strongly convex. For the nuclear norm penalty in (6.6) the uniqueness of W is less obvious. Therefore consider the following lemma.

Lemma 6.1. *For $\eta > 0$ the solution of* (6.6) *is unique in W, b and e_t.*

Proof. It has already been argued that b is the average of the data and as such it is unique. Considering the residuals $e_t = y_t - W^T\varphi(x_t)$ of a reduced problem with zero mean data, a sufficient condition for their uniqueness is the uniqueness of W. Then eliminating the residuals e_t yields the new optimization problem

$$\min_{W} \frac{1}{2}\|(\Phi^T W - Y^T)T^{\frac{1}{2}}\|_F^2 + \eta\|W\|_*. \tag{6.7}$$

A sufficient condition for the lemma to hold is that the solution to this problem is unique. To show this, the variable W will be decomposed into parts in the range and the nullspace of Φ and Φ^T respectively. In a first step strong convexity for the contribution in the range will be proved. The second step then shows that the part in the nullspace is equal to zero.

An orthogonal basis for Φ is given by the singular value decomposition $\Phi = U_1\Sigma_1V_1^T + U_2\Sigma_2V_2^T$ such that $\mathrm{range}(\Phi) = \mathrm{span}(U_1)$ and $\mathrm{null}(\Phi^T) = \mathrm{span}(U_2)$. Then W can be decomposed in terms of this basis as $W = U_1X_1 + U_2X_2$ with $X_1 = U_1W$ and $X_1 = U_1W$. Rewriting (6.7) in terms of this basis yields

$$\min_{X_1,X_2} \frac{1}{2}\|(V_1\Sigma_1X_1 - Y^T)T^{\frac{1}{2}}\|_F^2 + \eta\|U_1X_1 + U_2X_2\|_*. \tag{6.8}$$

As T is positive definite and hence has full rank, the first term is strongly convex in X_1. As the sum of the convex term with a strongly convex term the whole problem is unique in X_1. Thus it remains to be shown that $X_2 = \mathbf{0}$.

Therefore suppose that $X_2 \neq \mathbf{0}$ minimizes the nuclear norm term. In general one has

$$\|U_1X_1 + U_2X_2\|_* = \left\|\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}\right\|_*$$

as the nuclear norm is unitary invariant. Hence, it does not change under left multiplication of U^T. Instead of looking at the singular values of $[X_1^T, X_2^T]^T$ one can equivalently consider the eigenvalues of

$$\begin{bmatrix} X_1^T & X_2^T \end{bmatrix}\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = X_1^TX_1 + X_2^TX_2.$$

Horn and Johnson [1990, Theorem 4.3.1] connect the eigenvalues of the sum of symmetric matrices to the eigenvalues of the individual matrices. Let $\lambda_k(\boldsymbol{X})$ denote the k-th eigenvalue of $\boldsymbol{X}$ ordered from small to large eigenvalues, then $\lambda_k(\boldsymbol{A}+\boldsymbol{B}) \geq \lambda_k(\boldsymbol{A}) + \lambda_1(\boldsymbol{B})$. As $\boldsymbol{X}_2 \neq \boldsymbol{0}$ one has $\lambda_1(\boldsymbol{X}_2^T\boldsymbol{X}_2) \geq 0$. Hence, $\lambda_k(\boldsymbol{X}_1^T\boldsymbol{X}_1 + \boldsymbol{X}_2^T\boldsymbol{X}_2) \geq \lambda_k(\boldsymbol{X}_1^T\boldsymbol{X}_1)$. Furthermore note that $\operatorname{tr}(\boldsymbol{X}_1^T\boldsymbol{X}_1 + \boldsymbol{X}_2^T\boldsymbol{X}_2) = \operatorname{tr}(\boldsymbol{X}_1^T\boldsymbol{X}_1) + \operatorname{tr}(\boldsymbol{X}_2^T\boldsymbol{X}_2) > \operatorname{tr}(\boldsymbol{X}_1^T\boldsymbol{X}_1)$ as $\boldsymbol{X}_2 \neq \boldsymbol{0}$. One definition of the trace is the sum of the eigenvalues. Using the first relation one can conclude that all eigenvalues of $\boldsymbol{X}_1^T\boldsymbol{X}_1 + \boldsymbol{X}_2^T\boldsymbol{X}_2$ are at least as big as the corresponding eigenvalue of $\boldsymbol{X}_1^T\boldsymbol{X}_1$. The second relation states that the sum of eigenvalues of $\boldsymbol{X}_1^T\boldsymbol{X}_1 + \boldsymbol{X}_2^T\boldsymbol{X}_2$ is strictly greater than the sum of eigenvalues of $\boldsymbol{X}_1^T\boldsymbol{X}_1$. The combination of these two relations shows that either $\boldsymbol{X}_1^T\boldsymbol{X}_1 + \boldsymbol{X}_2^T\boldsymbol{X}_2$ has at least one additional positive eigenvalue or at least one of its eigenvalues is strictly greater than the corresponding eigenvalue of $\boldsymbol{X}_1^T\boldsymbol{X}_1$ or both. Due to monotonicity, the same relation also holds for the singular values of $[\boldsymbol{X}_1^T, \boldsymbol{X}_2^T]^T$ and $\boldsymbol{X}_1$. This concludes the proof as

$$\left\|\begin{bmatrix}\boldsymbol{X}_1\\ \boldsymbol{X}_2\end{bmatrix}\right\|_* > \left\|\begin{bmatrix}\boldsymbol{X}_1\\ \boldsymbol{0}\end{bmatrix}\right\|_*$$

is a contradiction to the assumption that $\boldsymbol{X}_2$ minimizes the norm. Therefore $\boldsymbol{X}_2 = 0$ and the solution to (6.6) is unique. □

6.3.2 Choosing the range of the regularization parameter

To select a model with good generalization performance the regularization parameter η needs to be chosen. To choose an appropriate range for η consider the following Lemma.

Lemma 6.2. *For $\eta \geq \eta_0 = \sigma_{\max}(\boldsymbol{\Phi}\boldsymbol{P}_1^{\perp}\boldsymbol{Y}^T\boldsymbol{T})$ the solution of* (6.6) *is given by*

$$\boldsymbol{W}_0 = \boldsymbol{0} \qquad \text{and} \qquad \boldsymbol{b}_0 = \frac{1}{N}\boldsymbol{T}^{-1}\boldsymbol{Y}\boldsymbol{1}_N. \tag{6.9}$$

Here, $\boldsymbol{P}_1^{\perp} = \frac{1}{N}\boldsymbol{1}_N\boldsymbol{1}_N^T$ denotes the projector onto the null space of $\boldsymbol{1}_N$ and $\sigma_{\max}(\boldsymbol{X})$ the largest singular value of $\boldsymbol{X}$.

Proof. By eliminating $\boldsymbol{e}_n$ from (6.6), the problem can be written in unconstrained form as $\min_{\boldsymbol{W},\boldsymbol{b}} \mathcal{J}(\boldsymbol{W},\boldsymbol{b})$ with

$$\mathcal{J}(\boldsymbol{W},\boldsymbol{b}) = \frac{1}{2}\|(\boldsymbol{\Phi}^T\boldsymbol{W} + \boldsymbol{1}_N\boldsymbol{b}^T - \boldsymbol{Y}^T)\boldsymbol{T}^{\frac{1}{2}}\|_F^2 + \eta\|\boldsymbol{W}\|_*.$$

A necessary condition [Bertsekas, 1999] for $(\boldsymbol{W}_0, \boldsymbol{b}_0)$ to be an optimal solution of (6.6) is that the subdifferential $\partial\mathcal{J}$ at $(\boldsymbol{W}_0, \boldsymbol{b}_0)$ contains the zero element $(\mathbf{0}, \mathbf{0})$. Using matrix calculus [Petersen and Pedersen, 2008] one obtains

$$\partial_{\boldsymbol{W}}\mathcal{J} = \boldsymbol{\Phi}(\boldsymbol{\Phi}^T\boldsymbol{W} + \mathbf{1}_N\boldsymbol{b}^T - \boldsymbol{Y}^T)\boldsymbol{T} + \eta\partial_{\boldsymbol{W}}\|\boldsymbol{W}\|_*,$$
$$\partial_{\boldsymbol{b}}\mathcal{J} = \boldsymbol{T}(\boldsymbol{W}^T\boldsymbol{\Phi} + \boldsymbol{b}\mathbf{1}_N^T - \boldsymbol{Y})\mathbf{1}_N.$$

Evaluating $\partial_{\boldsymbol{b}}\mathcal{J}$ for $\boldsymbol{W}_0 = \mathbf{0}$ and setting it equal to zero, one obtains the solution $\boldsymbol{b}_0$ stated above.

The subgradient of the nuclear norm is given in [Watson, 1992]. Let $\boldsymbol{U\Sigma V}^T$ be the thin singular value decomposition [Golub and Van Loan, 1996] of $\boldsymbol{X}$. Then $\partial\|\boldsymbol{X}\|_* = \{\boldsymbol{UV}^T + \boldsymbol{Z}, \boldsymbol{U}^T\boldsymbol{Z} = \mathbf{0}, \boldsymbol{ZV} = \mathbf{0}, \|\boldsymbol{Z}\|_2 \leq 1\}$ where $\boldsymbol{X}$ and $\boldsymbol{Z}$ are matrices of same dimension.

This allows the evaluation of the subdifferential $\partial_{\boldsymbol{W}}\mathcal{J}$ at $\boldsymbol{W}_0$. As $\boldsymbol{W}_0 = \mathbf{0}$, bases for its row and column spaces are given by $\boldsymbol{U} = \boldsymbol{V} = \mathbf{0}$. It follows that $\partial_{\boldsymbol{W}}\mathcal{J}(\boldsymbol{W}_0, \boldsymbol{b}_0) = -\boldsymbol{\Phi P}_1^{\perp}\boldsymbol{Y}^T\boldsymbol{T} + \eta\boldsymbol{Z}$ with $\|\boldsymbol{Z}\|_2 \leq 1$ has to contain $\mathbf{0}$. This translates to the necessary condition $\boldsymbol{\Phi P}_1^{\perp}\boldsymbol{Y}^T\boldsymbol{T} = \eta\boldsymbol{Z}$ which is satisfied for all $\eta \geq \eta_0$. □

6.4 Dual formulation of the model

So far the model formulated in (6.6) is parametric and requires the selection of appropriate basis functions. One advantage of kernel based modeling is that the choice of basis functions is simplified by reducing it to the choice of a kernel function. The kernel function often induces very large sets of basis functions, but the inherent regularization is an effective methodology to counter overfitting effects.

To exploit the kernel formulation already employed in the previous chapters, Mercer's condition must be applied, to transition from the parametric primal problem (6.6) to its nonparametric dual. One approach based on convex optimization will be the topic of the present section. The advantage of doing this in an optimization based setting is that additional information about the system can be incorporated in form of additional constraints, as shown in other chapters of this thesis or in [Espinoza et al., 2007].

Remark 6.1. Instead of engaging in the endeavor of kernelizing this complex regularization scheme, one can resort to more straightforward ways to integrate the ideas of support vector regression and nuclear norm regularization as in (6.6). In Section 4.3.1 the Nyström approximation has already been briefly described as a way to approximate a given kernel function on a given set of data. The result is a set of approximate basis functions that span the space induced by the kernel. This approximation is tailored to the distribution of the given data sample. By using this approximation, one has a straightforward mean to select basis functions for (6.6) while taking ideas from kernel based learning into account.

6.4.1 Dual optimization problem

In an optimization setting, there are at least two approaches to derive a dual formulation for (6.6), one relying on conic duality, the other based on the definition of the dual norm. Both arguments are conceptually similar and mostly differ in notation and level of abstractness. In this section the derivation will be based on the dual norm, while in some of the later chapters the conic duality approach will be utilized.

The definition of the dual norm $\|\cdot\|_D$ to the norm $\|\cdot\|_P$ is given by

$$\|\boldsymbol{X}\|_D = \max_{\|\boldsymbol{Y}\|_P \leq 1} \operatorname{tr}(\boldsymbol{X}^T \boldsymbol{Y}), \tag{6.10}$$

where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are matrices of identical dimensions. The dual norm for the nuclear norm is the matrix 2-norm, also known as operator or spectral norm. It is defined as $\|\boldsymbol{X}\|_2 := \sigma_{\max}(\boldsymbol{X})$ where $\sigma_{\max}(\boldsymbol{X})$ is the largest singular value of $\boldsymbol{X}$.

Then the kernel based, dual optimization problem corresponding to (6.6) is given by the following lemma.

Lemma 6.3. *The solution to* (6.6) *is equivalent to the solution of its Lagrange dual*

$$\begin{aligned} &\max_{\boldsymbol{A} \in \mathbb{R}^{M \times N}} \quad \operatorname{tr}(\boldsymbol{A}^T \boldsymbol{Y}) - \frac{1}{2} \operatorname{tr}(\boldsymbol{A}^T \boldsymbol{T}^{-1} \boldsymbol{A}) \\ &\text{subject to} \\ &\quad \boldsymbol{A} \mathbf{1}_N = \mathbf{0}_M, \quad \|\boldsymbol{G}\boldsymbol{A}^T\|_2 \leq \eta \end{aligned} \tag{6.11}$$

with $\boldsymbol{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N] \in \mathbb{R}^{M \times N}$, $\mathbf{1}_N \in \mathbb{R}^N$ *a vector of all ones and* $\mathbf{0}_M \in \mathbb{R}^M$ *a vector of all zeros. The matrix* $\boldsymbol{G}$ *is defined as a matrix square root, such that* $\boldsymbol{G}^T\boldsymbol{G} = \boldsymbol{\Omega}$. *The elements of the Gram matrix* $\boldsymbol{\Omega}$ *can be computed using the kernel trick* $\Omega_{ij} = \boldsymbol{\varphi}(\boldsymbol{x}_i)^T \boldsymbol{\varphi}(\boldsymbol{x}_j) = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ *for* $i, j = 1, \ldots, N$.

Proof. Introducing the short hand notation $\boldsymbol{E} = [\boldsymbol{e}_1, \ldots \boldsymbol{e}_N] \in \mathbb{R}^{M \times N}$, the objective function in (6.6) can be rewritten as $\eta\|\boldsymbol{W}\|_2 + \frac{1}{2}\operatorname{tr}(\boldsymbol{E}^T \boldsymbol{T} \boldsymbol{E})$. Similarly the modeling constraint can be reduced to $\boldsymbol{Y} = \boldsymbol{W}^T\boldsymbol{\Phi} + \boldsymbol{b}\mathbf{1}_N^T + \boldsymbol{E}$ where $\boldsymbol{\Phi} = [\varphi(x_1), \dots, \varphi(x_N)] \in \mathbb{R}^{n_h \times N}$. Using the definition of the dual norm (6.10), the nuclear norm term can be reformulated as $\eta\|\boldsymbol{W}\|_* = \eta \max_{\|C\|_2 \leq 1} \operatorname{tr}(\boldsymbol{C}^T\boldsymbol{W}) = \max_{\|C\|_2 \leq \eta} \operatorname{tr}(\boldsymbol{C}^T\boldsymbol{W})$ with $\boldsymbol{C} \in \mathbb{R}^{n_h \times M}$. Then the Lagrangian for (6.6) is

$$\mathscr{L}(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{E}, \boldsymbol{A}) = \max_{\|C\|_2 \leq \eta} \operatorname{tr}(\boldsymbol{C}^T\boldsymbol{W}) + \frac{1}{2}\operatorname{tr}(\boldsymbol{E}^T\boldsymbol{T}\boldsymbol{E}) - \operatorname{tr}(\boldsymbol{A}^T(\boldsymbol{W}^T\boldsymbol{\Phi} + \boldsymbol{b}\mathbf{1}_N^T + \boldsymbol{E} - \boldsymbol{Y}))$$

where $\boldsymbol{A}$ is the matrix of Lagrange multipliers for the equality constraints. Using matrix calculus [Petersen and Pedersen, 2008] one can take the Karush-Kuhn-Tucker (KKT) conditions for optimality [Boyd and Vandenberghe, 2004]:

$$\frac{\partial \mathscr{L}}{\partial \boldsymbol{W}} = 0: \quad \boldsymbol{C} = \boldsymbol{\Phi}\boldsymbol{A}^T, \tag{6.12a}$$

$$\frac{\partial \mathscr{L}}{\partial \boldsymbol{b}} = 0: \quad \boldsymbol{A}\mathbf{1}_N = \mathbf{0}_M, \tag{6.12b}$$

$$\frac{\partial \mathscr{L}}{\partial \boldsymbol{E}} = 0: \quad \boldsymbol{E} = \boldsymbol{T}^{-1}\boldsymbol{A}. \tag{6.12c}$$

Using these relations the Lagrangian can be reduced to the objective function of (6.11). Finally only the constraint $\|\boldsymbol{C}\|_2 = \|\boldsymbol{\Phi}\boldsymbol{A}^T\|_2 \leq \eta$ contains references to the unknown feature map. To kernelize the constraint, note that the following statement is equivalent $\|\boldsymbol{A}\boldsymbol{\Phi}^T\boldsymbol{\Phi}\boldsymbol{A}^T\|_2 = \|\boldsymbol{A}\boldsymbol{\Omega}\boldsymbol{A}^T\|_2 \leq \eta^2$. Using the matrix square root $\boldsymbol{G}$ defined earlier, so is $\|\boldsymbol{G}\boldsymbol{A}^T\|_2 \leq \eta$. □

6.4.2 Properties of the dual model

Analog to properties of the parametric problem derived in Section 6.3, similar results will be derived for the dual problem (6.11).

In case of the dual, uniqueness of the solution is straightforward to show.

Corollary 6.4. *For $\eta > 0$ the solution of* (6.11) *is unique in $\boldsymbol{A}$.*

Proof. The weighting matrix $\boldsymbol{T}$ is nonsingular by assumption. Hence, the term $\operatorname{tr}(\boldsymbol{A}^T\boldsymbol{T}^{-1}\boldsymbol{A})$ is strongly convex [Boyd and Vandenberghe, 2004]. As the feasible set is nonempty, Slater's condition is satisfied and (6.11) has a unique solution. □

The critical value for η for which the optimal solution stays constant from Lemma 6.2 can also be expressed in terms of the kernel matrix instead of the feature map.

Corollary 6.5. *For $\eta \geq \eta_0$ with $\eta_0 = \sigma_{\max}(\boldsymbol{G}\boldsymbol{P}_1^\perp \boldsymbol{Y}^T \boldsymbol{T})$ the solution for* (6.11) *is* $\boldsymbol{A} = \boldsymbol{T}\boldsymbol{Y}\boldsymbol{P}_1^\perp$.

Proof. The value of η_0 is already given by Lemma 6.2, $\eta_0 = \sigma_{\max}(\boldsymbol{\Phi}\boldsymbol{P}_1^\perp \boldsymbol{Y}^T \boldsymbol{T})$. However, it explicitly references the feature map $\boldsymbol{\Phi}$ which often is not accessible in a kernel based setting. Therefore note that $\eta_0^2 = \lambda_{\max}(\boldsymbol{T}\boldsymbol{Y}\boldsymbol{P}_1^\perp \boldsymbol{\Phi}^T \boldsymbol{\Phi}\boldsymbol{P}_1^\perp \boldsymbol{Y}^T \boldsymbol{T})$, where $\lambda_{\max}$ denotes the largest eigenvalue. The product $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ can be rewritten as $\boldsymbol{\Omega}$ using the kernel trick. Using the square root $\boldsymbol{G}$ of the kernel matrix, one obtains the critical value η_0 stated in the corollary.

To determine the optimal value of $\boldsymbol{A}$, note that for $\eta > \eta_0$ the inequality constraint will be inactive. Hence, it can be removed from the problem. The remaining problem has a quadratic objective and a linear equality constraint. This can be solved using Lagrangian duality. Let $\boldsymbol{d} \in \mathbb{R}^M$ denote Lagrange multipliers for the equality constraint, then the Lagrangian for (6.11) without the inequality constraint is

$$\mathscr{L}(\boldsymbol{A}, \boldsymbol{d}) = \operatorname{tr}(\boldsymbol{A}^T \boldsymbol{Y}) - \frac{1}{2} \operatorname{tr}(\boldsymbol{A}^T \boldsymbol{T}^{-1} \boldsymbol{A}) - \boldsymbol{d}^T \boldsymbol{A} \mathbf{1}_N.$$

The KKT condition for $\boldsymbol{A}$ yields $\boldsymbol{A} = \boldsymbol{T}(\boldsymbol{Y} - \boldsymbol{d}\mathbf{1}_N^T)$, whereas from the KKT condition for $\boldsymbol{d}$ one regains the equality constraint $\boldsymbol{A}\mathbf{1}_N = \mathbf{0}_M$. By substituting one into the other one obtains $\boldsymbol{d} = \frac{1}{N}\boldsymbol{T}\boldsymbol{Y}\mathbf{1}_N$ and $\boldsymbol{A} = \boldsymbol{T}\boldsymbol{Y}(\boldsymbol{I}_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T) = \boldsymbol{T}\boldsymbol{Y}\boldsymbol{P}_1^\perp$. □

Remark 6.2. Instead of relying on optimization theory and Lagrangian duality, SVM solutions can alternatively be derived using Representing Kernel Hilbert Spaces (RKHSs) [Wahba, 1998] by proving a representer theorem [Kimeldorf and Wahba, 1971]. For unitarily invariant matrix norms this has been done in [Argyriou et al., 2009] of which the nuclear norm is a special case.

The derivation presented here has the advantage of being constructive and that additional constraints can be integrated straightforwardly. Also it yields a different optimization problem.

6.5 Predictive model

From Lemma 6.3 one can obtain the optimal solution for (6.6) even if the feature map is not known explicitly. However, this solution is of no immediate use as the predictive model given by (6.2) is formulated in terms of the primal variables $\boldsymbol{w}_i$. For problems with Tikhonov type regularization like (6.4) establishing a link between the primal and dual variables is straightforward. From the KKT condition for w_i, c.f. Subsection 4.1.1, it directly follows that

$$w_i = \sum_{t=1}^{N} \alpha_t^{(i)} \varphi(x_t).$$

Establishing a similar relation between the solutions of (6.11) and (6.6) is however much more involved.

Preliminaries

The following original result will provide the basis to establish such a connection.

Theorem 6.6. *Let X and Z be $M \times N$ matrices. Furthermore assume that Z has r singular values equal to one while all others are strictly smaller. Finally let U_1 and V_1 denote the left and right singular vectors corresponding to the unit singular values of Z respectively. Then for any $\xi > 0$ it holds*

$$\begin{aligned} &\{X : \operatorname{tr}(X^T Z) = \xi, \|X\|_* = \xi\} \\ &\qquad = \{U_1 H_1 V_1^T : \operatorname{tr}(H_1) = \xi, 0 \preceq H_1 \in \mathbb{R}^{r\times r}\}. \end{aligned} \tag{6.13}$$

Proof. See Appendix A.1. □

The use of this result is within its combination with the definition of the dual norm, $\|X\|_* = \max_{\|Z\|_2 \le 1} \operatorname{tr}(X^T Z)$. From the theorem it follows that there is no one-to-one relation between X and its dual variables Z. In fact given a dual matrix Z, a whole set of primal matrices X with prespecified norm $\|X\|_* = \xi$ satisfies the definition of the dual norm. This gives rise to several insights:

- A given dual solution alone is not enough to recover the primal variables and derive a predictive model.
- To characterize the set of possible solutions, the norm of the primal solution is required.

The norm of W in (6.6) can be obtained in a straightforward manner as detailed in the following lemma.

Lemma 6.7. *Given the optimal solution A to problem* (6.11) *the optimal value for $\|W\|_*$ in* (6.6) *is given by*

$$\|W\|_* = \eta^{-1}(\operatorname{tr}(A^T Y) - \operatorname{tr}(A^T T^{-1} A)). \tag{6.14}$$

Proof. First it is shown that the duality gap between (6.6) and (6.11) is zero. In a second step the objective functions are equated to obtain the desired result.

The sufficient condition for a vanishing duality gap is strong duality. For Corollary 6.5 it has already been argued that Slater's condition – and therefore strong duality – holds for (6.11). As a direct consequence the duality gap is zero.

Therefore $\eta\|\boldsymbol{W}\|_* + \frac{1}{2}\operatorname{tr}(\boldsymbol{E}^T\boldsymbol{T}\boldsymbol{E}) = \operatorname{tr}(\boldsymbol{A}^T\boldsymbol{Y}) - \frac{1}{2}\operatorname{tr}(\boldsymbol{A}^T\boldsymbol{T}^{-1}\boldsymbol{A})$, where $\boldsymbol{E} = [\boldsymbol{e}_1, \dots, \boldsymbol{e}_N]$. Exploiting $\boldsymbol{E} = \boldsymbol{T}^{-1}\boldsymbol{A}$ taken from the KKT condition (6.12c) of (6.6) one obtains the result stated above. □

Linking dual solution to primal variables

Recovering the primal variables can be split in two parts. Determining $\boldsymbol{b}$ is straightforward and handled first. The recovery of $\boldsymbol{W}$ is more involved and explained thereafter.

Corollary 6.8. *The optimal value for* $\boldsymbol{b}$ *in* (6.6) *is given by*

$$\boldsymbol{b} = \frac{1}{N}(\boldsymbol{Y} - \boldsymbol{W}^T\boldsymbol{\Phi})\mathbf{1}_N. \tag{6.15}$$

Proof. Eliminating $\boldsymbol{e}_n$ from (6.6) yields

$$\min_{\boldsymbol{W},\boldsymbol{b}} \quad \eta\|\boldsymbol{W}\|_* + \frac{1}{2}\operatorname{tr}((\boldsymbol{W}^T\boldsymbol{\Phi} + \boldsymbol{b}\mathbf{1}_N^T - \boldsymbol{Y})^T\boldsymbol{T}(\boldsymbol{W}^T\boldsymbol{\Phi} + \boldsymbol{b}\mathbf{1}_N^T - \boldsymbol{Y})).$$

Taking the KKT condition for $\boldsymbol{b}$ then gives $\mathbf{1}_N^T\mathbf{1}_N\boldsymbol{T}\boldsymbol{b} - \boldsymbol{T}(\boldsymbol{Y} - \boldsymbol{W}^T\boldsymbol{\Phi})\mathbf{1}_N = 0$. The desired expression is then obtained by rearrangement. □

Using the solution for $\boldsymbol{b}$ and exploiting the knowledge about all matrices $\boldsymbol{W}$ that are consistent with the dual solution, it is possible to derive a closed form solution for $\boldsymbol{W}$ in terms of the dual variables $\boldsymbol{A}$.

Corollary 6.9. *Let* $\eta \leq \eta_0$ *and* $\boldsymbol{U\Sigma V}^T$ *denote the thin singular value decomposition of* $\boldsymbol{C}$ *from the proof of Lemma 6.3. Furthermore let the matrices* $\boldsymbol{U}_\eta$ *and* $\boldsymbol{V}_\eta$ *contain all left and right singular vectors corresponding to the largest singular value of* $\boldsymbol{C}$ *–* η *– respectively. Then*

$$\boldsymbol{W} = \boldsymbol{C}\boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T, \tag{6.16}$$

with $\boldsymbol{H}_\eta \succeq 0$, $\operatorname{tr}(\boldsymbol{H}_\eta) = \eta^{-1}\|\boldsymbol{W}\|_*$ *where* $\boldsymbol{H}_\eta$ *has compatible dimensions.*

Proof. It holds that $\eta\|\boldsymbol{W}\|_* = \mathrm{tr}(\boldsymbol{C}^T\boldsymbol{W})$ and $\|\boldsymbol{C}\|_2 = \eta$. Then by applying Theorem 6.6 one obtains $\boldsymbol{W} = \boldsymbol{U}_\eta\boldsymbol{H}_1\boldsymbol{V}_\eta^T$ with $\boldsymbol{H}_1 \geq 0$ and $\mathrm{tr}(\boldsymbol{H}_1) = \|\boldsymbol{W}\|_*$. Now consider $\boldsymbol{C}\boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T = \boldsymbol{U\Sigma V}^T\boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T = \eta\boldsymbol{U}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T$. As $\boldsymbol{H}_\eta = \eta^{-1}\boldsymbol{H}_1$, one obtains the desired relation. □

Note that $\boldsymbol{C} = \boldsymbol{\Phi}\boldsymbol{A}^T$, therefore $\boldsymbol{V}_\eta$ can be extracted from the finite dimensional eigenvalue decomposition of $\boldsymbol{A\Phi}^T\boldsymbol{\Phi A}^T = \boldsymbol{A\Omega A}^T$. The last missing piece is to determine the matrix $\boldsymbol{H}_\eta$.

Lemma 6.10. *Under the same conditions as defined by Lemma 6.3 and Corollary 6.9 one obtains*

$$\boldsymbol{H}_\eta = \eta^{-2}\boldsymbol{V}_\eta^T(\boldsymbol{Y}\boldsymbol{A}^T - \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{A}^T)\boldsymbol{V}_\eta. \tag{6.17}$$

Proof. Substituting (6.16) and the KKT condition (6.12c) for $\boldsymbol{E}$ into the equality constraint of (6.6) in its compressed form as used in Lemma 6.3 yields

$$\boldsymbol{Y} = \boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T\boldsymbol{C}^T\boldsymbol{\Phi} + \boldsymbol{b}\boldsymbol{1}_N^T + \boldsymbol{T}^{-1}\boldsymbol{A}.$$

Any solution has to satisfy the KKT conditions. As $\boldsymbol{W}$, $\boldsymbol{b}$ and $\boldsymbol{A}$ form the primal and dual optimal solution, the linear system above is guaranteed to be consistent. Therefore right multiplication with $\boldsymbol{A}^T$ does not remove any information. Recall that $\boldsymbol{C} = \boldsymbol{\Phi}\boldsymbol{A}^T$. In a first step the equation can then be simplified to

$$\boldsymbol{Y}\boldsymbol{A}^T = \boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T\boldsymbol{C}^T\boldsymbol{C} + \boldsymbol{b}\boldsymbol{1}_N^T\boldsymbol{A}^T + \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{A}^T.$$

Exploiting (6.12b) the KKT condition for $\boldsymbol{b}$, one can drop the term $\boldsymbol{b}\boldsymbol{1}_N^T\boldsymbol{A}^T$. Additionally the relation $\boldsymbol{C}^T\boldsymbol{C} = \boldsymbol{V\Sigma}^2\boldsymbol{V}^T$ allows simplifying another term. This then yields

$$\boldsymbol{Y}\boldsymbol{A}^T = \eta^2\boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T + \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{A}^T.$$

Solving this for $\boldsymbol{H}_\eta$, one finally obtains the expression given in (6.17). □

Remark 6.3. To check the accuracy of a numerical solution of the dual problem (6.11) one can check several properties of $\boldsymbol{H}_\eta$ as it has to be (i) symmetric, (ii) positive definite and (iii) its trace has to equal $\|\boldsymbol{W}\|_*$ as given by Lemma (6.7).

Finally the previous results can be combined to give direct relations between $\boldsymbol{W}$, $\boldsymbol{b}$ and $\boldsymbol{A}$.

Corollary 6.11. *The optimal values for $\boldsymbol{W}$ and $\boldsymbol{b}$ in (6.6) in terms of the dual optimal solution $\boldsymbol{A}$ are given by*

$$\boldsymbol{W} = \boldsymbol{\Phi}\boldsymbol{A}^T\boldsymbol{M} \qquad \textit{and} \qquad \boldsymbol{b} = \frac{1}{N}(\boldsymbol{Y} - \boldsymbol{M}\boldsymbol{A}\boldsymbol{\Omega})\mathbf{1}_N \tag{6.18}$$

with

$$\boldsymbol{M} = \eta^{-2}\boldsymbol{P}_\eta(\boldsymbol{Y}\boldsymbol{A}^T - \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{A}^T)\boldsymbol{P}_\eta \tag{6.19}$$

where $\boldsymbol{P}_\eta = \boldsymbol{V}_\eta\boldsymbol{V}_\eta^T$.

Proof. Substitution of (6.17) into (6.16) and using that $\boldsymbol{C} = \boldsymbol{\Phi}\boldsymbol{A}^T$ yields the first half of (6.18) along with (6.19). The second half of (6.18) follows from substituting the first half into (6.15) and applying the kernel trick $\boldsymbol{\Phi}^T\boldsymbol{\Phi} = \boldsymbol{\Omega}$. Note that as $\boldsymbol{H}_\eta$ is symmetric and so is $\boldsymbol{M}$. □

Predictive equation and algorithm

Using the link between primal and dual solutions established on the previous pages, it is possible to formulate a predictive model for the dual solution in a straightforward manner.

Corollary 6.12. *With the definitions from Corollary 6.11 the predictive model for a new point $\boldsymbol{z}$, in terms of the dual variables, is given by*

$$\widehat{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{z}) = \sum_{t=1}^{N} \tilde{\boldsymbol{\alpha}}_t K(\boldsymbol{x}_t, \boldsymbol{z}) + \boldsymbol{b}. \tag{6.20}$$

The variables $\tilde{\boldsymbol{\alpha}}_t$ form the matrix $\widetilde{\boldsymbol{A}} = [\tilde{\boldsymbol{\alpha}}_1, \dots, \tilde{\boldsymbol{\alpha}}_N]$, which is computed as $\widetilde{\boldsymbol{A}} = \boldsymbol{M}\boldsymbol{A}$.

Proof. For convenience the model formulated in (6.2) can be cast in vector notation as $\widehat{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{z}) = \boldsymbol{W}^T\boldsymbol{\varphi}(\boldsymbol{z}) + \boldsymbol{b}$. The predictive equation (6.20) then directly follows after substitution of $\boldsymbol{W}$ given in (6.18) and application of the kernel trick. □

An overview of the most important ingredients of the parametric or primal model on the one hand and the kernel based or dual model on the other hand is presented in Table 6.1. Finally the following algorithm summarizes all actions that are necessary to estimate a model from given data and using this model to generate predictions at an unknown point $\boldsymbol{z}$.

Table 6.1: Overview of parametric/primal and kernel based/dual estimation problems and the corresponding models.

	PARAMETRIC MODEL	KERNEL BASED MODEL
basis functions	choose $\boldsymbol{\varphi}$	choose kernel K
model estimation	solve (6.6) for $\boldsymbol{W}$ and $\boldsymbol{b}$	solve (6.11) for $\boldsymbol{A}$
obtaining model representation	prespecified	obtain $\boldsymbol{M}$ and $\boldsymbol{b}$ from Corollary 6.11
generating predictions	$f(z) = \boldsymbol{W}^T\boldsymbol{\varphi}(z) + \boldsymbol{b}$	$f(z) = \sum_{t=1}^{T} \tilde{\boldsymbol{\alpha}}_t K(\boldsymbol{x}_t, z) + \boldsymbol{b}$

Algorithm 6.1. Given a kernel function $K(\boldsymbol{x}, \boldsymbol{y})$, data $\{x_n, y_n\}_{n=1}^{N}$ and a regularization constant η[1] proceed as follows

1. Compute kernel matrix $\Omega_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $i, j = 1, \dots, N$.
2. Compute a matrix square root $\boldsymbol{G}$ such that $\boldsymbol{\Omega} = \boldsymbol{G}^T\boldsymbol{G}$.
3. Solve dual problem (6.11) to obtain $\boldsymbol{A}$.
4. Compute the thin SVD of $\boldsymbol{A\Omega A}^T$ and form the matrix $\boldsymbol{V}_\eta$ from the eigenvectors corresponding to the largest eigenvalue (η^2).
5. Evaluate (6.19) to obtain mixing matrix $\boldsymbol{M}$.
6. Generate predictions at a new point $\boldsymbol{z}$ by evaluating the model given by (6.20).

6.6 Extensions

In Subsection 6.2.1 it has already been briefly mentioned that the model given by (6.2) is not the only possible choice. In the following, the modifications needed to allow for a model structure as defined by (6.3), are discussed. At the same time the necessity to have measured every output variable at every time step is lifted. This corresponds to given data specified like $\{\{(\boldsymbol{x}_t^{(i)}, y_t^{(i)})\}_{t\in\mathbb{S}_i}\}_{i=1}^{M}$ where the sets $\mathbb{S}_i$ contain all time instances at which the i-th output is measured.

Another specialization of the problem definition (6.6) is the choice of the least-squares loss to penalize the modeling residuals. Of course, other loss functions like the ε-insensitive loss known from SVMs or the robust Huber

[1] The results from Lemma 6.2 and Corollary 6.5 can be used to aid the choice of η.

loss, are valid choices. In these cases the derivation of the dual problem are straightforward extensions of the previously presented material. However, the recovery of the mixing term $\boldsymbol{M}$ is more involved and requires solving a semidefinite programming problem in the primal. Basically the form (6.16) along with $\boldsymbol{C} = \boldsymbol{\Phi}\boldsymbol{A}^T$ has to substituted into a primal problem analogous to (6.6) and solved for $\boldsymbol{H}_\eta$ as well as $\boldsymbol{b}$. Due to the mostly technical nature of the re-derivation this is left as an exercise for the interested reader.

An important alternation in the context of this thesis is the application to overparametrized models as found in Chapters 7 and 8. For these, the model formulation is closer to the one given in (6.3), however the special structure of the regression vector allows reducing the computational complexity to a level only slightly larger than for the simpler (6.2).

6.6.1 Variable input and output data

For this subsection the model stated in (6.3) will be considered as well as possibly incompletely measured data. The data given by $\{\{\boldsymbol{x}_t^{(i)}, y_t^{(i)}\}_{t\in\mathbb{S}_i}\}_{i=1}^M$, where $|\mathbb{S}_i| = N_i$ and $\mathbb{S}_i = \{s_i\}_{i=1}^{N_i}$.

Both generalizations are formalized in the modified primal estimation problem

$$\min_{\boldsymbol{W},\boldsymbol{b},\boldsymbol{e}^{(i)}} \quad \eta\|\boldsymbol{W}\|_* + \frac{1}{2}\sum_{i=1}^{M} t_i \boldsymbol{e}^{(i)T}\boldsymbol{e}^{(i)} \tag{6.21}$$

subject to

$$\boldsymbol{y}^{(i)} = \boldsymbol{\Phi}_i^T \boldsymbol{w}_i + b_i \mathbf{1}_{N_i} + \boldsymbol{e}^{(i)}, \qquad i = 1, \dots, M,$$

where $\boldsymbol{y}^{(i)} = [y_{s_1}^{(i)}, \dots, y_{s_{N_i}}^{(i)}]^T \in \mathbb{R}^{N_i}$, $\boldsymbol{\Phi}_i = [\boldsymbol{\varphi}(\boldsymbol{x}_{s_1}^{(i)}), \dots, \boldsymbol{\varphi}(\boldsymbol{x}_{s_{N_i}}^{(i)})] \in \mathbb{R}^{n_h \times N_i}$, $\boldsymbol{e}^{(i)} = [e_{s_1}^{(i)}, \dots, e_{s_{N_i}}^{(i)}]^T \in \mathbb{R}^{N_i}$ and $\boldsymbol{w}_i$ the i-th column of $\boldsymbol{W}$.

Remark 6.4. Note that the weighting matrix $\boldsymbol{T}$ has to be reduced to simple weighting factors $t_i > 0$ with respect to the original problem formulation given by (6.6). This is necessary as the residuals $\boldsymbol{e}^{(i)}$ can have different dimensions. A possible mitigation to restore a completely flexible weighting matrix $\boldsymbol{T}$ would be padding the vectors $\boldsymbol{e}^{(i)}$ with zeros to equalize their lengths. This however will not be considered further.

Remark 6.5. Also note that the equality constraint in (6.21) is transposed with respect to the one in (6.6). It is also written in terms of the target variables (M constraints) instead of the samples (N constraints). This is again triggered by the possible different dimensionalities of the considered quantities.

The dual optimization problem for (6.21) is given in the following lemma which generalizes Lemma 6.3.

Lemma 6.13. *The solution to* (6.21) *is equivalent to the solution of its Lagrange dual*

$$\begin{aligned} &\max_{\boldsymbol{\alpha}_i} \quad \sum_{i=1}^{M} \boldsymbol{\alpha}_i^T \boldsymbol{y}^{(i)} - \frac{1}{2} \sum_{i=1}^{M} \frac{1}{t_i} \boldsymbol{\alpha}_i^T \boldsymbol{\alpha}_i \\ &\text{subject to} \\ &\quad \boldsymbol{\alpha}_i^T \mathbf{1}_{N_i} = 0, \quad i = 1, \dots, M, \\ &\quad \|\widetilde{\boldsymbol{G}}\widetilde{\boldsymbol{A}}\|_2 \leq \eta \end{aligned} \tag{6.22}$$

with $\boldsymbol{\alpha}_i \in \mathbb{R}^{N_i}$ *and the* $\tilde{N} \times M$ *block structured matrix* $\widetilde{\boldsymbol{A}}$ *with* $\tilde{N} = \sum_{i=1}^{M} N_i$, $\widetilde{\boldsymbol{A}}_{ii} = \boldsymbol{\alpha}_i$ *and* $\widetilde{\boldsymbol{A}}_{ij} = \mathbf{0}_{N_i}$ *for* $i \neq j$. *The matrix* $\widetilde{\boldsymbol{G}}$ *is defined as a matrix square root such that* $\widetilde{\boldsymbol{G}}^T \widetilde{\boldsymbol{G}} = \widetilde{\boldsymbol{\Omega}}$. *The* $\tilde{N} \times \tilde{N}$ *Gram matrix is block structured with* $(\widetilde{\boldsymbol{\Omega}})_{ij} = \widetilde{\boldsymbol{\Omega}}_{ij} \in \mathbb{R}^{N_i \times N_j}$ *for* $i, j = 1, \dots, M$ *and* $(\widetilde{\boldsymbol{\Omega}}_{ij})_{kn} = K(x_k^{(i)}, x_n^{(j)})$ *for* $k = 1, \dots, N_i$ *and* $n = 1, \dots, N_j$.

Proof. Taking the KKT conditions as in Lemma 6.3 one obtains $\boldsymbol{c}_i = \boldsymbol{\Phi}_i \boldsymbol{\alpha}_i$ for $\boldsymbol{w}_i$, $t_i \boldsymbol{e}^{(i)} = \boldsymbol{\alpha}_i$ for $\boldsymbol{e}^{(i)}$ and $\mathbf{1}_{N_i}^T \boldsymbol{\alpha}_i = 0$ for b_i. The vector $\boldsymbol{c}_i$ is the i-th column of the matrix $\boldsymbol{C}$ introduced in Lemma 6.3 and $\boldsymbol{\alpha}_i$ are the Lagrange multipliers corresponding to the equality constraints in (6.21). To kernelize the constraint $\|\boldsymbol{C}\|_2 \leq \eta$, let $\widetilde{\boldsymbol{\Phi}} = [\boldsymbol{\Phi}_1, \dots, \boldsymbol{\Phi}_M]$. This allows $\boldsymbol{C}$ to be written as $\widetilde{\boldsymbol{\Phi}}\widetilde{\boldsymbol{A}}$. The kernel trick can again be applied by squaring the constraint. The final expression with a linear argument follows by taking a matrix square root. □

In the following corollaries, the results from Section 6.5 are adapted to the extended problem formulation.

Corollary 6.14. *Given the optimal solution* $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_M$ *to problem* (6.22), *the optimal value for* $\|\boldsymbol{W}\|_*$ *in* (6.21) *is given by*

$$\|\boldsymbol{W}\|_* = \eta^{-1} \left(\sum_{i=1}^{M} \boldsymbol{\alpha}_i^T \boldsymbol{y}^{(i)} - \sum_{i=1}^{M} t_i^{-1} \boldsymbol{\alpha}_i^T \boldsymbol{\alpha}_i \right). \tag{6.23}$$

Proof. The proof is a straightforward adaption of the proof to Lemma 6.7. □

Corollary 6.15. *The matrix $\boldsymbol{H}_\eta$ introduced in Corollary 6.9 can be determined by solving the semidefinite programming problem*

$$\begin{aligned}
&\text{find} \quad \boldsymbol{H}_\eta \\
&\text{subject to} \\
&\quad \boldsymbol{H}_\eta \succeq 0, \operatorname{tr}(\boldsymbol{H}_\eta) = \xi \\
&\quad \boldsymbol{y}^{(i)} = [\boldsymbol{\Omega}_{i,1}\boldsymbol{\alpha}_1, \dots, \boldsymbol{\Omega}_{i,M}\boldsymbol{\alpha}_M]\boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T\boldsymbol{\varepsilon}_i + b_i\boldsymbol{1}_{N_i} + t_i^{-1}\boldsymbol{\alpha}_i, \quad i = 1, \dots, M,
\end{aligned} \tag{6.24}$$

where the $\boldsymbol{\varepsilon}_i$'s form the standard basis for $\mathbb{R}^M$ and $\xi = \eta^{-1}\|\boldsymbol{W}\|_*$.

Proof. The first two constraints follow directly from Corollary 6.9. The remaining constraints can be derived in a fashion similar to Lemma 6.10.

Let $\boldsymbol{w}_i = \boldsymbol{W}\boldsymbol{\varepsilon}_i$. Then the equality constraints of (6.21) can be expressed in terms of the dual variables by further using (6.16), describing the form of $\boldsymbol{W}$, and $\boldsymbol{e}^{(i)} = t_i^{-1}\boldsymbol{\alpha}_i$, following form the KKT condition for $\boldsymbol{e}^{(i)}$. This yields

$$\boldsymbol{y}^{(i)} = \boldsymbol{\Phi}_i^T[\boldsymbol{\Phi}_1\boldsymbol{\alpha}_1, \dots, \boldsymbol{\Phi}_M\boldsymbol{\alpha}_M]\boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T\boldsymbol{\varepsilon}_i + b_i\boldsymbol{1}_{N_i} + t_i^{-1}\boldsymbol{\alpha}_i, \tag{6.25}$$

for $i = 1, \dots, M$. Application of the kernel trick then results in the problem given above. □

Corollary 6.16. *Given the matrix $\boldsymbol{H}_\eta$, along with the optimal dual solution $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_M$, the primal optimal variables $\boldsymbol{w}_1, \dots, \boldsymbol{w}_M$ and $b_1, \dots, b_M$ can be represented as*

$$\boldsymbol{w}_i = \sum_{j=1}^{M} Q_{ji}\boldsymbol{\Phi}_j\boldsymbol{\alpha}_j \qquad \text{and} \qquad b_i = \frac{1}{N_i}\left(\boldsymbol{1}_{N_i}^T\boldsymbol{y}^{(i)} - \sum_{j=1}^{M} Q_{ji}\boldsymbol{1}_{N_i}^T\boldsymbol{\Omega}_{ij}\boldsymbol{\alpha}_j\right), \tag{6.26}$$

for $i = 1, \dots, M$ and with $(\boldsymbol{Q})_{ij} = Q_{ij}$ and $\boldsymbol{Q} = \boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T$.

Proof. The form of $\boldsymbol{w}_i$ is given by $\boldsymbol{w}_i = \boldsymbol{W}\boldsymbol{\varepsilon}_i$ and Corollary 6.9. Along with the definition of $\boldsymbol{C} = [\boldsymbol{\Phi}_1\boldsymbol{\alpha}_1, \dots, \boldsymbol{\Phi}_M\boldsymbol{\alpha}_M]$ one obtains

$$\boldsymbol{w}_i = [\boldsymbol{\Phi}_1\boldsymbol{\alpha}_1, \dots, \boldsymbol{\Phi}_M\boldsymbol{\alpha}_M]\boldsymbol{V}_\eta\boldsymbol{H}_\eta\boldsymbol{V}_\eta^T\boldsymbol{\varepsilon}_i.$$

Careful inspection reveals that this is identical to the first expression in (6.26). In analogy to Corollary 6.8 one can derive

$$b_i = \frac{1}{N_i}\left(\boldsymbol{1}_{N_i}^T\boldsymbol{y}^{(i)} - \boldsymbol{1}_{N_i}^T\boldsymbol{\Phi}_i^T\boldsymbol{w}_i\right),$$

for $i = 1, \dots, M$. The substitution of the previously derived expression for $\boldsymbol{w}_i$ into the form of b_i and the application of the kernel trick finally yields the second half of (6.26). □

Table 6.2: Size of data matrices for joint multiple output regression.

MODEL	DIMENSION OF $\boldsymbol{\Phi}$	DIMENSION OF $\boldsymbol{\Omega}$
basic (6.6)	$n_h \times N$	$N \times N$
extended (6.21)	$n_h \times (M \cdot N)$	$(M \cdot N) \times (M \cdot N)$

Finally combining all of the above, one can once more state a predictive model.

Corollary 6.17. *With the definitions from Corollary 6.16 the predictive model for a new point $(z^{(1)}, \dots, z^{(M)})$, in terms of the dual variables, is given by*

$$\widehat{y}_i = f(z^{(i)}) = \sum_{j=1}^{M} Q_{ji} \boldsymbol{k}_j(z^{(i)})^T \boldsymbol{\alpha}_j + b_i, \tag{6.27}$$

with $\boldsymbol{k}_j(\zeta) = [K(\zeta, \boldsymbol{x}_1^{(j)}), \dots, K(\zeta, \boldsymbol{x}_{N_j}^{(j)})]^T$ for $i = 1, \dots, M$.

Proof. The predictive equation (6.27) directly follows from substituting (6.26) into (6.3) and applying the kernel trick. □

Remark 6.6 (Numerical complexity). The extended model (6.21) presented in this section is a generalization of the basic model (6.6) discussed before. Hence, the question arises, how much more expensive is solving the more general, and consequently more powerful, problem. To aid this comparison it is assumed that $N_i \equiv N$ for $i = 1, \dots, M$.

Careful inspection of the referenced problems reveals that the number of unknowns, in the primal as well as in the dual problems, is identical for both model formulations. However, looking at the data, significant differences become evident. In the primal, the data is transformed using the feature map $\boldsymbol{\varphi}$ and stored in the matrix $\boldsymbol{\Phi}$. The corresponding roles in the dual are taken over by kernel function K along with the Gram matrix $\boldsymbol{\Omega}$. As shown in Table 6.2, the extended model uses a data representation that is M times larger in the primal, and even M^2 larger in the dual, than the one of the basic model.

However, this should not come as surprise, as the amount of input data processed by the extended model is also M times larger. Yet, it should be carefully considered when choosing one model or the other.

6.6.2 Overparametrized models

Consider a dynamical model of the form

$$\hat{y}_t = f(\boldsymbol{x}_{t-1}, \dots, \boldsymbol{x}_{t-M_y}) = \sum_{i=1}^{M_y} \boldsymbol{w}_i^T \boldsymbol{\varphi}(\boldsymbol{x}_{t-i}) + b \tag{6.28}$$

for $i = 1, \dots, M_y$. In essence this is close to using a linear filter on top of a nonlinear structure. In fact in the following two chapters expressions of the form $\beta_i f(\boldsymbol{x}_{t-i})$ will be relaxed to form $\boldsymbol{w}_i \boldsymbol{\varphi}(\boldsymbol{x}_{t-i})$ introduced above. The model derivation however is closely related to the other models introduced in this chapter, which is why it will be discussed at this point.

In contrast to the remainder of this chapter, a model of this form only has a single output variable. Nevertheless the most important property, a linear relation between the $\boldsymbol{w}_i$'s is present. Besides being limited to a single output problem, the formulation has another peculiarity, namely a special structure of the regressors. At time t the output is a function of $\boldsymbol{x}_{t-1}$ up to $\boldsymbol{x}_{t-M_y}$. Therefore the difference between t and $t+1$ is given by a single element of this sequence. This special structure has the advantage that the computational complexity, as well as the size of the matrices, is much closer to the basic model (6.4) than the extended one (6.3), c.f. Table 6.2.

Based on the model given above, the estimation problem can be formalized as

$$\min_{\boldsymbol{W},\boldsymbol{b},\boldsymbol{e}} \quad \eta \|\boldsymbol{W}\|_* + \frac{1}{2}\boldsymbol{e}^T\boldsymbol{e} \tag{6.29}$$

subject to

$$\boldsymbol{y} = \sum_{i=1}^{M_y} \boldsymbol{\Phi}_i^T \boldsymbol{w}_i + b\mathbf{1}_{N-M_y} + \boldsymbol{e},$$

where $\boldsymbol{y} = [y_{M_y+1}, \dots, y_N]^T \in \mathbb{R}^{N-M_y}$, $\boldsymbol{e} = [e_{M_y+1}, \dots, e_N]^T \in \mathbb{R}^{N-M_y}$, $\boldsymbol{\Phi}_i = [\boldsymbol{\varphi}(\boldsymbol{x}_{1+(M_y-i)}), \dots, \boldsymbol{\varphi}(\boldsymbol{x}_{N-i+1})]$ and $\boldsymbol{w}_i$ the i-th column of $\boldsymbol{W}$.

In the following the previously derived results will be tailored to this newly defined optimization problem. First of all, the solution can still be obtained in the finite dimensional dual domain.

Lemma 6.18. *The solution to* (6.29) *can be equivalently obtained from its Lagrange dual*

$$\begin{aligned} &\max_{\boldsymbol{\alpha}} \quad \boldsymbol{\alpha}^T \boldsymbol{y} - \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{\alpha} \\ &\text{subject to} \\ &\quad \mathbf{1}_{N-M_y}^T \boldsymbol{\alpha} = 0, \quad \|\boldsymbol{G}\mathcal{B}(\boldsymbol{\alpha})\|_2 \leq \eta. \end{aligned} \tag{6.30}$$

Here $\boldsymbol{\alpha} \in \mathbb{R}^{N-M_y}$ *denotes the Lagrange multipliers for the equality constraints of* (6.29). *The linear operator* $\mathcal{B} : \mathbb{R}^{N-M_y} \to \mathbb{R}^{N \times M_y}$ *maps the vector* $\boldsymbol{\alpha}$ *to a block structured matrix. It is given by*

$$\mathcal{B}(\boldsymbol{\alpha}) = \begin{bmatrix} \boldsymbol{\alpha} & & \mathbf{0}_{M_y} \\ & \ddots & \\ \mathbf{0}_{M_y} & & \boldsymbol{\alpha} \end{bmatrix}. \tag{6.31}$$

Proof. The derivation is a straightforward extension to the proof given for Lemma 6.3. In the first step, variables $\boldsymbol{c}_i = \boldsymbol{\Phi}_i \boldsymbol{\alpha}$ can be introduced. Note that stacking them as columns into a matrix $\boldsymbol{C}$ is equivalent to $\boldsymbol{C} = \boldsymbol{\Phi}\mathcal{B}(\boldsymbol{\alpha})$. Here the definition for the matrix $\boldsymbol{\Phi} = [\varphi(x_1), \ldots, \varphi(x_N)] \in \mathbb{R}^{n_h \times N}$ is the same as in Lemma 6.3.

Furthermore the KKT conditions for b and $\boldsymbol{e}$ take on simpler forms

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial b} &= 0 : \quad \mathbf{1}_{N-M_y}^T \boldsymbol{\alpha} = 0, \\ \frac{\partial \mathcal{L}}{\partial \boldsymbol{e}} &= 0 : \quad \boldsymbol{e} = \boldsymbol{\alpha}. \end{aligned}$$

The remainder of the proof contains no notable differences. □

Recovering a predictive model follows the same path as before and is summarized in the following corollaries.

Corollary 6.19. *Given the optimal solution* $\boldsymbol{\alpha}$ *to problem* (6.30), *the optimal value for* $\|\boldsymbol{W}\|_*$ *in* (6.29) *is given by*

$$\|\boldsymbol{W}\|_* = \eta^{-1}\left(\boldsymbol{\alpha}^T \boldsymbol{y} - \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{\alpha}\right). \tag{6.32}$$

Proof. The proof is a straightforward adaption of the proof to Lemma 6.7. □

Corollary 6.20. *The matrix $\boldsymbol{H}_\eta$ introduced in Corollary 6.9, as well as the primal variable b can be determined by solving the semidefinite programming problem*

$$
\begin{aligned}
&\text{find} \quad (\boldsymbol{H}_\eta, b) \\
&\text{subject to} \\
&\quad \boldsymbol{H}_\eta \succeq 0, \operatorname{tr}(\boldsymbol{H}_\eta) = \xi \\
&\quad \boldsymbol{y} = \sum_{i=1}^{M_y} [\boldsymbol{\Omega}_{i,1}\boldsymbol{\alpha}, \dots, \boldsymbol{\Omega}_{i,M_y}\boldsymbol{\alpha}] \boldsymbol{V}_\eta \boldsymbol{H}_\eta \boldsymbol{V}_\eta^T \boldsymbol{\varepsilon}_i + b\mathbf{1}_{N-M_y} + \boldsymbol{\alpha},
\end{aligned}
\tag{6.33}
$$

where the $\boldsymbol{\varepsilon}_i$'s' form the standard basis for $\mathbb{R}^{M_y}$ and $\xi = \eta^{-1}\|\boldsymbol{W}\|_*$. Furthermore $\boldsymbol{\Omega}_{i,j}$ denotes the subblock of $\boldsymbol{\Omega}$ containing the i through $N - M_y + i$ rows and j through $N - M_y + j$ columns, respectively.

Proof. The first two constraints follow directly from Corollary 6.9. The remaining constraints can be derived in a fashion similar to Lemma 6.10.

Let $\boldsymbol{w}_i = \boldsymbol{W}\boldsymbol{\varepsilon}_i$. Then the equality constraints of (6.29) can be expressed in terms of the dual variables by further using (6.16), describing the form of $\boldsymbol{W}$. This yields

$$
\boldsymbol{y} = \sum_{i=1}^{M_y} \boldsymbol{\Phi}_i^T [\boldsymbol{\Phi}_1\boldsymbol{\alpha}, \dots, \boldsymbol{\Phi}_M\boldsymbol{\alpha}] \boldsymbol{V}_\eta \boldsymbol{H}_\eta \boldsymbol{V}_\eta^T \boldsymbol{\varepsilon}_i + b\mathbf{1}_{N-M_y} + \boldsymbol{\alpha},
$$

where additionally the KKT condition for $\boldsymbol{e}$ has been used to substitute $\boldsymbol{e}$ with $\boldsymbol{\alpha}$. Application of the kernel trick directly yields the aforementioned feasibility problem. □

Corollary 6.21. *Given the matrix $\boldsymbol{H}_\eta$, along with the optimal dual solution $\boldsymbol{\alpha}$, the primal optimal variable $\mathbf{W}$ can be represented as*

$$
\boldsymbol{W} = \boldsymbol{\Phi}\mathcal{B}(\boldsymbol{\alpha})\boldsymbol{Q} \tag{6.34}
$$

with $\boldsymbol{Q} = \boldsymbol{V}_\eta \boldsymbol{H}_\eta \boldsymbol{V}_\eta^T$.

Proof. The matrix $\boldsymbol{C}$ is given by $\boldsymbol{C} = \boldsymbol{\Phi}\mathcal{B}(\boldsymbol{\alpha})$. The expression for $\boldsymbol{W}$ can be immediately obtained by combining this with Corollary 6.9. □

Finally combining all of the above, one can once more state a predictive model.

Corollary 6.22. *With the definition of* $\boldsymbol{Q}$ *from Corollary 6.21 the predictive model for a new point* $(z_{t-1}, \ldots, z_{t-M_y})$, *in terms of the dual variables, is given by*

$$\widehat{y}_t = f(z_{t-1}, \ldots, z_{t-M_y}) = \sum_{i,j=1}^{M_y} Q_{ji} \boldsymbol{k}_j(z_{t-i})^T \boldsymbol{\alpha} + b, \tag{6.35}$$

with $\boldsymbol{k}_j(\zeta) = [K(\zeta, x_{M_y-j+1}), \ldots, K(\zeta, x_{N-j})]^T$ *for* $i = 1, \ldots, M$ *and* $Q_{ij} = (\boldsymbol{Q})_{ij}$.

Proof. The predictive equation (6.35) directly follows from substituting (6.34) into (6.28) and applying the kernel trick. □

6.7 Numerical solution

In the previous sections several primal optimization problems were formulated and converted into the respective dual formulations. However, all these different formulations are only meaningful if they can be solved on actual data. As suggested in the introduction of this chapter, there are several possibilities to pick from, when deciding how and which problem is to be solved.

On the one hand there are the primal formulations (6.6), (6.21) & (6.29) with direct reference to the feature map φ and on the other hand one has the respective dual representations (6.11), (6.22) & (6.30). Which representation is more advantageous to solve depends on several factors. In case the feature map consists of only a few explicitly known basis function, the dimension n_h is most likely much smaller than the number of data. In this case solving the primal problem is not only much more straightforward, it is also much more efficient. In case the basis functions are not known explicitly or their number n_h is larger than the amount of data, the dual should be considered.

However, in contrast to the LS-SVM case with a single target variable as introduced in Chapter 4, the advantage of the dual representation is not so clear. Note that solving the dual problem (6.11) requires (i) computing a matrix square root and (ii) additional computations to transform the dual solution back to the primal space. Both these steps require additional work not necessary for plain LS-SVMs. One attractive alternative can be the Nyström method outlined in Section 4.3.1. The main cost involved in this procedure is computing the eigenvalue decomposition of the kernel matrix. However, also solving the dual requires computing a matrix square root. Hence, depending on the chosen method to obtain this square root, the eigenvalue decomposition needed for the Nyström approximation is at most a constant factor more expensive.

Regardless of the ultimately chosen representation of the problem, the next subsection will highlight some possible choices to obtain numerical solutions for either of the two representations. In case efficiency is not a concern, any of the optimization problems stated in this chapter can be directly modeled as it is, using a modeling tool like CVX [Grant and Boyd, 2011]. As the extension to related model formulations is straightforward, only the basic problem (6.6) and its dual (6.11) will be discussed.

6.7.1 Semi-definite programming representation

As stated in Section 3.3.3 the nuclear norm can be represented by the SDP problem (3.12). Relying on this result, a representation suitable for a general purpose SDP solver is as follows.

Corollary 6.23. *The optimization problem stated in* (6.6) *can be reformulated using linear matrix inequalities (LMIs) as*

$$\begin{aligned}
&\min_{\boldsymbol{Z}_1,\boldsymbol{Z}_2,\boldsymbol{W},\boldsymbol{b},\tau_n} \quad \eta(\mathrm{tr}(\boldsymbol{Z}_1) + \mathrm{tr}(\boldsymbol{Z}_2)) + \sum_{t=1}^{N} \tau_t \\
&\text{subject to} \\
&\quad \begin{bmatrix} \boldsymbol{Z}_1 & \boldsymbol{W} \\ \boldsymbol{W}^T & \boldsymbol{Z}_2 \end{bmatrix} \succeq 0, \quad \boldsymbol{Z}_1, \boldsymbol{Z}_2 \succeq 0, \\
&\quad \begin{bmatrix} \boldsymbol{T}^{-1} & \mathcal{A}_t(\boldsymbol{W},\boldsymbol{b}) \\ \mathcal{A}_t(\boldsymbol{W},\boldsymbol{b})^T & \tau_n \end{bmatrix} \succeq 0, \quad t = 1, \dots, N.
\end{aligned}$$

The affine operators $\mathcal{A}_t$ from $\mathbb{R}^{n_h \times M} \times \mathbb{R}^M$ to $\mathbb{R}^M$ are defined as $\mathcal{A}_t(\boldsymbol{W},\boldsymbol{b}) = \boldsymbol{y}_t - \boldsymbol{W}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) - \boldsymbol{b}$ for $t = 1, \dots, N$.

Proof. To obtain a linear objective function, first rewrite the nuclear norm according to Equation 3.12. Furthermore the quadratic forms $\boldsymbol{e}_t^T \boldsymbol{T} \boldsymbol{e}_t$ of the residuals $\boldsymbol{e}_t$ can be replaced by additional constraints $\boldsymbol{e}_t^T \boldsymbol{T} \boldsymbol{e}_t \leq \tau_t$ and then summing over τ_t in the objective. Applying the Schur complement [Boyd and Vandenberghe, 2004] to the newly introduced constraint one obtains $\begin{bmatrix} \boldsymbol{T}^{-1} & \boldsymbol{e}_t \\ \boldsymbol{e}_t^T & \tau_t \end{bmatrix} \succeq 0$. The residuals $\boldsymbol{e}_t$ can be eliminated by rewriting the equality constraints of (6.6) as $\boldsymbol{e}_t = \mathcal{A}_t(\boldsymbol{W},\boldsymbol{b})$. □

Remark 6.7. Note that the reformulation of the residual term $\boldsymbol{e}_t = \mathcal{A}_t(\boldsymbol{W},\boldsymbol{b})$ as an LMI constraint is not very efficient. While some SDP solvers, like

CVXOPT [Dahl and Vandenberghe, 2011], support the solution of quadratic cost functions directly, other popular ones, like SDPT3 [Toh et al., 1999], require the reduction to a linear objective.

Similarly to the nuclear norm, the spectral norm is also SDP representable [Boyd and Vandenberghe, 2004]. The necessary derivations to transform (6.11) into a LMI are detailed below.

Corollary 6.24. *The solution for* (6.11) *can be obtained from the LMI problem*

$$\begin{aligned}
&\max_{\boldsymbol{A},\tau_t} \quad \operatorname{tr}(\boldsymbol{A}^T\boldsymbol{Y}) - \frac{1}{2}\sum_{t=1}^{N}\tau_t \\
&\text{subject to} \\
&\quad \begin{bmatrix} \boldsymbol{T} & \boldsymbol{\alpha}_t \\ \boldsymbol{\alpha}_t^T & \tau_t \end{bmatrix} \succeq 0, \quad t = 1, \dots, N, \\
&\quad \begin{bmatrix} \eta\boldsymbol{I}_N & \boldsymbol{G}\boldsymbol{A}^T \\ \boldsymbol{A}\boldsymbol{G}^T & \eta\boldsymbol{I}_M \end{bmatrix} \succeq 0, \quad \boldsymbol{A}\boldsymbol{1}_N = \boldsymbol{0}_M,
\end{aligned}$$

with $\boldsymbol{\alpha}_t$ being the t-th column of $\boldsymbol{A}$.

Proof. The spectral norm term $\|\boldsymbol{G}\boldsymbol{A}^T\|_2 \leq \eta$ is equal to $\|\boldsymbol{A}\boldsymbol{G}^T\boldsymbol{G}\boldsymbol{A}^T\|_2 \leq \eta^2$. Using the Schur complement [Boyd and Vandenberghe, 2004] it can be rewritten as the second LMI constraint in the optimization problem above. Note that $\operatorname{tr}(\boldsymbol{A}^T\boldsymbol{T}^{-1}\boldsymbol{A}) = \sum_{t=1}^{N}\boldsymbol{\alpha}_t^T\boldsymbol{T}^{-1}\boldsymbol{\alpha}_t$. Then the N LMI constraints representing $\boldsymbol{\alpha}_n^T\boldsymbol{T}^{-1}\boldsymbol{\alpha}_t$ are derived using the Schur complement in the same fashion as for $\boldsymbol{e}_t^T\boldsymbol{T}\boldsymbol{e}_t$ in Corollary 6.23. □

The main drawback of general purpose SDP solvers is that they do not scale to larger problem sizes which severely limits their practical use. The performance can be improved by exploiting the specific problem structure and tailoring a custom SDP solver as done in [Liu and Vandenberghe, 2009] for a nuclear norm regularized problem. Yet when doing this several of the advantages have to be sacrificed and the scaling issue has only been reduced but is still present.

6.7.2 First order methods

An alternative to interior point algorithms has been driven from efforts mainly in the compressed sensing area. These first-order algorithms have already been briefly outlined in Subsection 3.4.2. Problems with nuclear norm regularization have gained serious interest as evident by the number of publications dedicated to it, [Tseng, 2010; Ji and Ye, 2009; Toh and Yun, 2010; Pong et al., 2010; Cai et al., 2010; Jaggi and Sulovský, 2010]. Most of these algorithms are based on gradient projection. The crucial aspect for all algorithms are efficient projections, which for the nuclear norms were first derived in [Cai et al., 2010]. This result and a similar one for spectral norms are given in the next paragraphs.

Projections and proximal minimization

With $g(\boldsymbol{x})$ being the nuclear norm $\|\boldsymbol{X}\|_*$, the proximal minimization problem $\mathcal{P}(\boldsymbol{Y})$ given by

$$\mathcal{P}(\boldsymbol{y}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \eta g(\boldsymbol{x}) \tag{6.36}$$

can be solved based on the SVD of $\boldsymbol{Y}$.

Theorem 6.25 (Singular value thresholding [Cai et al., 2010, Section 2.1])**.** *Let $\boldsymbol{Y} \in \mathbb{R}^{M\times N}$ have the thin SVD [Golub and Van Loan, 1996] $\boldsymbol{Y} = \boldsymbol{U\Sigma V}^T$ with rank r and $\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1, \dots, \sigma_r)$. Also define the singular value thresholding operator $(\boldsymbol{\Sigma})_+ = \operatorname{diag}(\{\max(0, \sigma_i - \eta)\}_{i=1}^r)$ for $\eta > 0$. Then*

$$\mathcal{P}_{SVT}(\boldsymbol{Y}) = \boldsymbol{U}(\boldsymbol{\Sigma})_+\boldsymbol{V}^T \tag{6.37}$$

is the solution of the proximal minimization problem (6.36) *with g being the nuclear norm.*

The proof for this result is given in [Cai et al., 2010, Section 2.1]. A similar result can be obtained for the spectral norm.

Theorem 6.26 (Singular value clipping)**.** *Let $\boldsymbol{Y} \in \mathbb{R}^{M\times N}$ have rank r such that the thin SVD [Golub and Van Loan, 1996] is given by $\boldsymbol{Y} = \boldsymbol{U\Sigma V}^T$ with $\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1, \dots, \sigma_r)$. Define the piecewise linear map $m_{\boldsymbol{\Sigma}}(\sigma) = \sum_{i=1}^r \max(0, \sigma_i - \sigma)$. For $\eta > 0$ let $\sigma_C = m_{\boldsymbol{\Sigma}}^{-1}(\eta)$ if $\sum_{i=1}^r \sigma_i > \eta$ and $\sigma_C = 0$ otherwise. Finally define the singular value clipping operator $(\boldsymbol{\Sigma})_- = \operatorname{diag}(\{\min(\sigma_C, \sigma_i)\}_{i=1}^r)$. Then*

$$\mathcal{P}_{SVC}(\boldsymbol{Y}) = \boldsymbol{U}(\boldsymbol{\Sigma})_-\boldsymbol{V}^T \tag{6.38}$$

is the solution of the proximal minimization problem (6.36) *with g being the spectral norm.*

For a proof see Appendix A.2. Note that the dual problem (6.11) is not stated in the regularized form $\min_x f(x) + g(x)$ but in constrained form. The following corollary links both problems for g being the spectral norm.

Corollary 6.27. *For $0 < \eta' \leq \|\boldsymbol{Y}\|_2$ the constrained problem*

$$\min_{\boldsymbol{X}} \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{Y}\|_F^2 \quad \text{subject to} \quad \|\boldsymbol{X}\|_2 \leq \eta',$$

is solved by the singular value clipping operation (6.38) *with $\eta = m_{\boldsymbol{\Sigma}}(\eta')$ where $m_{\boldsymbol{\Sigma}}$ is defined as in Theorem 6.26.*

Proof. The Lagrangian for the constrained problem is $\mathscr{L}(\boldsymbol{X}, \alpha) = \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{Y}\|_F^2 + \alpha(\|\boldsymbol{X}\|_2 - \eta')$. The minimum in $\boldsymbol{X}$ can be obtained from Theorem 6.26 for $\eta = \alpha$. The KKT condition for α is $\eta' = \|\boldsymbol{X}\|_2$ as the constraint is always active. Therefore one has $\sigma_0 = m_{\boldsymbol{\Sigma}}^{-1}(\alpha)$. From Theorem 6.26 one also has $\|\boldsymbol{X}\|_2 = \|\mathcal{P}_{SVC}(\boldsymbol{Y})\|_2 = \sigma_0$. Therefore $\sigma_0 = \eta'$ and $\alpha = m_{\boldsymbol{\Sigma}}(\eta')$ follow. □

Remark 6.8. For $\eta \geq \eta_{SVT,0} = \|\boldsymbol{Y}\|_2$ the singular value thresholding operation yields $\mathcal{P}_{SVT}(\boldsymbol{Y}) = \boldsymbol{0}$. Similarly for $\eta \leq \eta_{SVC,0} = \|\boldsymbol{Y}\|_*$ the singular value clipping operation yields $\mathcal{P}_{SVC}(\boldsymbol{Y}) = \boldsymbol{0}$.

Implementation

The interest in the compressed sensing community is mainly for problems of the form

$$\min_{\boldsymbol{X}} \quad \frac{1}{2}\|\mathcal{A}(\boldsymbol{X}) - \boldsymbol{y}\|_2^2 + \eta\|\boldsymbol{X}\|_*,$$

where $\mathcal{A}$ is a linear operator. For problems in this form, several implementations of first order algorithms are publicly available, like SLEP [Liu and Vandenberghe, 2009; Ji and Ye, 2009; Pong et al., 2010], NNLS [Toh and Yun, 2010] and in principle FPCA [Goldfarb and Ma, 2009]. Rewriting (6.6) to the form given above is possible by eliminating the residuals $\boldsymbol{e}_t$. Then only minor modifications are necessary to account for the fact that only $\boldsymbol{W}$ and not $\boldsymbol{b}$ is subject to the nuclear norm penalty.

For the dual optimization problem (6.11) these algorithms cannot be applied directly. However, it is possible to come up with a gradient projection algorithm as outlined in 3.4.2 using the projection for the spectral norm as given in Theorem 6.26. Therefore it is needed to apply a a variable transformation $\boldsymbol{M} = \boldsymbol{G}\boldsymbol{A}^T$ to (6.11) such that the spectral norm constraint becomes $\|\boldsymbol{M}\|_2 \leq \eta$. The drawback of this approach is that (at least implicitly) the square root of the kernel matrix $\boldsymbol{G}$ needs to be inverted. In case the kernel matrix is not well conditioned this yields algorithms that are numerically not robust.

Remark 6.9 (Alternatives). A numerically more robust scheme can be obtained by adding a smoothing term $\mu||\boldsymbol{A} - \boldsymbol{A}_0||_F^2$ with $\mu > 0$ and $\boldsymbol{A}_0 \in \mathbb{R}^{M \times N}$ to the optimization objective. However, the bias introduced by the smoothing destroys the relations needed to reconstruct a predictive model.

A second alternative, that allows using the existing solvers for the primal problem, is solving the dual of the dual. Although the derivation as well as recovery of the dual solution are straightforward, this approach has no real benefit over using the Nyström approximation for directly solving the primal. This latter approach has to solve the same optimization problem with very similar or identical sizes for the data matrices. However, it is much easier to derive and does not require additional work to employ the dual solution in a predictive equation.

6.8 Numerical validation

Due to the numerical complexity only small scale problems can be considered in reasonable time. Therefore a simple toy example is used to illustrate the basic method.

6.8.1 Experimental setup

To validate the derived dual formulation versus the primal alternative the example is chosen with an explicitly known feature map. The data is generated according to the model

$$\boldsymbol{y}_t = \boldsymbol{W}_0^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + \boldsymbol{b}_0 + \boldsymbol{v}_t. \tag{6.39}$$

The dimension of $\boldsymbol{y}_t$, i.e. the number of outputs, is chosen as $M = 20$, while the rank of $\boldsymbol{W}_0$, i.e. the number of independent components, is set to 3. The dimension of the feature map is selected as $n_h = 50$. Finally the matrix $\boldsymbol{W}_0$ is generated as the product of one $n_h \times 3$ and another $3 \times M$ dimensional matrix, denoted by $\boldsymbol{W}_{0,B}$ and $\boldsymbol{W}_{0,M}$ respectively. The elements of both matrices are drawn from a standard normal distribution. For simplicity $\boldsymbol{b}_0$ is chosen identically zero.

The total number of generated data is 300. This data is split into three independent sets. 50 samples are used to solve the estimation problems (6.6) and (6.11) respectively. Another 100 samples are used as validation set to select the regularization parameter η. The remaining 150 samples are used for final model assessment.

Instead of choosing some feature map $\boldsymbol{\varphi}$ and evaluating it on data $\mathcal{D} = \{\boldsymbol{x}_t\}_{t=1}^{300}$ to obtain the matrix $\boldsymbol{\Phi}$ containing evaluations of the feature map on the data, the matrix $\boldsymbol{\Phi}$ is generated directly for the sake of simplicity. Once more the elements of the matrix $\boldsymbol{\Phi}$ are drawn from a standard normal distribution.

To make the problem nontrivial, the data is corrupted by additive white Gaussian noise, in the form of $\boldsymbol{v}_t$ which is generated with a standard deviation of 0.1.

The proposed method, denoted by MIMO, corresponding to (6.6) is compared to three alternatives.

OLS Ordinary least squares:

$$\min_{\boldsymbol{W}} \quad \sum_{t=1}^{N} \|\boldsymbol{y}_t - \boldsymbol{W}^T \boldsymbol{\varphi}(\boldsymbol{x}_t)\|_2^2,$$

with $\boldsymbol{W} \in \mathbb{R}^{n_h \times M}$,

OLS + Oracle Ordinary least squares coupled with an oracle which provides exact and complete information for the structure of $\boldsymbol{W}_0$. Recall that $\boldsymbol{W}_0$ is generated as $\boldsymbol{W}_{0,B} \cdot \boldsymbol{W}_{0,M}$. The oracle specifies $\boldsymbol{W}_{0,M}$ as used during the construction of $\boldsymbol{W}_0$ such that the problem is reduced to estimating the parameters of the three independent components in $\boldsymbol{W}_{0,B}$. For doing so, $M = 20$ measurements can be utilized.

$$\min_{\boldsymbol{W}_B} \quad \sum_{t=1}^{N} \|\boldsymbol{y}_t - \boldsymbol{W}_{M,0}^T \boldsymbol{W}_B^T \boldsymbol{\varphi}(\boldsymbol{x}_t)\|_2^2,$$

with $\boldsymbol{W}_B \in \mathbb{R}^{n_h \times 3}$.

RR Ridge regression. Ridge regression has no means to take advantage of the known low rank structure. Furthermore the estimation problem can be split into M independent estimation problems. To enable a fair comparison, the regularization parameter for each of the problems is chosen independently. Therefore for each of the M components one has to solve the problem

$$\min_{\boldsymbol{w}_m} \quad \eta_m \, \boldsymbol{w}_m^T \boldsymbol{w}_m + \sum_{t=1}^{N} (y_t^{(m)} - \boldsymbol{w}_m^T \boldsymbol{\varphi}(\boldsymbol{x}_t))^2.$$

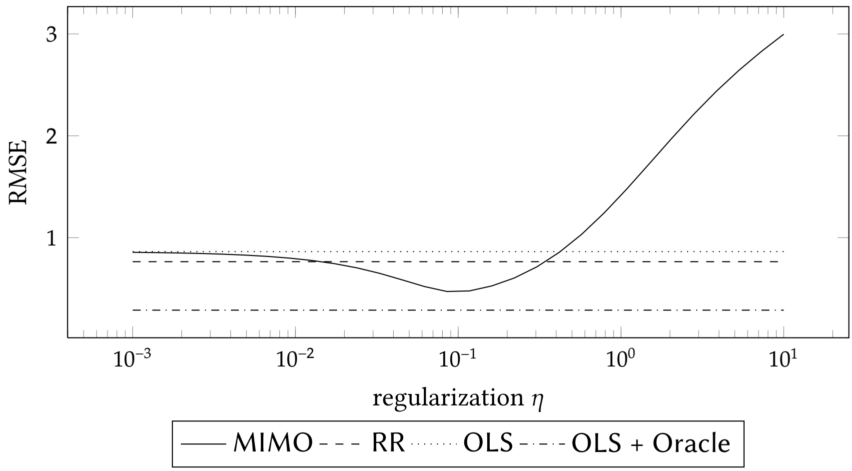

Figure 6.1: Validation performance of different multivariate model structures for toy dataset. The predictive performance is shown as a function of the regularization parameter. The root mean squared error in this figure is computed with respect to time t and component m.

6.8.2 Results

Using this simple example, three experiments are conducted. The first one considers the prediction performances of the different models and compares them. Another performance measure for the compared methods is their ability to reconstruct the true parameters $\boldsymbol{W}_0$ used to generate the data. The last experiment looks at the numerical difference between solving the primal and dual problems as shown in Table 6.1.

Prediction performance

Figure 6.1 shows the root mean squared error for the four different models on the validation set as a function of the regularization parameter η. Note that the models OLS and OLS + Oracle are not regularized and thus are constant over the whole parameters range. The model obtained by ridge regression has 20 different regularization parameters, one for each output. To simplify the presentation the best parameter for each of the outputs is chosen and the resulting overall best model is illustrated by the third constant. The reported performance is the RMSE not only with respect to time but also with respect

Table 6.3: Predictive performance for multivariate toy dataset. All given quantities are RMSE values with respect to time and the output, for different partitions of the data.

	TRAINING	VALIDATION	TEST
OLS	0	0.8633	0.8459
OLS + Oracle	0.0920	0.2885	0.2981
RR	0.0175	0.7644	0.7867
MIMO	0.0184	0.4716	0.4974

to the outputs, i.e.

$$\mathrm{RMSE} = \sqrt{\frac{1}{N \cdot M} \sum_{t=1}^{N} \sum_{m=1}^{M} (y_t^{(m)} - \hat{y}_t^{(m)})^2}.$$

Table 6.3 shows the performance on training, validation and test set, for the best model of each type.

One can see that ridge regression is slightly better than ordinary least squares. It is also evident that providing exact structural information as in OLS + Oracle greatly improves the prediction performance of the model. The proposed scheme clearly outperforms OLS as well as RR in predictive performance. However, it is still significantly worse than OLS + Oracle. This is to be expected as MIMO only has the prior information that there is some low rank structure in the parameters while the OLS + Oracle is supplied with the exact dependencies linking the different outputs.

Reconstruction performance

A second test for the considered models is the accuracy at which the matrix $\boldsymbol{W}_0$ is reconstructed. The relative accuracy is measured as

$$\frac{\|\boldsymbol{W}_0 - \widehat{\boldsymbol{W}}\|_F}{\|\boldsymbol{W}_0\|_F}.$$

Figure 6.2 depicts the accuracy as function of the regularization parameter. The absolute performances for the best models are: 5.97% for OLS, 2.01% for OLS + Oracle, 5.48% for RR and 3.42% for MIMO. The overall picture is very similar to that of the predictive performance. The most notable difference is

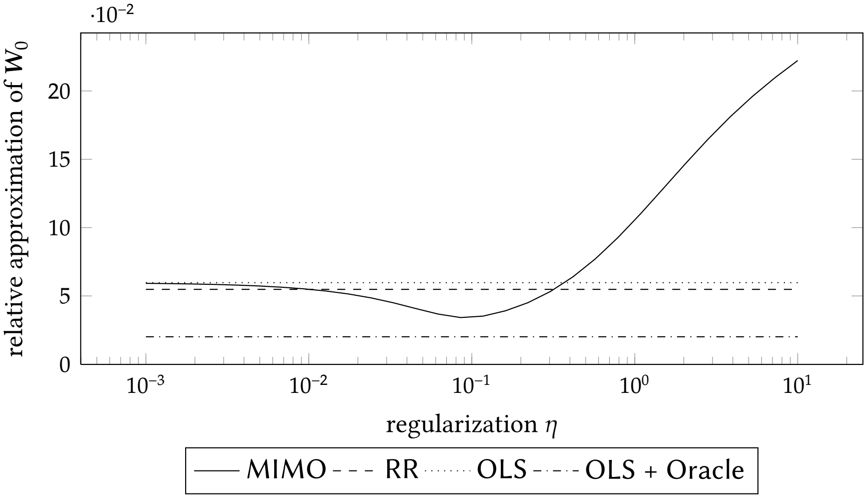

Figure 6.2: The relative approximation error is computed as $\|\widehat{W} - W_0\|_F / \|W_0\|_F$.

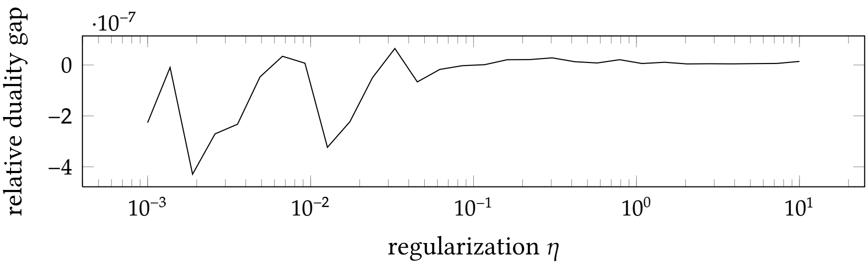

Figure 6.3: Relative duality gap between optimal values of (6.6) and (6.11) for toy example. Denote the primal optimal value by p^* and the dual optimal value by d^* respectively. Then the reported quantity is $(p^* - d^*)/p^*$.

that the advantage of the proposed method is a bit bigger here than it was for the predictive performance.

One interesting observation is that the optimal regularization parameter coincides for both performance measures. However, in a real application W_0 is unknown. Hence, only the predictive performance can be used for selection of the regularization parameter.

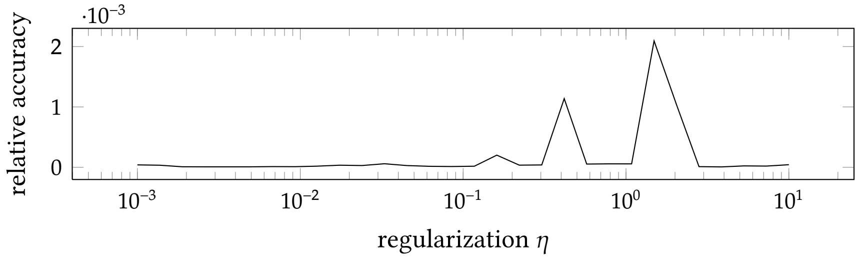

Figure 6.4: Relative accuracy of parameter estimate $\widehat{\boldsymbol{W}}$ for toy example. For the primal problem (6.6) the value $\boldsymbol{W}_{primal}$ is a direct result of the optimization problem. For the dual (6.11) the optimal parameter $\boldsymbol{W}_{dual}$ is recovered using the equation from Corollary 6.11. The reported quantity is $||\boldsymbol{W}_{primal} - \boldsymbol{W}_{dual}||_F / ||\boldsymbol{W}_{primal}||_F$.

Numerical comparison of primal and dual solutions

One major claim of this chapter to verify numerically is the equivalence of the derived primal and dual solutions as summarized in Table 6.1. Therefore consider the duality gap as a function of the regularization parameter as shown in Figure 6.3. With an overall scale of 10^{-7} it is evident that, within the limits of the solver, the achieved duality gap is zero. Note that the obtained negative values of the gap can most likely be accounted to numerical imprecisions of the solver. The values are strictly positive for $\eta \gtrapprox 0.1$.

A more demanding quality check is depicted in Figure 6.4, namely the agreement of the estimated parameter $\widehat{\boldsymbol{W}}$. While for the primal problem, the parameter can be directly extracted from the optimization problem, in case of the dual an additional set of equations as stated in Corollary 6.11 has to be evaluated. Again the solutions of primal and dual problems are in very good agreement with an overall scale of the results of 10^{-3}. Except for the three visible peaks the majority of the curve is even on a scale of 10^{-5}.

A second numerical example, based on the overparametrized formulation outlined in Subsection 6.6.2, is given in next chapters Subsection 7.4.3. There also some timing results are given.

6.9 Conclusions

This chapter considered a regularization scheme suitable for systems with multiple outputs. By encouraging linear dependence between the model parameters for different outputs, information can flow from one model part to another. After giving a thorough motivation in Subsections 6.1 and 6.2, some important properties of the proposed problem are studied in the following section. The main contribution is the derivation of the kernel based estimation problem in a primal-dual setting, which is discussed in Section 6.4. The second crucial result worked out as part of this chapter in Section 6.5 is the form of the predictive model in terms of the dual solution. Whereas for LS-SVMs it is straightforwardly obtained from the KKT condition for w, in the case considered here several auxiliary problems have to be solved. The chapter continues in Section 6.6 and presents extensions of the basic formulation used in the remainder of this chapter. In particular it generalizes the estimation problem to be applicable to a larger class of systems. Furthermore it is tailored to a specific model structure that will be used in the following chapters. The chapter continues by reviewing different possibilities to obtain numerical solutions. To validate the presented material the proposed method is finally studied on a numerical example.

7 Block structured models

Based on the publications

- Falck, T., Dreesen, P., De Brabanter, K., Pelckmans, K., De Moor, B., and Suykens, J. A. K. (Nov. 2012). "Least-Squares Support Vector Machines for the Identification of Wiener-Hammerstein Systems". In: *Control Engineering Practice* 20(11), pp. 1165–1174,
- Falck, T., Suykens, J. A. K., Schoukens, J., and De Moor, B. (Dec. 2010). "Nuclear Norm Regularization for Overparametrized Hammerstein Systems". In: *Proceedings of the 49th IEEE Conference on Decision and Control.* (Atlanta, GA, USA, Dec. 15–17, 2010), pp. 7202–7207.

Further publications within this context [Falck, Pelckmans, et al., 2009; Goethals et al., 2010].

7.1 Introduction

Prior knowledge about a nonlinear system is often available in the form of its structure. For example a sensor or an actuator might have nonlinear characteristics, like saturation, while the remainder of the system can be approximated well by a linear system. Models that contain linear dynamical and static nonlinear blocks are referred to as Hammerstein and Wiener systems. If the first element of a system is a nonlinear function followed by a linear block it is denoted as Hammerstein system. In case the order is reversed and a linear system is followed by a static nonlinearity the concatenation is known as Wiener system. Combinations like Hammerstein-Wiener and Wiener-Hammerstein systems generalize these basic building blocks. An example for

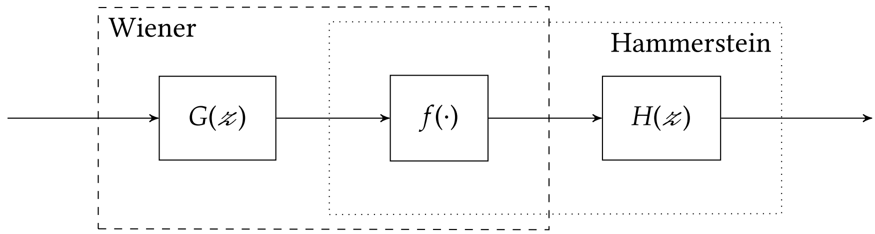

Figure 7.1: General structure of a Wiener-Hammerstein system with linear blocks $G(z)$ and $H(z)$ and static nonlinearity $f(\cdot)$. The dashed and dotted boxes indicate subsystems of Wiener and Hammerstein type respectively.

a block structured Wiener-Hammerstein model is shown in Figure 7.1. This chapter will focus on Wiener-Hammerstein systems which will be treated by extending an existing model for Hammerstein systems.

One of the first identification techniques for Wiener-Hammerstein structured systems was proposed by Billings and Fakhouri [1978]. Since then there has been a large interest in the identification of block structured models. A recent monograph that presents several state of the art approaches is [Giri and Bai, 2010]. However most attention so far has been paid to Wiener and Hammerstein systems while their combinations were subject to relatively little research [Tan and Godfrey, 2002; Bershad et al., 2001; Enqvist and Ljung, 2005; Greblicki and Pawlak, 2008]. More recently this has changed partially due to a dedicated special session at SYSID2009 [Schoukens et al., 2009].

The work presented in this chapter is founded on the identification approach for Hammerstein systems introduced by Goethals [Goethals et al., 2005b,a, 2010]. There LS-SVM based models are modified such that they incorporate knowledge about the structure of the underlying system. The major advantage of the proposed technique over many others is that it is based on convex optimization and therefore gives consistent results regardless of the initialization. In the work of Goethals the starting point is a nonconvex optimization problem that is then approximated by a convex problem. The approach taken is known as overparametrization and has been briefly introduced in Subsection 3.5.2.

Structure of this chapter The new contributions in this chapter are given in the following sections. At first in Section 7.2 the overparametrization approach for the identification of Hammerstein systems using LS-SVMs pro-

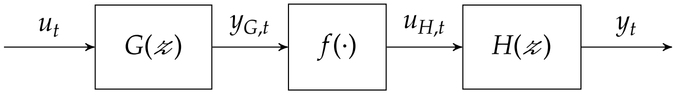

Figure 7.2: General structure of a Wiener-Hammerstein system with measured input signal u_t and output signal y_t. The intermediate signals $y_{G,t}$ and $u_{H,t}$ are assumed to be unobserved.

posed by Goethals is presented. Along with the basic technique, the necessary modifications for handling the Wiener-Hammerstein structure as an extended version of Hammerstein systems are outlined. Furthermore the projection step that is required to go from the overparametrized model back to the original model structure is revisited and an improved scheme is presented. The next section puts the focus on handling of large scale data sets using ideas presented in Section 7.3 that have already been proven to work well for NARX models. The approximation that is carried out to obtain a convex problem is based on dropping a rank constraint from the optimization problem that captures all the nonconvexity. In Section 7.4 the results from Chapter 6 are used to obtain an improved convex approximation based on the nuclear norm. Finally the proposed methods are studied on a large scale real life data set in Section 7.5. For the high quality measurements of a nonlinear electrical circuit, which constitutes the Wiener-Hammerstein benchmark [Schoukens et al., 2009] data set, the structured models introduced in this chapter are compared to unstructured NARX models. The chapter is then concluded in Section 7.6.

7.2 Exploiting information on the model structure

In this section it is demonstrated how to integrate structural information in a LS-SVM model. As outlined earlier it is based on the work of Goethals et al. [2005a, 2010] carried out for Hammerstein systems. However the key idea to obtain a convex problem formulation goes back to [Chang and Luus, 1971; Bai, 1998] which proposed overparametrization for the identification of Hammerstein systems.

This section first proposes a suitable parametrization of the Wiener-Hammerstein structure depicted in Figure 7.2. For this aim the linear blocks are represented by rational transfer functions and the static nonlinearity is described by a LS-SVM model. This parametrization gives rise to a nonconvex optimization problem. The next subsection simplifies the model so far that the overparametrization technique can be applied to it. This allows capturing all nonconvexity of the problem in a rank-1 constraint. Subsection 7.2.3 then derives a kernel based representation for a convex approximation of this problem. Approaches to recover a block structured model class from the approximation are discussed in the following subsection. The final subsection introduces a toy data set which will be used to provide numerical results in this as well as in later sections.

7.2.1 Model parametrization and nonlinear estimation problem

Assume that the linear blocks $G(\mathscr{z})$ and $H(\mathscr{z})$ can be described by rational transfer functions. Let the first block $G(\mathscr{z})$ have p_G poles and q_G zeros while the second linear block $H(\mathscr{z})$ has p_H poles and q_H zeros respectively. The signal $y_{G,t}$ at the output of the first block as depicted in Figure 7.2 is unobserved. The same holds true for $u_{H,t}$, the unobserved input of the linear system $H(\mathscr{z})$, shown in the same figure. Based on these definitions the output signals for both systems can be written as

$$y_{G,t} = \sum_{k=0}^{q_G} b_{G,k} u_{t-k} - \sum_{k=1}^{p_G} a_{G,k} y_{G,t-k}, \tag{7.1}$$

$$\hat{y}_t = \sum_{k=0}^{q_H} b_{H,k} u_{H,t-k} - \sum_{k=1}^{p_H} a_{H,k} y_{t-k}, \tag{7.2}$$

with parameters $\{a_{G,k}\}_{k=1}^{p_G}$ and $\{b_{G,k}\}_{k=0}^{q_G}$ for $G(\mathscr{z})$ and $\{a_{H,k}\}_{k=1}^{p_H}$ and $\{b_{H,k}\}_{k=0}^{q_H}$ for $H(\mathscr{z})$ respectively. The relation between the unobserved output $y_{G,t}$ of $G(\mathscr{z})$ and the unobserved input $u_{H,t}$ of $H(\mathscr{z})$ is given by the static nonlinearity, such that

$$u_{H,t} = f(y_{G,t}). \tag{7.3}$$

Choosing a LS-SVM structure as explained in Chapter 4 as parametrization for $f(\cdot)$, the nonlinearity can be expressed as

$$f(x) = \boldsymbol{w}^T \boldsymbol{\varphi}(x) + c. \tag{7.4}$$

Note that the regression variable x is scalar and accordingly $\boldsymbol{\varphi} : \mathbb{R} \to \mathbb{R}^{n_h}$. To allow compact formulations the following notation is introduced for the model parameters $\boldsymbol{a}_G = [a_{G,1}, \dots, a_{G,p_G}]^T \in \mathbb{R}^{p_G}$, $\boldsymbol{b}_G = [b_{G,0}, \dots, b_{G,q_G}]^T \in \mathbb{R}^{q_G+1}$, $\boldsymbol{a}_H = [a_{H,1}, \dots, a_{H,p_H}]^T \in \mathbb{R}^{p_H}$ and $\boldsymbol{b}_H = [b_{H,0}, \dots, b_{H,q_H}]^T \in \mathbb{R}^{q_H+1}$.

Furthermore matching vectors of lagged inputs and outputs are defined as $\boldsymbol{u}_t = [u_t, \dots, u_{t-q_G}]^T \in \mathbb{R}^{q_G+1}$, $\boldsymbol{y}_{t-1} = [y_{t-1}, \dots, y_{t-p_H}]^T \in \mathbb{R}^{p_H}$ and similarly for the unobserved intermediate signals $\boldsymbol{y}_{G,t-1} = [y_{G,t-1}, \dots, y_{G,t-p_G}]^T \in \mathbb{R}^{p_G}$ and $\boldsymbol{u}_{H,t} = [u_{H,t}, \dots, u_{H,t-q_H}]^T \in \mathbb{R}^{q_H+1}$. Then all parameters can be estimated from a nonlinear least squares problem

$$
\begin{aligned}
\min_{\boldsymbol{a}_G, \boldsymbol{b}_G, \boldsymbol{a}_H, \boldsymbol{b}_H, \boldsymbol{w}, c, y_{G,t}, u_{H,t}, e_t} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{t=D_H}^{N} e_t^2 \\
\text{subject to} \quad & y_t = \boldsymbol{b}_H^T\boldsymbol{u}_{H,t} - \boldsymbol{a}_H^T\boldsymbol{y}_{t-1} + e_t, \qquad t = D_H, \dots, N, \\
& u_{H,t} = \boldsymbol{w}^T\boldsymbol{\varphi}(y_{G,t}) + c, \qquad t = D_G, \dots, N, \\
& y_{G,t} = \boldsymbol{b}_G^T\boldsymbol{u}_t - \boldsymbol{a}_G^T\boldsymbol{y}_{G,t-1}, \qquad t = D_G, \dots, N, \\
& ||\boldsymbol{b}_G||_2 = 1, ||\boldsymbol{b}_H||_2 = 1,
\end{aligned} \tag{7.5}
$$

with $D_G = \max(p_G, q_G) + 1$ and $D_H = \max(p_H + 1, q_H + D_G)$. The last two scaling constraints remove an ambiguity, as a constant gain can be shuffled between the linear blocks $G(\not{z})$ and $H(\not{z})$ and the nonlinearity $f(\cdot)$ [Boyd and Chua, 1983, 1985].

7.2.2 Overparametrization of a simplified model

Problem (7.5) is strongly nonlinear and nonconvex, hence it has to be simplified to allow an efficient solution. The modified model will then enable a convex relaxation. The procedure operates in two steps to eliminate all references to the unobserved signals $y_{G,t}$ and $u_{H,t}$ from (7.5). Otherwise the optimization problem would contain products of the unknown signals with unknown model parameters resulting in nonconvex bilinear expressions.

1. In order to eliminate the dependency on $y_{G,t}$ one can relax the model structure from Wiener-Hammerstein to an extended Hammerstein model. In this structure the first linear block $G(\not{z})$ and the static nonlinearity $f(\cdot)$ are jointly modeled.
2. In a second step all remaining nonconvexity due to $u_{H,t}$ is captured in a rank constraint. This is achieved by introducing new variables for each product of $b_{H,k}$ with $\boldsymbol{w}$. To ensure that the number of free parameters stays constant, a rank-1 constraint is placed on a suitably defined matrix containing the new variables.

To carry out the first step, one is restricted to model parametrizations which only make use of the measured input signal u_t. Otherwise the unknown signal

$y_{G,t}$ would remain present and the problem nonconvex. Therefore a NFIR model structure is chosen to jointly model the first linear block $G(\mathcal{z})$ and the static nonlinearity $f(\cdot)$. Hence the static variable x in (7.4) is replaced by a vector of lagged inputs $\boldsymbol{u}_{f,t} = [u_t, \dots, u_{t-q_f}]^T \in \mathbb{R}^{q_f+1}$. Here q_f has to be chosen large enough to approximate the impulse response of $G(\mathcal{z})$. Then the joint model can be written as

$$u_{H,t} = f(\boldsymbol{u}_{f,t}) = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{u}_{f,t}) + c, \tag{7.6}$$

where now $\boldsymbol{\varphi} : \mathbb{R}^{q_f} \to \mathbb{R}^{n_h}$.

Note that the modified model still depends on $u_{H,t}$. Thus a second step is needed to also eliminate this signal from the modified model. This is achieved by substituting (7.6) into the model (7.2) for $H(\mathcal{z})$, yielding

$$y_t = \sum_{k=0}^{q_H} b_{H,k}(\boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{u}_{f,t-k}) + c) - \boldsymbol{a}_H^T \boldsymbol{y}_{t-1} + e_t. \tag{7.7}$$

Note that now the only remaining nonconvexity of the problem is due to the bilinear products of $b_{H,k}$ with $\boldsymbol{w}$ and c. The latter one is straightforwardly removed by introducing a new variable d with $d = c\sum_{k=0}^{q_H} b_{H,k}$. For the former let $\boldsymbol{\Phi}_t = [\boldsymbol{\varphi}(\boldsymbol{u}_{f,t}), \dots, \boldsymbol{\varphi}(\boldsymbol{u}_{f,t-q_H})]$, then the sum $\sum_{k=0}^{q_H} b_{H,k}\boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{u}_{f,t-k})$ can be rewritten as $\boldsymbol{b}_H^T \boldsymbol{\Phi}_t^T \boldsymbol{w}$. As this is scalar it can be replaced by $\operatorname{tr}(\boldsymbol{b}_H^T \boldsymbol{\Phi}_t^T \boldsymbol{w})$. Using the cyclic property of the trace this is also equivalent to $\operatorname{tr}(\boldsymbol{\Phi}^T \boldsymbol{w}\boldsymbol{b}_H^T)$. Now the rank-1 product $\boldsymbol{w}\boldsymbol{b}_H^T$ can be replaced by $\boldsymbol{W}$, a matrix of free variables and the additional constraint $\operatorname{rank}(\boldsymbol{W}) = 1$. This leads to a new optimization problem

$$\begin{aligned} &\min_{\boldsymbol{W},d,\boldsymbol{a}_H,e_t} \quad \frac{1}{2}\|\boldsymbol{W}\|_F^2 + \frac{1}{2}\gamma \sum_{t=D}^{N} e_t^2 \\ &\text{subject to} \\ &\quad y_t = \operatorname{tr}(\boldsymbol{\Phi}_t^T \boldsymbol{W}) + d - \boldsymbol{a}_H^T \boldsymbol{y}_{t-1} + e_t,\ t = D, \dots, N, \\ &\quad \operatorname{rank}(\boldsymbol{W}) = 1, \end{aligned} \tag{7.8}$$

with $D = \max(p_H, q_f + q_H) + 1$. The constraints are a reformulation of (7.7). This is also the case for the regularization term $\boldsymbol{w}^T\boldsymbol{w}$ of (7.5). Note that it can be rewritten as $\operatorname{tr}(\boldsymbol{w}\boldsymbol{w}^T)$. Due to the scaling constraints $\boldsymbol{b}_H^T\boldsymbol{b}_H = 1$, one can further write $\boldsymbol{w}^T\boldsymbol{w} = \operatorname{tr}(\boldsymbol{w}(\boldsymbol{b}_H^T\boldsymbol{b}_H)\boldsymbol{w}^T) = \operatorname{tr}(\boldsymbol{W}\boldsymbol{W}^T) = \|\boldsymbol{W}\|_F^2$.

7.2.3 Convex relaxation and dual model representation

Starting from the nonconvex optimization problem (7.5), based on a straightforward parametrization of a Wiener-Hammerstein system, an alternative optimization problem (7.8) for a simplified model has been derived. The only nonconvex term in this alternative formulation is the rank-1 constraint. Hence, a convex approximation of the problem can be obtained by dropping the rank constraint from the problem.

In [Goethals et al., 2005a, 2010] it has been noted that model performance can be improved by centering the nonlinear contributions $\boldsymbol{w}_k^T \boldsymbol{\Phi}_{(k)}$ for $k = 0, \dots, q_H$, where $\boldsymbol{W} = [\boldsymbol{w}_0, \dots, \boldsymbol{w}_{q_H}]$, $\boldsymbol{\Phi}_{(k)} = [\boldsymbol{\varphi}(\boldsymbol{u}_{f,D-q_H+k}), \dots, \boldsymbol{\varphi}(\boldsymbol{u}_{f,N-q_H+k})]$ and $\tilde{N} = N - D + 1$. This can be ensured by the introduction of new constraints $\boldsymbol{w}_k^T \boldsymbol{\Phi}_{(k)} \mathbf{1}_{\tilde{N}} = 0$ for $k = 0, \dots, q_H$. In view of Chapter 5 this ensures that the kernel based models are orthogonal to a constant which is already modeled by d. Not including the constraints might therefore result in an ambiguity.

Dropping the rank constraint and complementing the problem by these centering constraints yields a convex relaxation of (7.8) which is given by

$$
\begin{aligned}
\min_{\boldsymbol{W}, d, \boldsymbol{a}_H, e_t} \quad & \frac{1}{2} \sum_{k=0}^{q_h} \boldsymbol{w}_k^T \boldsymbol{w}_k + \frac{1}{2} \gamma \sum_{t=D}^{N} e_t^2 \\
\text{subject to} \quad & \boldsymbol{y} = \sum_{k=0}^{q_h} \boldsymbol{\Phi}_{(k)}^T \boldsymbol{w}_k + \mathbf{1}_{\tilde{N}} d - \boldsymbol{\Upsilon}^T \boldsymbol{a}_H + \boldsymbol{e}, \\
& \boldsymbol{w}_k^T \boldsymbol{\Phi}_{(k)} \mathbf{1}_{\tilde{N}} = 0, \quad k = 0, \dots, q_H,
\end{aligned}
\tag{7.9}
$$

where $\boldsymbol{y} = [y_D, \dots, y_N]^T$, $\boldsymbol{e} = [e_D, \dots, e_N]^T$ and $\boldsymbol{\Upsilon} = [\boldsymbol{y}_{D-1}, \dots, \boldsymbol{y}_{N-1}]$. Here the trace $\operatorname{tr}(\boldsymbol{\Phi}_t \boldsymbol{W})$ has once more been rewritten as summation and the squared Frobenius norm of $\boldsymbol{W}$ is expressed in terms of its columns as $\sum_{k=0}^{q_h} \boldsymbol{w}_k^T \boldsymbol{w}_k$.

Problem (7.9) is now in the form of a primal kernel based model like in Subsection 4.1.1. Treated in that way the primal formulation needs to be replaced by the dual, because in many cases an explicit and low dimensional expression of the feature map $\boldsymbol{\varphi}$ is not available. Therefore, in analogy to Lemma 4.2, the Lagrange dual containing only references to the known kernel K is derived.

Lemma 7.1. *Let $\boldsymbol{\alpha} \in \mathbb{R}^{\tilde{N}}$ denote the Lagrange multipliers for the first equality constraint of* (7.9) *and $\boldsymbol{\beta} = [\beta_0, \dots, \beta_{q_H}]$ denote the Lagrange multipliers for the centering constraints. Then the solution of* (7.9) *can be obtained in the dual from the linear system*

$$
\begin{bmatrix}
\boldsymbol{\Omega}_A + \frac{1}{\gamma} \boldsymbol{I}_{\tilde{N}} & \boldsymbol{\Omega}_C & \mathbf{1}_{\tilde{N}} & -\boldsymbol{\Upsilon}^T \\
\boldsymbol{\Omega}_C^T & \boldsymbol{\Omega}_D & \mathbf{0}_{(q_H+1)} & \mathbf{0}_{\boxtimes} \\
\mathbf{1}_{\tilde{N}}^T & \mathbf{0}_{(q_H+1)}^T & 0 & \mathbf{0}_{p_H}^T \\
-\boldsymbol{\Upsilon} & \mathbf{0}_{\boxtimes} & \mathbf{0}_{p_H} & \mathbf{0}_{\boxtimes}
\end{bmatrix}
\begin{bmatrix}
\boldsymbol{\alpha} \\ \boldsymbol{\beta} \\ d \\ \boldsymbol{a}_H
\end{bmatrix}
=
\begin{bmatrix}
\boldsymbol{y} \\ \mathbf{0}_{(q_H+1)} \\ 0 \\ \mathbf{0}_{p_H}
\end{bmatrix}
\tag{7.10}
$$

where $\boldsymbol{\Omega}_A = \sum_{k=0}^{q_h} \boldsymbol{\Omega}_k$, $\boldsymbol{\Omega}_C = [\boldsymbol{\Omega}_0 \mathbf{1}_{\tilde{N}}, \dots, \boldsymbol{\Omega}_{q_H} \mathbf{1}_{\tilde{N}}]$, $\boldsymbol{\Omega}_D = \mathrm{diag}(\mathbf{1}_{\tilde{N}}^T \boldsymbol{\Omega}_0 \mathbf{1}_{\tilde{N}}, \dots, \mathbf{1}_{\tilde{N}}^T \boldsymbol{\Omega}_{q_H} \mathbf{1}_{\tilde{N}})$ *and* $(\boldsymbol{\Omega}_k)_{ij} = K(\boldsymbol{u}_{f,N+(i-\tilde{N})+(k-q_H)}, \boldsymbol{u}_{f,N+(j-\tilde{N})+(k-q_H)})$ *for* $i, j = 1, \dots, \tilde{N}$.

Proof. The Lagrangian for (7.9) is given by

$$\begin{aligned}\mathscr{L}(\boldsymbol{w}_k, \boldsymbol{a}_H, d, \boldsymbol{e}, \boldsymbol{\alpha}, \beta_k) &= \sum_{k=0}^{q_h} \boldsymbol{w}_k^T \boldsymbol{w}_k + \frac{1}{2}\gamma \sum_{t=D}^{N} e_t^2 \\ &\quad - \boldsymbol{\alpha}^T \left(\sum_{k=0}^{q_h} \boldsymbol{\Phi}_{(k)}^T \boldsymbol{w}_k + \mathbf{1}_{\tilde{N}} d - \boldsymbol{Y}^T \boldsymbol{a}_H + \boldsymbol{e} - \boldsymbol{y} \right) - \sum_{k=0}^{q_H} \beta_k \boldsymbol{w}_k^T \boldsymbol{\Phi}_{(k)} \mathbf{1}_{\tilde{N}}.\end{aligned}$$

The KKT condition for $\boldsymbol{e}$ is the same as in the proof of Lemma 4.2. The conditions for optimality for the dual variables are the constraints of the primal problem. The remaining conditions for the primal variables are given by

$$\begin{aligned}\mathbf{0}_{n_h} &= \frac{\partial \mathscr{L}}{\partial \boldsymbol{w}_k} = \boldsymbol{w}_k - \boldsymbol{\Phi}_{(k)}(\boldsymbol{\alpha} + \beta_k \mathbf{1}_{\tilde{N}}), \quad k = 0, \dots, q_H \\ \mathbf{0}_{(q_H+1)} &= \frac{\partial \mathscr{L}}{\partial \boldsymbol{a}_H} = \boldsymbol{Y} \boldsymbol{\alpha}, \\ 0 &= \frac{\partial \mathscr{L}}{\partial d} = -\boldsymbol{\alpha}^T \mathbf{1}_{\tilde{N}}.\end{aligned}$$

Therefore the primal vectors $\boldsymbol{w}_k$ can be expressed in terms of the dual variables as $\boldsymbol{\Phi}_{(k)}(\boldsymbol{\alpha} + \beta_k \mathbf{1}_{\tilde{N}})$. This allows to write $\boldsymbol{\Phi}_{(k)}^T \boldsymbol{w}_k$ as $\boldsymbol{\Omega}_k \boldsymbol{\alpha} + \beta_k \boldsymbol{\Omega}_k \mathbf{1}_{\tilde{N}}$ where the kernel trick has been applied to $\boldsymbol{\Phi}_{(k)}^T \boldsymbol{\Phi}_{(k)}$ replacing it by the kernel matrix $\boldsymbol{\Omega}_k$. Substituting this relation into the KKT conditions for the dual variables, i.e. the constraints of the dual problem, and combining those with the conditions for $\boldsymbol{a}_H$ and d yields the dual system stated in (7.10). □

The corresponding model can be evaluated at a new point $(\boldsymbol{u}_{f,t}, \dots, \boldsymbol{u}_{f,t-q_H}, \boldsymbol{y}_{t-1})$ using the one-step-ahead predictive equation

$$\hat{y}_t = \sum_{k=0}^{q_H} \boldsymbol{k}_k(\boldsymbol{u}_{f,t-k})^T (\boldsymbol{\alpha} + \mathbf{1}_{\tilde{N}} \beta_k) + d - \boldsymbol{a}_H^T \boldsymbol{y}_{t-1}, \tag{7.11}$$

where $\boldsymbol{k}_k(\boldsymbol{z}) = [K(\boldsymbol{u}_{f,D+(k-q_H)}, \boldsymbol{z}), \dots, K(\boldsymbol{u}_{f,N+(k-q_H)}, \boldsymbol{z})]^T \in \mathbb{R}^{\tilde{N}}$.

7.2.4 Recovery of the original model class

Problem (7.8) is guaranteed to have a rank-1 solution such that the original parametrization in terms of $\boldsymbol{w}$ and $\boldsymbol{b}_H$ can be easily obtained by computing a rank-1 factorization of $\boldsymbol{W}$. However this is not the case for the relaxed problem (7.9). Therefore one has to resort to approximations. Explicitly recovering $\boldsymbol{w}$ from the matrix $\boldsymbol{W}$ is impossible if $\boldsymbol{\varphi}$ is only defined implicitly which is often the case. However if a factor $\boldsymbol{b}_H$ that corresponds to $\boldsymbol{W}$ can be found, then it can be fixed and used to estimate $\boldsymbol{w}$. This can be achieved using (7.7) as modeling constraint and the standard LS-SVM cost function $\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma\boldsymbol{e}^T\boldsymbol{e}$.

In the following two approaches to recover an estimate for $\boldsymbol{b}_H$ are discussed. The first one exploits only the kernel matrix while the second one is based on the analysis of model predictions.

1. Assume that $\boldsymbol{W}$ has the singular value decomposition $\boldsymbol{U\Sigma V}^T$. Then a possible approximation for $\boldsymbol{b}_H$ is given by the right singular vector corresponding to largest singular value as $\boldsymbol{W}$ is the relaxation of the rank-1 product $\boldsymbol{w}\boldsymbol{b}_H^T$. Note that the matrix $\boldsymbol{V}$ containing the right singular vectors is finite dimensional and can be obtained from the eigenvalue decomposition of $\boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{V\Sigma}^2\boldsymbol{V}^T$. This matrix can be computed as
$$\boldsymbol{W}^T\boldsymbol{W} = \begin{bmatrix} \boldsymbol{\alpha}_0^T\boldsymbol{\Omega}_{0,0}\boldsymbol{\alpha}_0 & \cdots & \boldsymbol{\alpha}_0^T\boldsymbol{\Omega}_{0,q_H}\boldsymbol{\alpha}_{q_H} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\alpha}_{q_H}^T\boldsymbol{\Omega}_{q_H,0}\boldsymbol{\alpha}_0 & \cdots & \boldsymbol{\alpha}_{q_H}^T\boldsymbol{\Omega}_{q_H,q_H}\boldsymbol{\alpha}_{q_H} \end{bmatrix} \tag{7.12}$$
where $\boldsymbol{\alpha}_k = \boldsymbol{\alpha} + \mathbf{1}_{\tilde{N}}\beta_k$, $\boldsymbol{\Omega}_{k,l} = \boldsymbol{\Phi}_{(k)}^T\boldsymbol{\Phi}_{(l)}$ and $(\boldsymbol{\Omega}_{k,l})_{ij} = K(\boldsymbol{u}_{f,N+(i-\tilde{N})+(k-q_H)}, \boldsymbol{u}_{f,N+(j-\tilde{N})+(l-q_H)})$. This follows from expressing the ij-th element of $\boldsymbol{W}^T\boldsymbol{W}$ as $\boldsymbol{w}_i^T\boldsymbol{w}_j$ and the expansion of $\boldsymbol{w}_k$ in terms of the dual variables as given in the proof of Lemma 7.1.
2. Instead of working with $\boldsymbol{W}$ one can also analyze the predictions $\boldsymbol{w}_k^T \boldsymbol{\varphi}(\boldsymbol{u}_{f,t})$ as done by Goethals et al. [2005a, 2010]. Here one has to notice that $\boldsymbol{w}_k$ is a relaxation of $b_k\boldsymbol{w}$. Hence, the vector of predictions generated by the components of the model $\hat{\boldsymbol{y}}_{C,t} = [\boldsymbol{w}_0^T\boldsymbol{\varphi}(\boldsymbol{u}_{f,t}), \dots, \boldsymbol{w}_{q_H}^T\boldsymbol{\varphi}(\boldsymbol{u}_{f,t})]$ should be equal to $\boldsymbol{b}_H^T$ scaled by $\boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{u}_{f,t})$. This has to hold for all t of the training data set. As only finite data is available and all components have to be evaluated, this can only be done for $t = D, \dots, N - q_H$. If this is carried out for all available data, the matrix $\boldsymbol{Y}_C = [\hat{\boldsymbol{y}}_D, \dots, \hat{\boldsymbol{y}}_{N-q_H}]^T$ can be formed. As each row should be a scaled version of $\boldsymbol{b}_H$, once more the dominant right singular vector of $\boldsymbol{Y}_C$ is an estimate for $\boldsymbol{b}_H$.

Here in both approaches only the dominant singular vector has been used. Hence, all information contained in the remaining singular vectors is neglected. Instead of solely relying on the dominant information one can also consider a randomized approach that is able to take all information into account. Therefore a vector $\boldsymbol{v}$ is drawn from a zero mean Gaussian distribution with covariance matrix $\boldsymbol{V}\boldsymbol{\Sigma}^2\boldsymbol{V}^T$. Then one can generate estimates for $\boldsymbol{b}_H$ by choosing $\boldsymbol{b}_H = \boldsymbol{v}/\|\boldsymbol{v}\|_2$. These are then candidate values which have to be tested for their applicability on the data.

Below a complete algorithm is summarized to estimate an overparametrized LS-SVM Wiener-Hammerstein model and project it back onto the original model class.

Algorithm 7.1 (Estimation of structured LS-SVM model)**.**

1. Choose model orders p_H, q_H and q_f.
2. Select a regularization parameter γ and a kernel function K (and its parameters).
3. Compute the kernel matrices $\boldsymbol{\Omega}_k$, its average $\boldsymbol{\Omega}_A$, sums $\boldsymbol{\Omega}_D$ and row sums $\boldsymbol{\Omega}_C$ as defined in Lemma 7.1.
4. Solve the dual linear system (7.10).

Projection onto original model class:

5. Obtain estimate for $\boldsymbol{b}_H$ by one of the techniques described in Subsection 7.2.4.
6. Estimate a model with $\min_{w,c,a_H,e} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma\boldsymbol{e}^T\boldsymbol{e}$ subject to (7.7) for $t = D, \dots, N$.

7.2.5 Numerical example

In the following an artificial data set is used to evaluate the advantages of using the proposed structured modeling approach over an unstructured NARX type model. The compared model classes are

NARX LS-SVM model with NARX structure,
STRCTRD-1 structured model with projection based on $\boldsymbol{W}$ (method 1),
STRCTRD-2 structured model with projection based on component-wise predictions (method 2) and
OVERPRZD overparametrized model, i.e. model without projection.

The corresponding model equations and algorithms are referenced in Table 7.1. The example system used for the evaluation is defined in the following.

System 7.1 (Wiener-Hammerstein system)**.** The first linear block $G(\mathfrak{z})$ is given by a fourth order Chebychev type II digital low-pass filter with a stopband edge frequency 0.5 (normalized) and a stopband ripple of 40 dB. Its

Table 7.1: List of considered model structures, the corresponding one-step-ahead predictors and the algorithms needed for their estimation.

Model	Model equation	Algorithm
NARX	(4.6)	4.1
STRCTRD-1	(7.7)†	7.1 with projection 1
STRCTRD-2	(7.7)†	7.1 with projection 2
OVRPRZD	(7.11)	7.1 without projection

†with $\boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{u}_{f,t-k}) = \sum_{l=0}^{P} b_{H,l} \sum_{n=1}^{\tilde{N}} \alpha_n K(\boldsymbol{u}_{f,t-k}, \boldsymbol{u}_{f,n-l})$.

transfer function is given by

$$G(z) = \frac{0.0458 + 0.0755z^{-1} + 0.1024z^{-2} + 0.0754z^{-3} + 0.0458z^{-4}}{1 - 1.5233z^{-1} + 1.2537z^{-2} - 0.4602z^{-3} + 0.0747z^{-4}},$$

where all coefficients have been truncated to four significant digits. The static nonlinearity is a hyperbolic tangent, $f(\cdot) = \tanh(\cdot)$, representing a mildly nonlinear saturation characteristic. The second linear block is given by a 6th order comb filter with transfer function

$$H(z) = \frac{\prod_{k=1}^{3}(z^2 - 2m_{\zeta,k}\cos(\zeta_k)z + m_{\zeta,k}^2)}{\prod_{k=1}^{3}(z^2 - 2m_{\xi,k}\cos(\xi_k)z + m_{\xi,k}^2)}$$

with $\zeta_k = \frac{2}{30}(2k)\pi$, $\xi_k = \frac{2}{30}(2k-1)\pi$ and $m_{\zeta,k} = (0.9, 0.9, 0.8)$, $m_{\xi,k} = (0.7, 0.9, 0.9)$. The frequency response of $H(z)$ is also shown in Figure 7.3.

The input data is generated from a standard normal distribution and the output is corrupted with additive white Gaussian noise having a standard deviation of 0.01. To suppress transient effects the first 500 samples are always discarded.

Unless noted otherwise a training set of size 1000 is used to estimate all models. As kernel the RBF kernel is used throughout this chapter. The bandwidth parameter σ of the kernel and the regularization parameter γ are selected such that the RMSE for one-step-ahead predictions on a validation set with 1000 samples is minimized. The candidate values for σ and γ are taken from a grid. The performance is always evaluated on an independent test data set of 1000 samples.

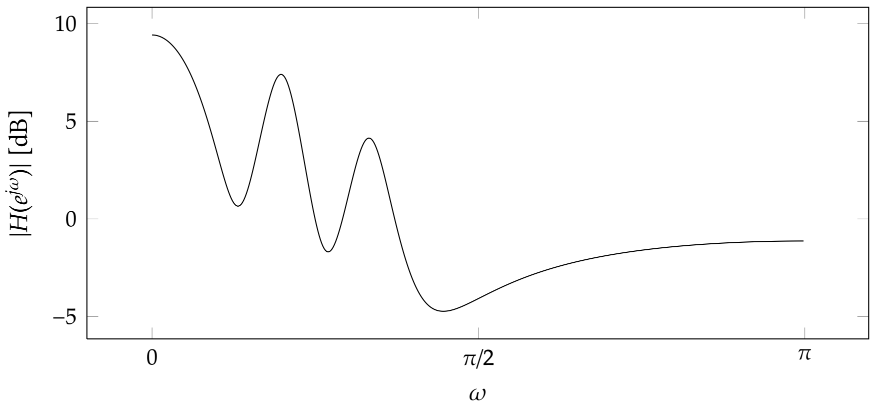

Figure 7.3: Transfer function of the second linear block in the Wiener-Hammerstein structure of System 7.1.

Model selection

The NARX models as well as the structured models introduced in this section have several parameters specifying their model order. In case of NARX models these are p and q while for the structured models q_f, p_H and q_H have to be selected. To keep the computational burden small, the parameters are chosen such that $p = q$ and $p_H = q_H$. The optimal model order is selected by estimating models for different model orders and choosing the one with lowest RMSE for one-step-ahead predictions on an independent test set. Carrying this out for the NARX structure yields a model order of $p = q = 12$ as can be seen from Figure 7.4. In case of the structured models, model order selection is performed based on OVRPRZD, the model without projection. The obtained model orders are $p_H = q_H = 15$ and $q_f = 5$ which follow from the performances shown in Figure 7.5.

Comparison of projection schemes

In Subsection 7.2.4 two projection schemes were introduced to obtain a model within the original model class. Based on the system described at the beginning of this section, 100 realizations of data are generated and used to estimate models with the optimal model orders as obtained in the last subsection. For each realization both projection schemes are used to obtain models. These are then evaluated on an independent test set. The used evaluation criterion is the root mean squared error for one-step-ahead predictions as well as for

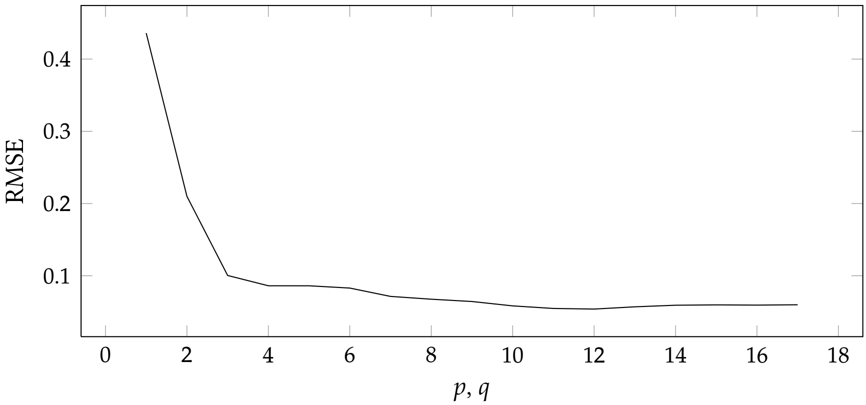

Figure 7.4: Model order selection for NARX model of Wiener-Hammerstein System 7.1. The RMSE is computed for one-step-ahead predictions on an independent test set.

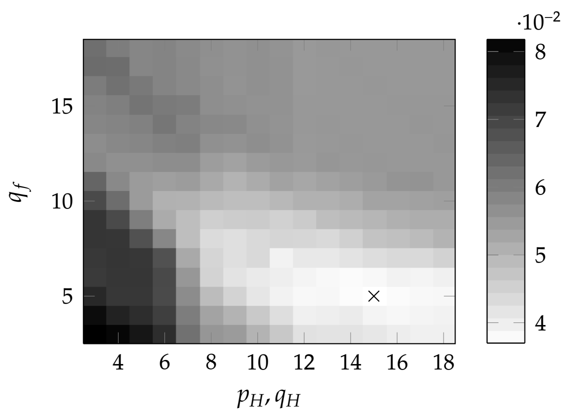

Figure 7.5: Model selection for Wiener-Hammerstein System 7.1 based on OVRPRZD. The RMSE for one-step-ahead prediction on an independent test set is shown. The cross marks the optimal model order $p_H = q_H = 15$ and $q_f = 5$.

Table 7.2: Root mean squared error on an independent test set for System 7.1 comparing the different projection schemes and the unprojected model. The training set size is 1000 samples. All shown values are scaled by 100. In total 100 realizations of the data are generated and the reported values correspond to mean and standard deviation.

MODEL/ESTIMATION	ONE-STEP-AHEAD	SIMULATION
OVRPRZD	3.86 (0.23)	7.38 (1.37)
STRCTRD-1	4.65 (0.35)	9.45 (1.10)
STRCTRD-2	4.95 (0.45)	9.93 (1.85)

fully simulated values. The performances are summarized in Table 7.2. It can be concluded that both projection schemes result in a degradation of predictive performance when compared to the unprojected model. However, STRCTRD-1 does slightly better than STRCTRD-2.

Besides the predictive performance the projection also gives insight into the quality of the convex approximation. Given a perfect solution the matrices $\boldsymbol{W}$ and $\boldsymbol{Y}$ would be of rank one, i.e. with all energy concentrated in the largest eigenvalue. For the projection based on $\boldsymbol{W}$ 60% (± 2%) of the energy is concentrated in the largest eigenvalue. In case of working with the matrix of predictions $\boldsymbol{Y}_C$ 85% (± 8%) of the energy is concentrated in the largest singular value. From the values based on $\boldsymbol{W}$ it can be clearly seen that the approximation is far from the optimal rank-1 solution.

The structured models allow the extraction of the parameters $\boldsymbol{a}_H$ and $\boldsymbol{b}_H$ of the second linear block $H(z)$. Based on the 100 data realizations the covariance matrix of these parameters can be computed for both projection schemes as well as their average values. For ease of comparison the transfer function of $H(z)$ is visualized in Figure 7.6 in the frequency domain. While both methods are not able to identify the true transfer function correctly, STRCTRD-1 does a little better on average. However the variance of the parameters for STRCTRD-2 is so large that it cannot be visualized, while for STRCTRD-1 the confidence regions are very small.

Behavior for different number of support vectors

To study the advantages of the structured models over their unstructured counterparts, models with different numbers of support vectors have been estimated. The considered training set sizes are 100, 250, 500, 1000 and 2500.

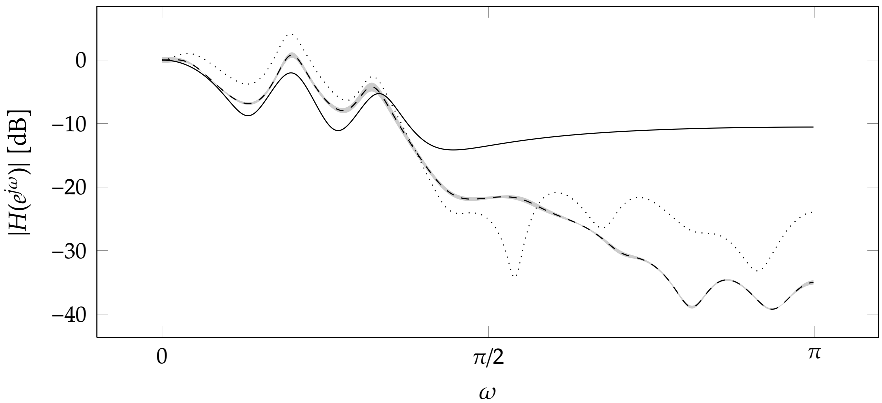

Figure 7.6: Estimated transfer function of the second linear block $H(z)$ of System 7.1. The solid line is true underlying system, the dashed and the dotted lines are obtained by STRCTRD-1 and STRCTRD-2 respectively. The shown transfer functions are computed from the mean of the parameters estimated from 100 different realizations of the data. All transfer functions are scaled to unit gain. The gray area indicates 2.45 standard deviations of the parameter estimates transferred to frequency domain. For STRCTRD-2 the parameter variance is too large to be visualized.

Note that in the dual formulation each training sample corresponds to a support vector. The results are summarized in Table 7.3.

It can be concluded that the one-step-ahead predictions are in general quite good. The prediction error in simulation mode is roughly twice as big as for the one-step-ahead predictions independent of the considered model structure and the number of support vectors. While for sample sizes up to 500 SVs the model structure STRCTRD-1 outperforms all other models, the advantage shifts to the OVRPRZD models for sample sizes of at least 1000 SVs.

7.3 Handling of large data sets

Standard LS-SVMs for NARX models can be extended to handle large data sets as described in Section 4.3. These ideas can also be extended to the structured models described in this chapter. One possibility is to use the fixed-size approach which has the advantage that no modifications have to be made to the problem, but the disadvantage that it cannot handle as much data as in

Table 7.3: Root mean squared error on an independent test set for System 7.1 comparing the different model structures for varying numbers of support vectors. All shown values are scaled by 100. In total 100 realizations of the data were generated and the reported values correspond their mean. The first value is for predictions generated in simulation mode while the value in parenthesis corresponds to predictive performance for one-step-ahead predictions.

NUMBER OF SVs	NARX	OVRPRZD	STRCTRD-1
100	17.75 (8.71)	19.22 (7.32)	16.96 (6.76)
250	15.31 (7.03)	14.63 (6.24)	12.68 (5.83)
500	15.12 (6.60)	10.14 (5.14)	9.92 (4.98)
1000	13.69 (5.69)	7.27 (3.82)	9.22 (4.57)
2500	9.80 (4.20)	5.24 (3.15)	8.06 (4.41)

the unstructured case. The second approach is also based on the Nyström approximation, but still solves the problem in the dual domain. This way it is possible to handle as much data as for the unstructured models. However, incorporating the centering constraints and a reconstruction of the original model class as described in the previous section is impossible.

7.3.1 A fixed-size structured model

Using the approximate feature map $\hat{\varphi}$ from Subsection 4.3 to form the matrices $\boldsymbol{\Phi}_{(k)}$ one can state the following corollary.

Corollary 7.2. *Let* $\boldsymbol{\Phi}_C = [\boldsymbol{\Phi}_{(0)}^T, \dots, \boldsymbol{\Phi}_{(q_H)}^T]^T$, *and* $\boldsymbol{\Phi}_S$ *be a block matrix with* $(\boldsymbol{\Phi}_S)_{kk} = \boldsymbol{\Phi}_{(k)}\mathbf{1}_{\tilde{N}} \in \mathbb{R}^{M\times 1}$ *for* $k = 0, \dots, q_H$ *and all other blocks zero. Also define* $\boldsymbol{S} = [\boldsymbol{\Phi}_C^T, -\boldsymbol{Y}^T, \mathbf{1}_{\tilde{N}}]$, $\boldsymbol{\omega} = [\boldsymbol{w}_0^T, \dots, \boldsymbol{w}_{q_H}^T, \boldsymbol{a}_H^T, d]$ *and* $\boldsymbol{P} = [\boldsymbol{\Phi}_S, \mathbf{0}_{\boxtimes}]^T$ *where the block of zeros is of dimension* $(q_H + 1) \times (p_H + 1)$*. Then the extension of* (4.19) *to the structured model* (7.9) *is given by*

$$\begin{bmatrix} \boldsymbol{S}^T\boldsymbol{S} + \gamma^{-1}\boldsymbol{I}_{M,q_H,0} & \boldsymbol{P} \\ \boldsymbol{P}^T & \mathbf{0}_{\boxtimes} \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{\beta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{S}^T\boldsymbol{y} \\ \mathbf{0}_{(q_H+1)} \end{bmatrix} \tag{7.13}$$

where $\boldsymbol{\beta}$ *are the Lagrange multipliers for the centering constraints and* $\boldsymbol{I}_{M,q_H,0}$ *is a diagonal matrix of dimension* $(M \cdot q_H + p_H + 1) \times (M \cdot q_H + p_H + 1)$ *whose first* $M \cdot q_H$ *elements are one and all others are zero.*

The one-step-ahead predictor for this overparametrized form is given by

$$\hat{y}_t = \mathrm{tr}(\boldsymbol{\Phi}_t^T \boldsymbol{W}) + \boldsymbol{a}_H^T \boldsymbol{y}_{t-1} + d, \tag{7.14}$$

where $\boldsymbol{W} = [\boldsymbol{w}_0, \dots, \boldsymbol{w}_{q_H}]$.

Whereas an unstructured model (4.1) can be solved from a linear system (4.19) in $M + 1$ variables, the structured model (7.9) corresponds to a linear system (7.13) in $(M + 1) \cdot q_H + p_H + 1$ variables. Therefore even for a subsample with $M \ll \tilde{N}$, the solution can be computationally infeasible. The projection onto the original model class is straightforward as the matrix $\boldsymbol{W}$ is finite dimensional and explicitly known. Then the estimation procedure is a combination of Algorithms 4.3 & 7.1.

Algorithm 7.2 (Estimation of structured fixed-size model)**.**

1. Choose model orders p_H, q_H and q_f.
2. Select a subset $S_M \subset \{\boldsymbol{u}_{f,t}\}_{t=q_f+1}^{N}$ of $M \ll \tilde{N}$ data points from the full data set.
3. Select a regularization parameter γ and a kernel function K (and its parameters).
4. Build the kernel matrix $\boldsymbol{\Omega}_M$ evaluated on S_M and compute its eigendecomposition $\boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^T$.
5. Use (4.16) to form $\boldsymbol{\Phi}_k$ by evaluating $\hat{\boldsymbol{\varphi}}$ for all $\boldsymbol{u}_{f,t}$.
6. Solve the primal linear system (7.13).
7. Obtain estimates for $\boldsymbol{w}$ and $\boldsymbol{b}_H$ from the dominant singular vectors of $\boldsymbol{W}$.
8. *Optional:* Fix $\boldsymbol{b}_H$ and estimate a refined model with $\min_{\boldsymbol{w},c,\boldsymbol{a}_H,\boldsymbol{e}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma\boldsymbol{e}^T\boldsymbol{e}$ subject to (7.7) for $t = D, \dots, N$.

7.3.2 A large-scale overparametrized model

An alternative approach is to use the original idea of Williams and Seeger [2001] to compute an approximate low rank factorization of the kernel matrix as described in Subsection 4.3.2. Therefore a new kernel function has to be defined summing up the individual contributions. This is only feasible without considering the centering constraints in (7.9) as these act on the individual submodels $\boldsymbol{w}_k^T\boldsymbol{\varphi}(\cdot)$. Therefore they are dropped from the problem.

Lemma 7.3. *Define a new kernel* $K_A\left((\boldsymbol{u}_{f,t}, \dots, \boldsymbol{u}_{f,t-q_f}), (\boldsymbol{u}_{f,n}, \dots, \boldsymbol{u}_{f,n-q_f})\right) = \sum_{k=0}^{p} K(\boldsymbol{u}_{f,t-k}, \boldsymbol{u}_{f,n-k})$*. Let* $\boldsymbol{\Omega}_{A,M}$ *denote the kernel matrix on a subsample of* *size M such that $(\boldsymbol{\Omega}_{A,M})_{ij} = K_A(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $i, j = 1, \dots, M$. Then the kernel matrix $\boldsymbol{\Omega}_A$ has the approximate low rank factorization*

$$\boldsymbol{\Omega}_A \simeq \boldsymbol{\Omega}_{A,M\tilde{N}}^T \boldsymbol{\Omega}_{A,M}^{-1} \boldsymbol{\Omega}_{A,M\tilde{N}}, \tag{7.15}$$

where $(\boldsymbol{\Omega}_{A,MN})_{ij} = K_A(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $i = 1, \dots, M$ and $j = 1, \dots, \tilde{N}$.

Then (7.10) can be solved efficiently by exploiting this low rank factorization and the matrix inversion lemma.

Corollary 7.4. *Let $\boldsymbol{\Omega}_{A,M}$ have a factorization such that $\boldsymbol{\Omega}_{A,M} = \boldsymbol{G}^T\boldsymbol{G}$ and define $\boldsymbol{F} = \boldsymbol{G}^{-1}\boldsymbol{\Omega}_{A,M\tilde{N}}$. Then an approximate solution of* (7.10) *without the centering constraints is given in terms of the dual variables as*

$$\boldsymbol{\alpha} \simeq \boldsymbol{A}^{-1}(\boldsymbol{y} - \boldsymbol{Y}\boldsymbol{a}_H - \boldsymbol{1}_{\tilde{N}} d), \tag{7.16a}$$

$$\begin{bmatrix} \boldsymbol{a}_H \\ d \end{bmatrix} \simeq \boldsymbol{C}^{-1}[-\boldsymbol{Y}^T, \boldsymbol{1}_{\tilde{N}}]^T \boldsymbol{A}^{-1}\boldsymbol{y} \tag{7.16b}$$

where $\boldsymbol{A}^{-1} \simeq \gamma \boldsymbol{I}_{\tilde{N}} - \gamma \boldsymbol{F}^T(\gamma^{-1}\boldsymbol{I}_M + \boldsymbol{F}\boldsymbol{F}^T)^{-1}\boldsymbol{F}$ and $\boldsymbol{C} = [-\boldsymbol{Y}^T, \boldsymbol{1}_{\tilde{N}}]^T \boldsymbol{A}^{-1}[-\boldsymbol{Y}^T, \boldsymbol{1}_{\tilde{N}}]$.

Proof. Without the centering constraints (7.10) can be written as

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^T & \boldsymbol{0}_{\boxtimes} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\theta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{0}_{(p_H+1)} \end{bmatrix} \tag{7.17}$$

where $\boldsymbol{A} = \boldsymbol{\Omega}_A + \gamma^{-1}\boldsymbol{I}_{\tilde{N}}$, $\boldsymbol{B} = [-\boldsymbol{Y}^T, \boldsymbol{1}_{\tilde{N}}]$ and $\boldsymbol{\theta} = [\boldsymbol{a}_H^T, d]^T$. Solving the system in block form one obtains $\boldsymbol{\alpha} = \boldsymbol{A}^{-1}(\boldsymbol{y} - \boldsymbol{B}\boldsymbol{\theta})$ and $\boldsymbol{\theta} = \boldsymbol{C}^{-1}\boldsymbol{B}^T\boldsymbol{A}^{-1}\boldsymbol{y}$ with $\boldsymbol{C} = \boldsymbol{B}^T\boldsymbol{A}^{-1}\boldsymbol{B}$. Using the Nyström approximation of $\boldsymbol{\Omega}_A$ the matrix $\boldsymbol{A}$ can be approximated as $\gamma^{-1}\boldsymbol{I}_{\tilde{N}} + \boldsymbol{F}^T\boldsymbol{F}$. The expression for $\boldsymbol{A}^{-1}$ is a result from this approximation and the matrix inversion lemma exploiting its low rank factorization. □

Note that the most expensive operation is to solve linear systems in M variables. Therefore this approach scales as good as the fixed-size approach for the unstructured models. Finally the one-step-ahead predictor is given in terms of the new kernel function K_A

$$\hat{y}_t = \boldsymbol{\alpha}^T \boldsymbol{k}_A((\boldsymbol{u}_{f,t}, \dots, \boldsymbol{u}_{f,t-q_H})) + d - \boldsymbol{a}_H^T \boldsymbol{y}_{t-1}, \tag{7.18}$$

where $\boldsymbol{k}_A(\boldsymbol{z}) = [K_A(\boldsymbol{x}_{f,D}, \boldsymbol{z}), \dots, K(\boldsymbol{x}_{f,N}, \boldsymbol{z})]^T \in \mathbb{R}^{\tilde{N}}$ and $\boldsymbol{x}_{f,t} = (\boldsymbol{u}_{f,t}, \dots, \boldsymbol{u}_{f,t-q_H})$. The drawback of this approach is that from the modified kernel function K_A, the product $\boldsymbol{W}^T\boldsymbol{W}$ cannot be computed. Therefore it is impossible to project the model back onto the original model class.

Table 7.4: List of considered model structures, the corresponding one-step-ahead predictors and the algorithms needed for their estimation.

Model	Model equation	Algorithm
FS-NARX	(4.20)	4.3
FS-STRCTRD	(7.7)	7.2
FS-OVRPRZD	(7.14)	7.2 without projection
LS-OVRPRZD	(7.18)	7.3

Algorithm 7.3 (Estimation of large-scale overparametrized model)**.**

1. Choose model orders p_H, q_H and q_f.
2. Select a subset $S_M \subset \{\boldsymbol{x}_{f,t}\}_{t=D}^{N}$ of $M \ll \tilde{N}$ data points from the data set.
3. Select a regularization parameter γ and a kernel function K (and its parameters).
4. Build the kernel matrix $\boldsymbol{\Omega}_{A,M}$ evaluated on S_M and compute for example its Cholesky decomposition $\boldsymbol{G}^T\boldsymbol{G}$. Also compute $\boldsymbol{\Omega}_{A,M\tilde{N}}$.
5. Solve the dual linear system (7.10) (without centering constraints, i.e. (7.17)) via (7.16).

7.3.3 Numerical example

This subsection complements Subsection 7.2.5 with numerical examples for large scale data sets. Therefore the same system is considered but the size of the training set is increased to 10,000. Based on this, the following large scale model structures are compared with each other:

FS-NARX fixed-size LS-SVM model with NARX structure,
FS-STRCTRD fixed-size structured model,
FS-OVRPRZD fixed-size overparametrized model and
LS-OVRPRZD large-scale overparametrized model.

For reference the model equations and estimation algorithms are summarized in Table 7.4. Due to the computational complexity that scales with q_H, the model order is adapted. The optimal model order according to Figure 7.5 is $p_H = q_H = 15$. Yet the degradation when choosing $p_H = q_H = 12$ is quite small, therefore it is chosen to reduce the computational complexity. The values for q_f and the model orders p, q of the NARX structure are kept fixed. Model performances are evaluated for different numbers of support vectors, namely

Table 7.5: Root mean squared error on an independent test set for System 7.1 comparing the different model structures for varying numbers of support vectors. All shown values are scaled by 100. In total 100 realizations of the data were generated and the reported values correspond to their mean. The first value is for predictions generated in simulation mode while the value in parenthesis corresponds to predictive performance for one-step-ahead predictions.

Model	100 SVs	250 SVs	500 SVs	1000 SVs
FS-NARX	13.64 (5.86)	12.82 (5.78)	12.28 (4.99)	7.86 (3.44)
FS-STRCTRD	6.92 (4.29)	7.55 (4.40)	7.44 (4.35)	7.22 (4.40)
FS-OVRPRZD	4.63 (3.05)	4.70 (3.08)	4.67 (3.01)	4.52 (3.04)
LS-OVRPRZD	11.64 (5.56)	8.86 (4.61)	5.24 (3.22)	4.75 (3.13)

100, 250, 500 and 1000. The results are reported in Table 7.5. In general the fixed size overparametrized model yields the best performance regardless of the number of support vectors. For small amounts of support vectors the fixed size structured model is also better than the remaining ones. With increasing numbers of support vectors the large-scale overparametrized model gains advantage.

7.4 Improved convex relaxation based on nuclear norms

In the next subsection a nuclear norm based convex relaxation is introduced in a parametric setting. The following subsection considers the corresponding kernel based model. Finally some numerical examples are studied using the parametric formulation in the last subsection.

7.4.1 Parametric approach based on the fixed size formulation

In Subsection 7.2.2 it has been argued that in the transition from $\boldsymbol{w}$ to $\boldsymbol{W}$ the regularization term $\boldsymbol{w}^T\boldsymbol{w}$ can be replaced by $||\boldsymbol{W}||_F^2$ without changing the problem. Actually the same holds for some other matrix norms as well. In particular consider the nuclear norm which can be computed as the sum of the singular values. Assuming that $\boldsymbol{W}$ is a rank-1 matrix there is only a single nonzero singular value. Multiplying $\boldsymbol{W}$ from the right with $\boldsymbol{b}_H$ one obtains

$\boldsymbol{W}\boldsymbol{b}_H = \boldsymbol{w}\boldsymbol{b}_H^T\boldsymbol{b}_H = \boldsymbol{w}$ as $\boldsymbol{b}_H$ was chosen to have unit norm to avoid modeling ambiguities. Therefore the single singular value of $\boldsymbol{W}$ is $\|\boldsymbol{w}\|_2$ and hence $\|\boldsymbol{W}\|_* = \|\boldsymbol{w}\|_2$. Squaring both sides of the equation then yields the desired equivalence $\|\boldsymbol{W}\|_*^2 = \boldsymbol{w}^T\boldsymbol{w}$. Hence, the Frobenius norm, in the nonconvex problem (7.8) including the rank constraint, can be replaced by the nuclear norm. Now by dropping the rank constraint a tighter convex relaxation is obtained due to the property of the nuclear norm being the convex envelope of the rank function. For algorithmic reasons it is easier to work with unsquared norms than their squared counterparts. Therefore the convex relaxation based on the nuclear norm will drop the square, however as shown in Lemma 9.6 for a similar problem, both forms can be transformed into each other by scaling the regularization term.

Proposition 7.5. *An improved convex relaxation for* (7.8) *is given by*

$$\begin{aligned} \min_{\boldsymbol{W},d,\boldsymbol{a}_H,e_t} \quad & \eta\|\boldsymbol{W}\|_* + \frac{1}{2}\sum_{t=D}^{N} e_t^2 \\ \text{subject to} \quad & \boldsymbol{y} = \sum_{k=0}^{q_h} \boldsymbol{\Phi}_{(k)}^T\boldsymbol{w}_k + \mathbf{1}_{\tilde{N}}d - \boldsymbol{Y}^T\boldsymbol{a}_H + \boldsymbol{e}. \end{aligned} \tag{7.19}$$

In a purely parametric setting one could choose any set of basis functions to define the feature map $\boldsymbol{\varphi}$. In kernel based setting the most natural choice for the basis functions is to follow Subsection 7.3.1 and extract them from a fixed size primal approximation of the kernel function.

7.4.2 Kernel based approach

As shown in Chapter 6 it is possible to formulate kernel based models based on nuclear norms. Applying those results to the Hammerstein identification problem in (7.19) yields the following formulation.

Lemma 7.6. *Let* $\bar{N} = N - q_f + 1$ *and define the kernel matrix* $\boldsymbol{\Omega}$ *as* $\Omega_{ij} = K(\boldsymbol{u}_{f,N+(i-\bar{N})}, \boldsymbol{u}_{f,N+(j-\bar{N})})$ *for* $i,j = 1,\dots,\bar{N}$. *Furthermore let the kernel matrix have a factorization such that* $\boldsymbol{\Omega} = \boldsymbol{G}^T\boldsymbol{G}$. *Then the kernel based model for* (7.19) *is*

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & -\boldsymbol{\alpha}^T\boldsymbol{y} - \frac{1}{2}\boldsymbol{\alpha}^T\boldsymbol{\alpha} \\ \text{subject to} \quad & \mathbf{1}_{\tilde{N}}^T\boldsymbol{\alpha} = 0, \quad \boldsymbol{Y}\boldsymbol{\alpha} = \mathbf{0}_{p_H}, \quad \|\boldsymbol{G}\mathcal{B}(\boldsymbol{\alpha})\|_2 \le \eta, \end{aligned} \tag{7.20}$$

with $\boldsymbol{\alpha} \in \mathbb{R}^{\tilde{N}}$ *and* $\tilde{N}$, $\boldsymbol{y}$ *defined as before in Subsection 7.2.3. The operator* $\mathcal{B}$ *is defined as in Lemma 6.18.*

Proof. Only one aspect has to be considered in this proof as this lemma is a minor extension of Lemma 6.18. For the remainder the proof of Lemma 6.18 holds without modification. As only additional element the KKT conditions have to be extended for the variable $\boldsymbol{a}_H$,

$$\frac{\partial \mathscr{L}}{\partial \boldsymbol{a}_H} = 0 : \quad \boldsymbol{Y}\boldsymbol{\alpha} = \boldsymbol{0}_{p_H}.$$

The inclusion of this additional constraint in the dual problem concludes the derivation. □

Based upon the results derived in Subsection 6.6.2, a one-step-ahead predictive equation for a new point can be stated. Whereas Corollary 6.21 can be used unaltered, the necessary modifications to Corollary 6.20 are carried out in the following derived result.

Corollary 7.7. *The matrix $\boldsymbol{H}_\eta$ introduced in Corollary 6.9, as well as the primal variables $\boldsymbol{a}_H$ and d, can be determined by solving the semidefinite programming problem*

$$\begin{aligned} &\text{find} \quad (\boldsymbol{H}_\eta, \boldsymbol{a}_H, d) \\ &\text{subject to} \\ &\quad \boldsymbol{H}_\eta \succeq 0, \operatorname{tr}(\boldsymbol{H}_\eta) = \xi \\ &\quad \boldsymbol{y} = \sum_{i=0}^{q_H} [\boldsymbol{\Omega}_{i,0}\boldsymbol{\alpha}, \dots, \boldsymbol{\Omega}_{i,q_H}\boldsymbol{\alpha}] \boldsymbol{V}_\eta \boldsymbol{H}_\eta \boldsymbol{V}_\eta^T \boldsymbol{\varepsilon}_{i+1} + d\boldsymbol{1}_{\bar{N}} - \boldsymbol{Y}^T \boldsymbol{a}_H + \boldsymbol{\alpha}, \end{aligned} \tag{7.21}$$

where $\boldsymbol{\varepsilon}_i$ form the standard basis for $\mathbb{R}^{q_H+1}$ and $\xi = \eta^{-1}\|\boldsymbol{W}\|_*$.

Proof. Besides adapting the notation, the only necessary change with respect to the proof of Corollary 6.20 is the inclusion of the variable $\boldsymbol{a}_H$. □

Corollary 7.8. *With the definition of $\boldsymbol{Q}$ from Corollary 6.21 the predictive model for a new point $(\boldsymbol{u}_{f,t}, \dots, \boldsymbol{u}_{f,t-q_H}, \boldsymbol{y}_{t-1})$, in terms of the dual variables, is given by*

$$\widehat{y}_t = \sum_{i,j=0}^{q_H} Q_{ji} \boldsymbol{k}_j(\boldsymbol{u}_{f,t-i})^T \boldsymbol{\alpha} - \boldsymbol{y}_{t-1}^T \boldsymbol{a}_H + d, \tag{7.22}$$

with $Q_{ij} = (\boldsymbol{Q})_{i+1,j+1}$ and $\boldsymbol{k}_j(\boldsymbol{z})$ defined as for (7.11).

Proof. The predictive equation (7.22) directly follows from substituting (6.34) into the convex relaxation of (7.7) and applying the kernel trick. □

7.4.3 Numerical example

To study the effects of the proposed regularization scheme, some key properties are evaluated on a toy example. Due to the numerical complexity only the primal formulation is used with very few basis functions and data. The methods that are compared are

RR the model estimated by (7.9) which essentially facilitates ridge regression and,

NUC the nuclear norm based relaxation (7.19) proposed in this section.

To illustrate possible dependence on the choice of basis functions, two different bases are used.

Hinge Hinge functions. Defined as

$$\psi_1^H(x) = 1,$$
$$\psi_2^H(x) = x \text{ and}$$
$$\psi_m^H(x) = \begin{cases} x - b_m, & \text{for } x \geq b_m \text{ and} \\ 0, & \text{otherwise,} \end{cases}$$

for $m = 3, \dots, M$. The location of the kinks is given by the parameters b_m which are assumed given. For the experiments in this section a uniform distribution is used.

RBFN Gaussian Radial Basis Functions. The definition is $\psi_m^{RBF}(\boldsymbol{x}) = K(\boldsymbol{x}, \boldsymbol{z}_m)$, where K is the RBF kernel. For the experiments the bandwidth σ is fixed to $\sigma = 1$. The supporting points $\boldsymbol{z}_m$ are drawn from a uniform distribution.

As test systems several Hammerstein systems with $f(u_t) = \text{sinc}(u_t)$ as nonlinearity are generated. The linear block $H(z)$ is given by randomly chosen minimum phase systems with 5 poles and zeros. To obtain a slightly more challenging estimation problem, the input signal u_t is correlated. It is generated by filtering the white Gaussian noise process $v_t \sim \mathcal{N}(0, 1)$ according to $u_t = 0.9u_{t-1} + v_t$. For each example 300 samples corrupted by additive white Gaussian noise with variance $\sigma^2 = 0.2^2$ are computed. The data is split into three equal parts for training, validation and test. All examples are carried out with 30 basis functions and use the true model orders for the linear block.

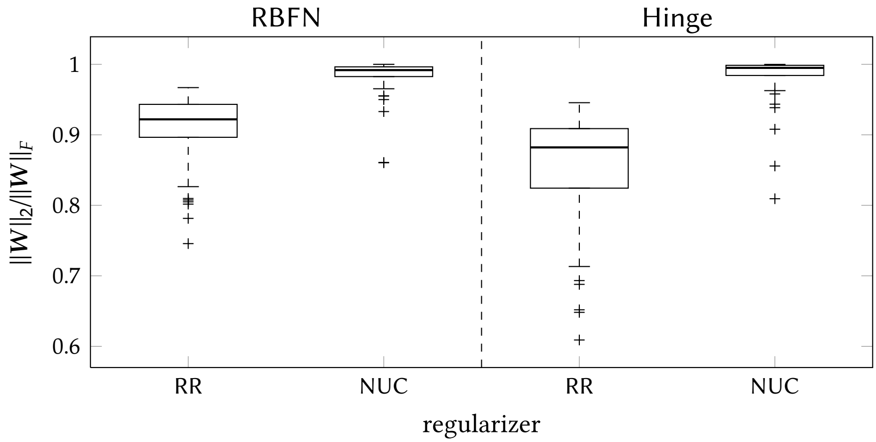

Figure 7.7: Analysis of rank one constraint violation for the different regularization schemes. The statistics are generated from 100 consecutive runs. The experimental conditions are according to Subsection 7.4.3.

Rank constraint violation The ratio $\|W\|_2/\|W\|_F$ is a measure for the closeness of W to rank one. Therefore it is an indication how close the relaxation is to the true solution. From the results shown in Figure 7.7 it can be concluded that the nuclear norm is superior to ridge regression in terms of low rank solutions. This trend is more pronounced for the Hinge basis.

Projection schemes The main performance criterion however is prediction performance. One interesting aspect in this regard is the performance after a projection from the full matrix W onto a factored solution in terms of w and b_H. A beneficial step to improve prediction performance can be to fix either w or b_H while re-estimating the other along with a_H. To illustrate the differences, four scenarios are compared.

OVER The overparametrized model using W and a_H.

DIRECT The structured model with w and b_H chosen as the dominant singular vectors of W along with a_H.

FIX-W The structured model, however only w is obtained from the SVD of W, while b_H and a_H are obtained by fixing w and estimating the other two.

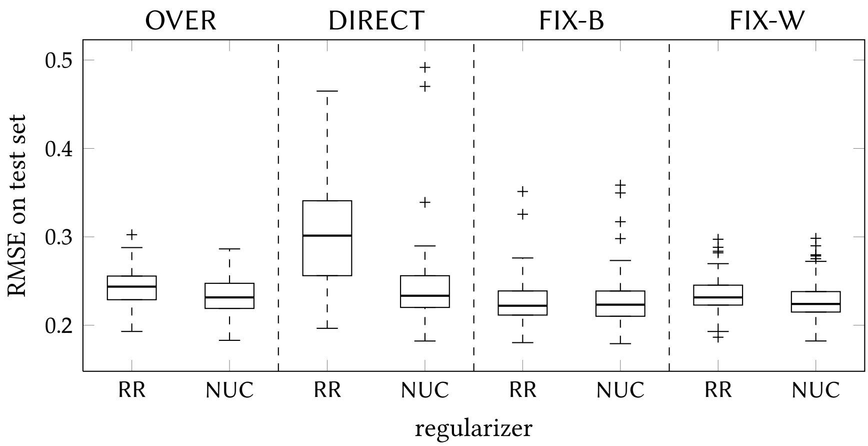

Figure 7.8: Comparison of different projection schemes applied to Hinge basis functions. The plots depict generalization performances for the unprojected model, as well as for the different projections listed in Subsection 7.4.3. The statistics are generated from 100 consecutive runs.

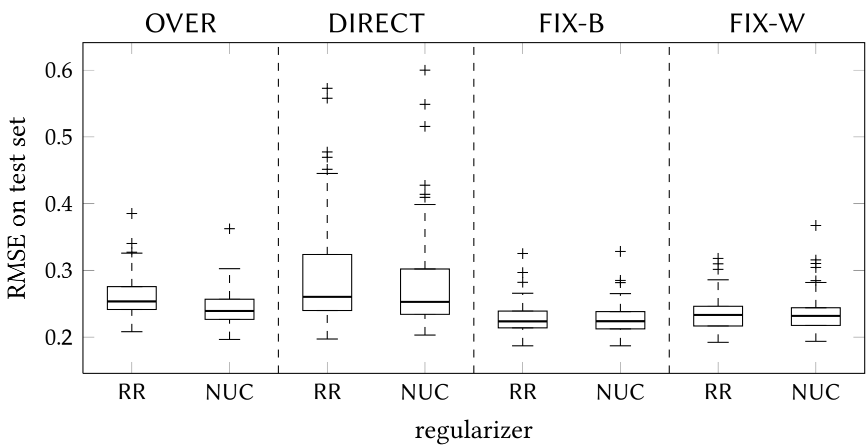

Figure 7.9: Comparison of different projection schemes applied to RBFN basis. The plots depict generalization performances for the unprojected model, as well as for the different projections listed in Subsection 7.4.3. The statistics are generated from 100 consecutive runs.

FIX-B The structured model, however only $\boldsymbol{b}_H$ is obtained from the SVD of $\boldsymbol{W}$, while $\boldsymbol{w}$ and $\boldsymbol{a}_H$ are obtained by fixing $\boldsymbol{b}_H$ and estimating the other two.

The results are shown in Figures 7.8 & 7.9. While FIX-W and FIX-B usually improve the generalization performance, DIRECT leads to a degradation. Furthermore using the estimate for $\boldsymbol{b}_H$ to re-estimate the remaining parameters is slightly better than using $\widehat{\boldsymbol{w}}$ for that purpose. In general the projection levels the performance of the regularization schemes. Even though the differences are small, the nuclear norm usually has a small advantage.

Parameter estimates Figure 7.10 shows the angle between the true coefficients for $H(\mathscr{z})$ and their estimated values. The parameter estimates are compared for the unprojected estimates and their values after projection and re-estimation. The best correlation is obtained for FIX-W. Using FIX-B yields only small improvements. This is in contrast to the result for generalization performance in the previous section where the refined model FIX-B was best. Comparing ridge regression and nuclear norm in the upper and lower panel of Figure 7.10 respectively, shows that the nuclear norm is slightly better in recovering the parameters.

Numerical complexity In Figure 7.11 the runtime of a single nuclear norm based estimation problem (7.19) is shown. The measurements were taken on a single node of the VIC3 supercomputer[1] at the KU Leuven. Only a single core of a Xeon 5420 with 2.5 GHz was used for the simulation. It can be seen that the computation time increases on a linear scale in the number of training samples N and approximately exponentially in the number of basis functions M and numerator coefficients q_H. The high computational cost is partially due to using CVX [Grant and Boyd, 2011] in combination with a general purpose SDP solver.

A test based on a real world data set is given as part of the next Section in particular Subsection 7.5.4.

[1] https://vscentrum.be/

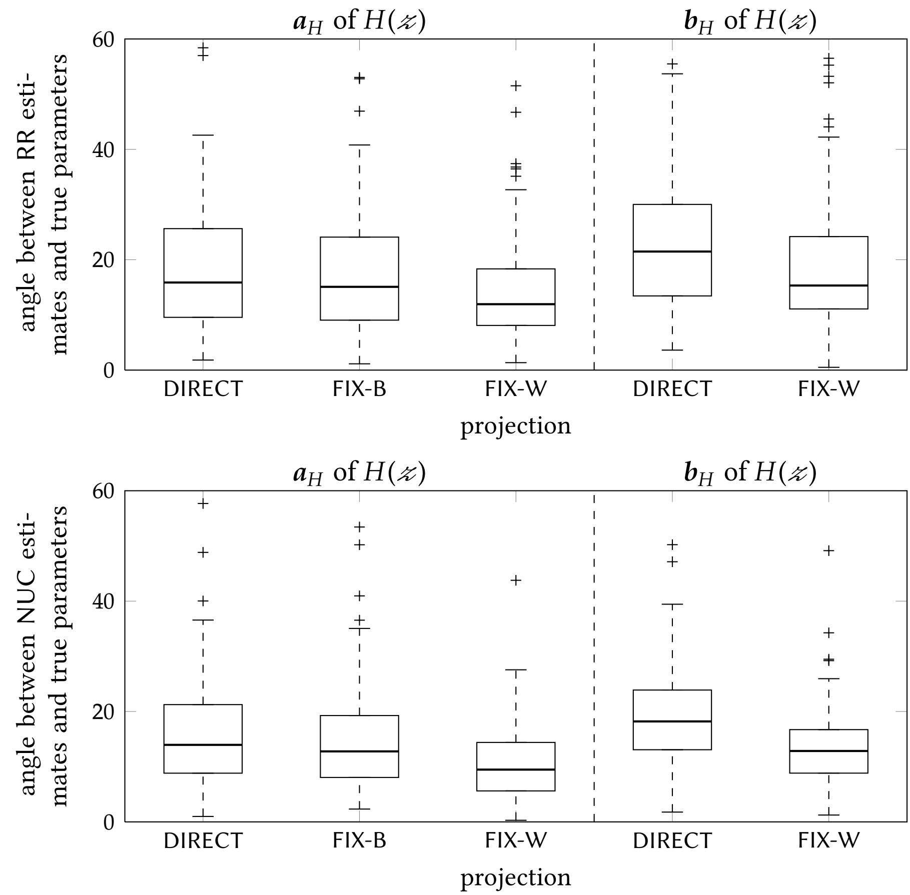

Figure 7.10: Comparison of parameter estimates for coefficients of the linear block $H(\varkappa)$ as defined in Subsection 7.4.3. The estimates are compared for the unprojected as well as the projected models. The statistics are generated from 100 consecutive runs.

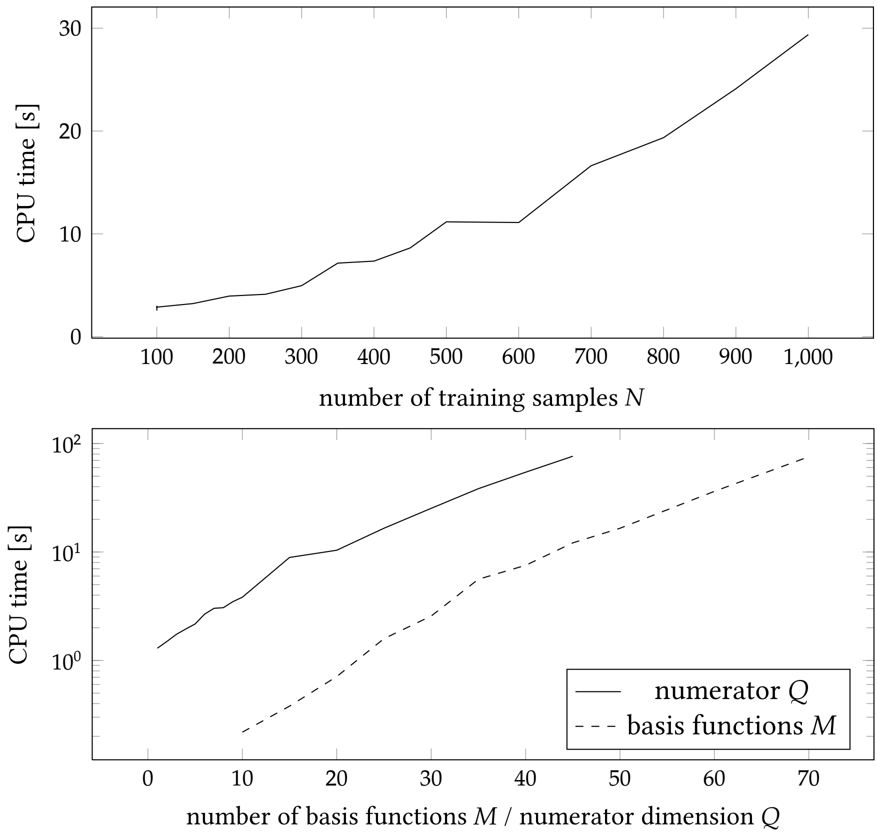

Figure 7.11: Time to estimate a single instance of (7.19) as a function of training samples N, of basis functions M and of numerator coefficients q_H while fixing the other quantities. The plots show average CPU times for 20 executions of the same problem.

7.5 Results on the Wiener-Hammerstein benchmark data set

7.5.1 Description of data set

This section describes results of all developed methods applied on the Wiener-Hammerstein benchmark data set [Schoukens et al., 2009]. The Wiener-Hammerstein benchmark data set consists of input and output data $\{(u_t, y_t)\}_{t=1}^{188,000}$ which were obtained from measurements of a real-life electronic nonlinear system.

The available data are obtained with a low noise level. As the real system is of Wiener-Hammerstein type one can expect that exploiting the system structure will be advantageous. Moreover, the availability of a large number of measurements suggests the use of fixed-size and large-scale methods. Along with NARX, STRCTRD and OVRPRZD from Section 7.2 the fixed-size model structures FS-NARX, FS-STRCTRD and FS-OVRPRZD as introduced in Section 7.3 are compared. In order to compare their predictive powers, the number of support vectors (SVs) is varied. In case of the fixed size models the number of data used for estimation is always $\tilde{N} \simeq 50{,}000$ and the number of support vectors corresponds to the dimension of the approximated feature map M. For all other models the number of support vectors is equal to the number of estimation data $\tilde{N}$.

The data set containing 188,000 measurements is split into several parts. The first 10,000 data points are discarded. The remaining 90,000 data points of the estimation data are then split up into blocks of 50,000 data points for model estimation, 20,000 data points as validation set $\mathcal{V}_1$ to select the regularization parameter γ and bandwidth σ of the RBF kernel and 20,000 data points as additional validation set $\mathcal{V}_2$ to select the model orders p, q and p_H, q_H, q_f respectively. The remaining 88,000 data points from the complete data set are left untouched during the whole model selection and estimation process. They are only used to assess the quality of the finally obtained models.

7.5.2 Model order selection

Model order selection and selection of the regularization parameter γ and the kernel bandwidth σ of the RBF kernel are performed as outlined in the previous section. For the NARX model the result is shown in Figure 7.12,

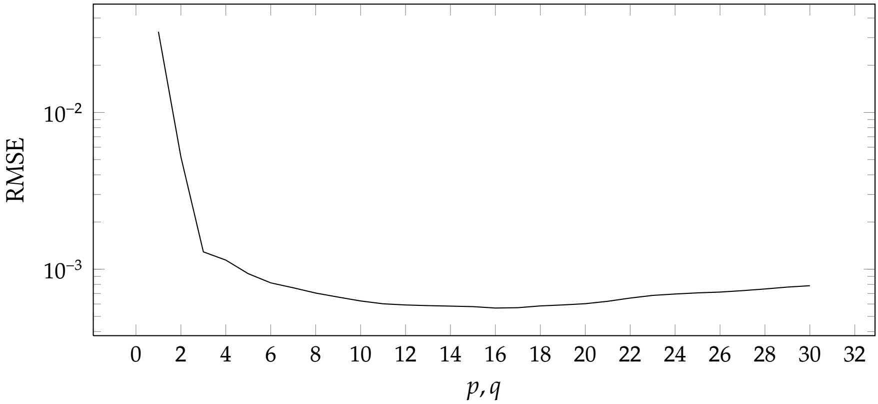

Figure 7.12: Model order selection for NARX model for the Wiener-Hammerstein benchmark data set. The RMSE is computed for one-step-ahead predictions on the independent test set $\mathcal{V}_2$.

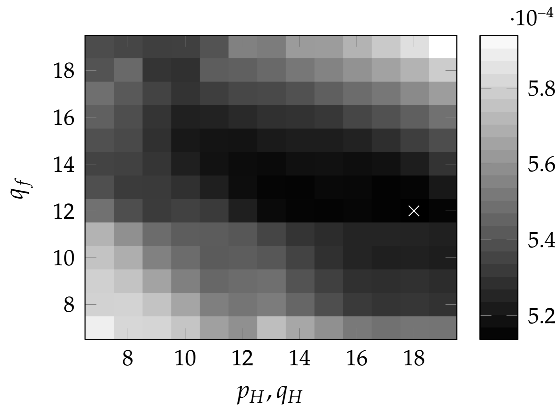

Figure 7.13: Model selection for structured Wiener-Hammerstein model. The RMSE for one-step-ahead prediction on the independent test set $\mathcal{V}_2$ is shown. The cross marks the optimal model order $p_H = q_H = 18$ and $q_f = 12$.

the optimal model order is $p = q = 16$. In case of the structured models one obtains $q_f = 12$ and $p_H = q_H = 18$ as shown in Figure 7.13.

One can conclude that both the structured as well as the unstructured models achieve similar (good) performance over a wide range of model orders. For the fixed-size structured models the computational cost scales with q_H.

Table 7.6: RMSE $\times 10^3$ on test set for the Wiener-Hammerstein benchmark data set and different number of support vectors. The first value gives the performance of the model in recursive simulation mode; the value between parentheses is its one-step-ahead performance. The methods reported in this table have access to a number of data points equal to the number of support vectors (# SVs = $\tilde{N}$).

# SVs	NARX	STRCTRD	OVRPRZD
100	41.33 (5.54)	20.44 (0.74)	57.04 (0.86)
250	52.27 (1.59)	15.09 (0.67)	27.84 (0.66)
500	29.31 (0.75)	13.27 (0.61)	13.99 (0.57)
1,000	12.75 (0.56)	13.17 (0.57)	10.94 (0.51)
2,500	10.04 (0.52)	7.95 (0.50)	10.00 (0.48)
5,000	9.50 (0.50)	6.22 (0.48)	6.06 (0.43)
10,000	8.89 (0.49)	5.84 (0.47)	4.60 (0.40)

Table 7.7: RMSE $\times 10^3$ for recursive simulation on test set for the Wiener-Hammerstein benchmark data set and different number of support vectors. The number of data used for estimation in the so-called fixed-size models is always $\tilde{N} \simeq 50{,}000$ and the number of support vectors corresponds to the dimension of the approximated map (# SVs = $M \ll \tilde{N}$). The estimate for b_H used for FS-STRCTRD with 2,500 SVs is obtained from FS-OVRPRZD with 1,000 SVs.

# SVs	FS-NARX	FS-STRCTRD	FS-OVRPRZD
100	23.46 (0.65)	8.74 (0.51)	7.00 (0.43)
250	9.38 (0.52)	5.65 (0.54)	4.51 (0.40)
500	8.85 (0.48)	4.46 (0.45)	3.90 (0.39)
1,000	8.41 (0.47)	4.27 (0.44)	3.85 (0.38)
2,500	5.08 (0.43)	4.18 (0.44)	—

Therefore the model orders are chosen as $p_H = q_H = 12$ and $q_f = 14$ as these only result in a slight drop of the prediction performance.

Table 7.8: Generalization performance and rank one constraint violation for the Wiener-Hammerstein benchmark data. Comparison of ridge regression RR and nuclear norm NUC as defined in Section 7.4.

	$10^3 \cdot$ RMSE, using				
regularization	OVER	DIRECT	FIX-B	FIX-W	$\|\|W\|\|_2/\|\|W\|\|_F$
ridge regression (RR)	2.41	43.2	3.33	5.09	0.62
nuclear norm (NUC)	2.38	19.0	2.75	4.72	0.77

7.5.3 Performance for different number of support vectors

The main results in terms of prediction performance for different number of support vectors are given in Tables 7.6 and 7.7. For the recursively simulated values one can see that including structural information is able to increase the performance substantially over the NARX model. In case of the structured models STRCTRD is much better than OVRPRZD for small numbers of support vectors ($\tilde{N} < 500$). In contrast to all other models, the fixed-size models use 50,000 data points for the estimation. The results clearly indicate that in the presence of many measurements the prediction performance can be significantly improved. As with the previous models, including model structure yields better models than the simple NARX structure. Time domain and frequency domain plots of the residual signal are given in Figures 7.14 through 7.17. These are presented for the best models shown in Tables 7.6 and 7.7.

Note that for all models shown here, the modeling errors are dominant. Several other techniques presented in the benchmark session [Schoukens et al., 2009; Hjalmarsson et al., 2012] were able to reduce RMSE value roughly one order of magnitude further. One source for modeling errors in the overparametrized model formulations is the approximation of the first linear block by a finite impulse response.

7.5.4 Performance based on nuclear norm regularization

This subsection picks up the improved convex relaxation presented in Section 7.4. Note that the objective is only to evaluate the nuclear norm regularization scheme on a real world data set. Without special solvers this problem can only be solved for a small number of basis functions. Therefore the absolute performances are much lower than that in the previous subsection.

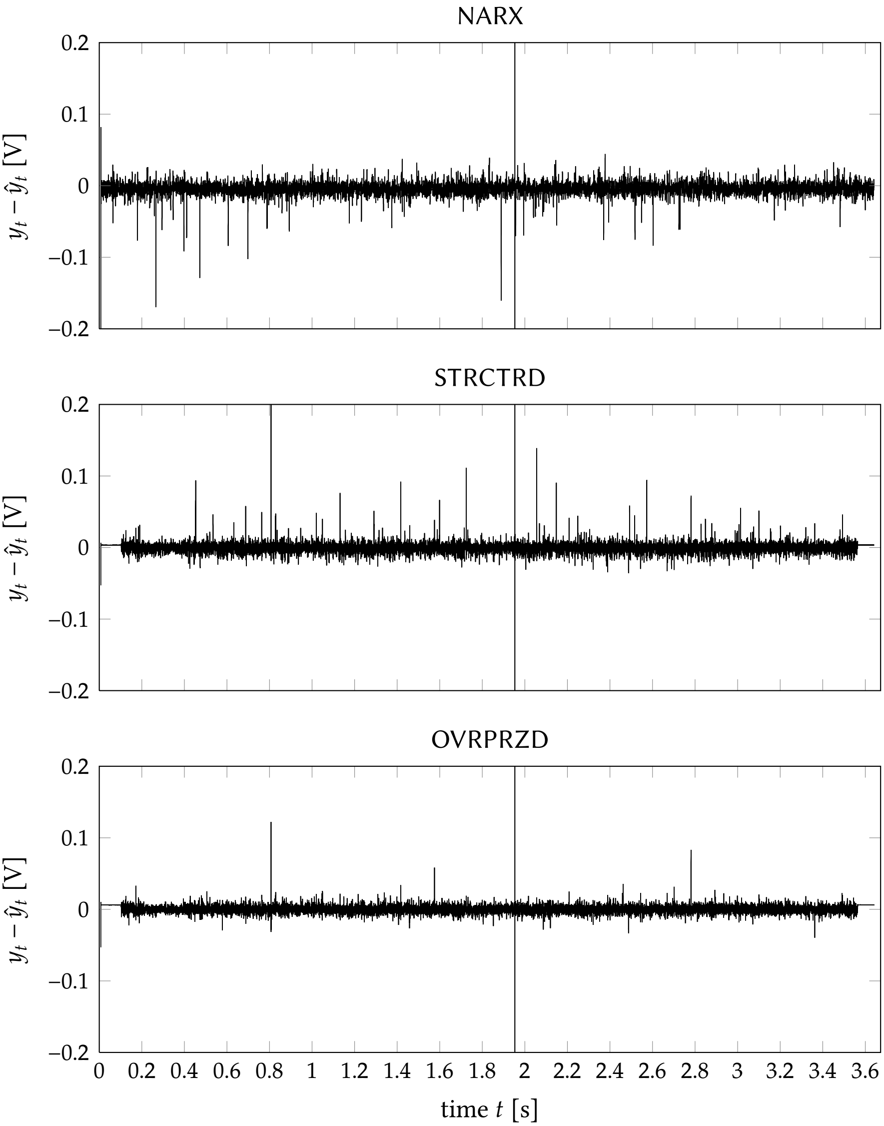

Figure 7.14: Time domain plots for prediction error $y_t - \hat{y}_t$ for the NARX and structured models on the Wiener-Hammerstein benchmark data set. All values are computed for recursive simulations of $\hat{y}_t$. The best models from Table 7.6 are shown.

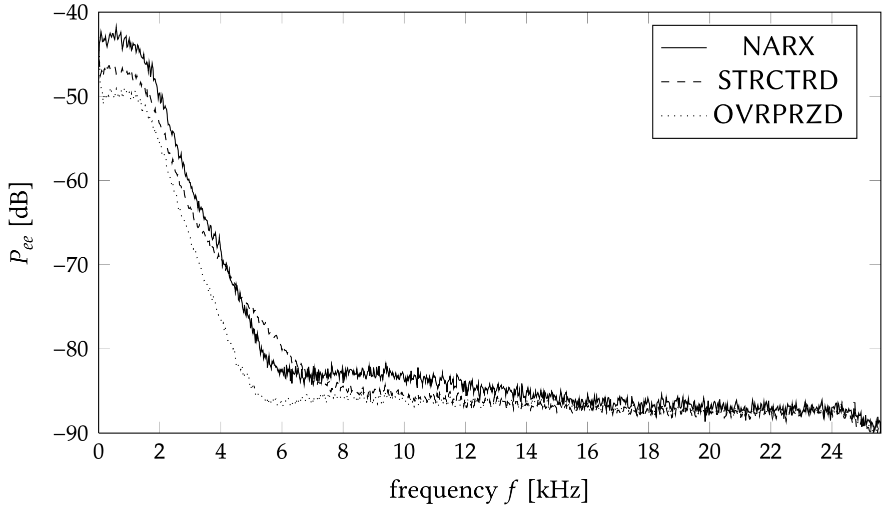

Figure 7.15: Frequency domain plots for prediction error $y_t - \hat{y}_t$ for the NARX and structured models on the Wiener-Hammerstein benchmark data set. All values are computed for recursive simulations of $\hat{y}_t$. The best models from Table 7.6 are shown.

As previously a validation set is used for model selection along with an independent test set for the final evaluations. However, training, validation and test set are reduced to 1000 samples long consecutive parts of the complete time series starting from sample 10,000. The simulations are carried out with just 30 RBFN basis functions, which are drawn from a normal distribution. The bandwidth of these basis functions, the standard deviation of the normal distribution as well as the model orders p_H, q_H, q_F and the regularization parameter η are selected based on performance on the validation set. For timing reasons, the selection procedure is run for the model with ridge regression RR. The selected model orders are $\hat{p}_H = 13$, $\hat{q}_H = 20$ and $\hat{q}_F = 2$.

Table 7.8 summarizes the results. It can be seen that the results for the artificial problems analyzed in Subsection 7.4.3 can be transferred to a real data set. Notably, the nuclear norm regularization results in a much better approximation of the rank-1 constraint. Yet in terms of generalization performance the model obtained by ridge regression yields very similar values.

The overall poor performance and the decrease in performance after projection are due to the low number of basis functions. As the model order q_F grows, the fixed number of basis functions have to cover a larger space.

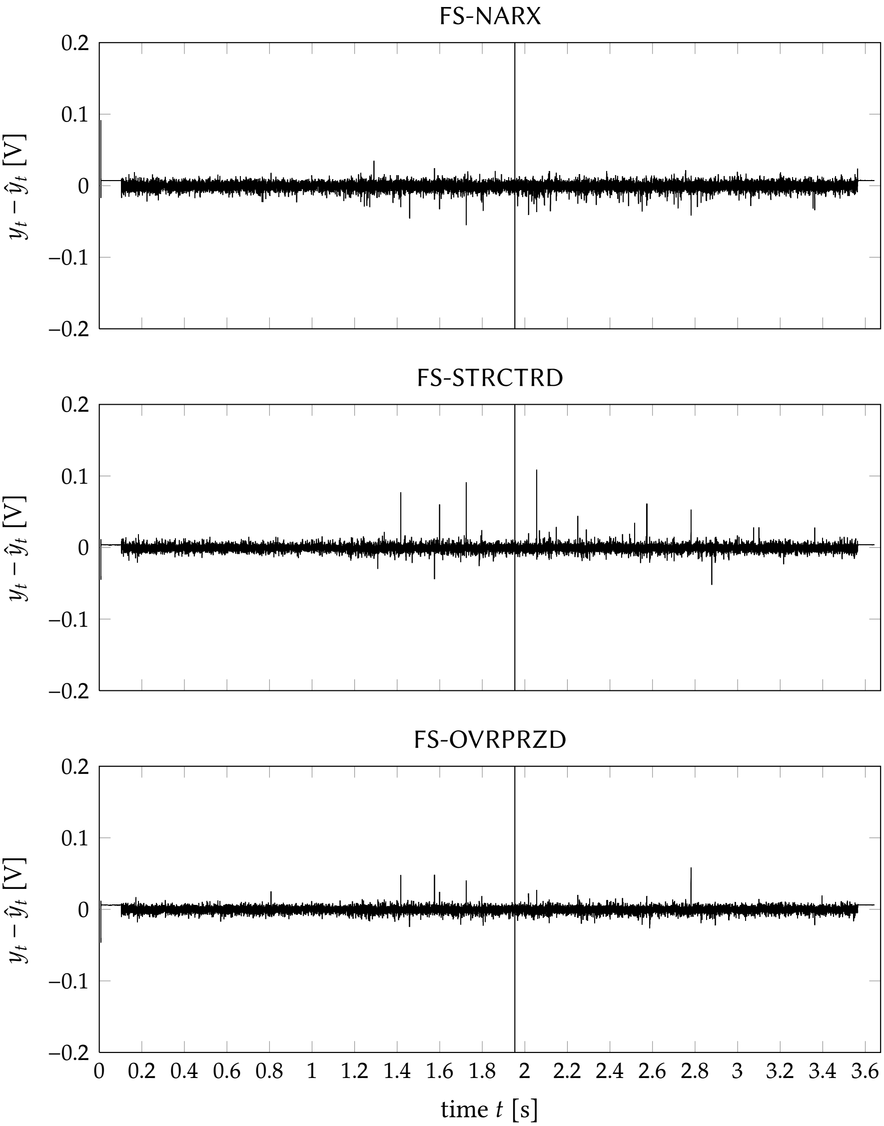

Figure 7.16: Time domain plots for prediction error $y_t - \hat{y}_t$ for the fixed-size models on the Wiener-Hammerstein benchmark data set. All values are computed for recursive simulations of $\hat{y}_t$. The best models from Table 7.7 are shown.

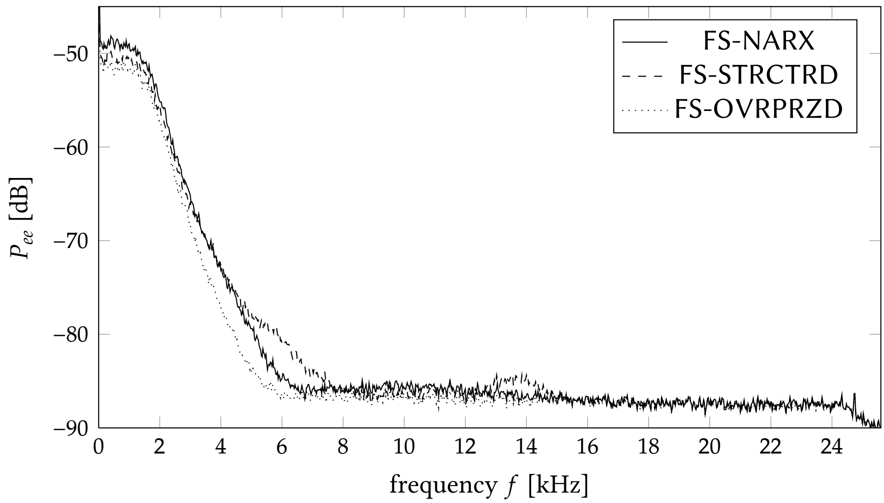

Figure 7.17: Frequency domain plots for prediction error $y_t - \hat{y}_t$ for the fixed-size models on the Wiener-Hammerstein benchmark data set. All values are computed for recursive simulations of $\hat{y}_t$. The best models from Table 7.7 are shown.

Therefore the selected model orders are chosen suboptimally. The estimate for q_F is selected small enough to be described by 30 basis functions. Then the orders p_H and q_H are chosen high such that they can compensate for the loss of expressive power.

7.6 Conclusions

This chapter describes the tailoring of LS-SVM based models to Wiener-Hammerstein systems. Section 7.2 starts by gradually deriving a kernel based convex approximation for an initially nonconvex estimation problem. Besides discussing the convex relaxation itself, the projection back onto the original model class is subject of some attention. The section concludes by validating different aspects of the problem on numerical examples.

The following section is dedicated to extensions for handling large scale data sets. The proposed methods are directly compared on a numerical example at the end of this section. Picking up the results from the last chapter, Section 7.4, derives an improved convex relaxation based on nuclear norm regularization. Again some general properties are analyzed using numerical examples.

In Section 7.5 all proposed methods are validated on the Wiener-Hammerstein benchmark problem. The first conclusion that can be drawn from this, as well as the artificial examples studied earlier, is that all models converge quite fast as the number of support vectors grows. Due to the large size of the data set, tests with fixed-size models were performed. By making use of the whole data set, improved prediction performances are obtained when compared to their classical counterparts. As before incorporating structure into the model is beneficial for the prediction performance of the models. Finally the evaluation of the improved convex relaxation based on nuclear norm regularization suffers from its inability to scale to larger problem sizes. However, it shows promising results and it would be insightful to benchmark its performance once the computational complexity can be handled.

Linear noise models 8

Based on the publication Falck, T., Suykens, J. A. K., and De Moor, B. (Dec. 2010). "Linear Parametric Noise Models for Least Squares Support Vector Machines". In: *Proceedings of the 49th IEEE Conference on Decision and Control.* (Atlanta, GA, USA, Dec. 15–17, 2010), pp. 6389–6394.

In the previous chapters the dominant quality measure for all models has been their prediction performance on unknown data. This measure has several advantages like being the quality of interest for many applications and having a simple representation and interpretation as a single number. However knowing only the predictive performance has the major disadvantage, that there is no insight how far the current model is from the best model. In this context the best model is a model that produces a sequence of residuals e_t that is mutually independent and also independent of the inputs u_t. Hence, such a model captures all information in the data as no more predictable information is contained in the residuals. This insight gives rise to two tests, known as residual tests, capable of analyzing a given model with respect to its distance to the best model. True statistical independence is impossible to test. Therefore one mostly considers correlation. For a linear system with linear noise dynamics and Gaussian noise this is equivalent to independence. The first test checks that the cross-correlation between the inputs u_t and the residuals e_t is (approximately) zero. If this is not the case, this is a strong indicator that the model order is not sufficient to describe the system dynamics. In the second iteration one should check that the autocorrelation of the residual signal is (approximately) equal to a Kronecker delta function $\sigma\,\delta(t)$, where

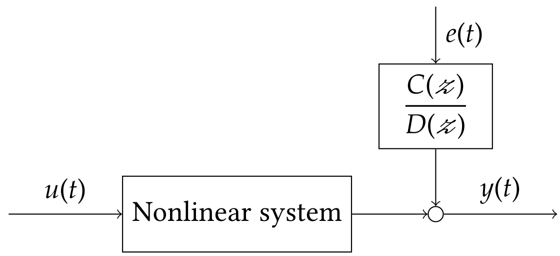

Figure 8.1: Nonlinear system with linear parametric noise model.

σ^2 is the variance of the corresponding unpredictable innovation sequence. This general scheme can be extended by also testing higher order statistics as these capture nonlinear behavior up to a certain degree. In the context of this chapter only linear correlations of the noise will be analyzed however.

In case the residuals are not white, one can improve the model by not only describing the system dynamics but by also trying to capture the dynamic behavior of the noise process. This yields *noise models* which, in case of polynomial models as described in Subsection 2.3.2, are described by the A, C and D polynomials. Here the A polynomial has to be handled with care as it represents only the particular case in which the system and the noise share the same denominator dynamics.

Under this aspect, the Box-Jenkins model structure is optimal as it offers the greatest flexibility for model as well as noise dynamics. However, as with all model structures apart from ARX and FIR, the corresponding estimation problem is nonconvex. For linear system identification this problem is less severe as there are several techniques to obtain good initial estimates. These can then be refined by nonlinear optimization. For nonlinear systems the situation is much more involved as there is no universal low order parametrization for general nonlinear systems. Therefore, on the one hand good initial estimates are even more important as the models are more prone to local minima and on the other hand useful initial estimates are harder to obtain. The results in [Suykens and Vandewalle, 2000] on a recurrent formulation of LS-SVMs indicate the capabilities of such an approach but also its limitations.

Noise models in a nonlinear setting can come in different flavors. The most general formulations are given in Table 2.2. This basically covers the case when the autocorrelation test of the residuals, introduced in the first paragraph, fails. In imitation of the model structures defined in Subsection 2.3.2 this model structure will be denoted as ARMA-NARX here. In the next section of this chapter it is briefly shown how to integrate an a priori known ARMA noise model with a LS-SVM based NARX model to obtain an improved predictive performance. As the assumption of having an a priori known noise model is very restrictive, the remainder of the chapter discusses how to jointly estimate a noise model and a LS-SVM model. To make the estimation tractable the model class for the noise model is reduced to purely autoregressive models. The estimation is based on the idea of overparametrization that has already been used in Chapter 7 for the identification of Hammerstein systems. The proposed methods are then evaluated on numerical examples in Section 8.3. This is done for an artificial data set as well as for real world data representing loads in a large computer network.

Structure of the chapter The first section of this chapter reviews several possibilities to incorporate a linear parametric noise model with a LS-SVM model for the system dynamics. Section 8.2 builds upon the same ideas as Chapter 7, namely overparametrization, to jointly estimate nonlinear system dynamics and linear noise model via a convex relaxation. Before concluding the chapter in the last section, some numerical results are given in Section 8.3.

8.1 Incorporating linear noise models in LS-SVMs

For the scope of this section assume that a noise model with polynomials $C(\mathscr{z})$ and $D(\mathscr{z})$ of orders n_c and n_d respectively is given. Without loss of generality one can further assume that both polynomials are monic, i.e. the coefficient of the highest power is one. Based on this, a correlated noise process r_t can be defined as $D(\mathscr{z})r_t = C(\mathscr{z})e_t$ where e_t is a white noise process. Then the basic model given by (4.1) can be augmented with a colored noise process such that

$$y_t = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b + r_t. \tag{8.1}$$

Integrating this model in the primal LS-SVM formulation (4.2) yields

$$\begin{aligned}
\min_{\boldsymbol{w},b,e_t,r_t} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{t=1}^{N} e_t^2 \\
\text{subject to} \quad & y_t = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b + r_t, \quad t = 1, \dots, N, \\
& D(\mathscr{z})r_t = C(\mathscr{z})e_t, \qquad\quad t = \tau, \dots, N.
\end{aligned} \tag{8.2}$$

Here two choices for τ are considered i) $\tau = 1$ with zero initial conditions for e_t and r_t for $t \leq 0$ and ii) $\tau = \max(n_c, n_d) + 1$. The first choice is simpler in its formulation, however as the second option does not need an assumption on the initial conditions it is more powerful. For an AR noise model structure, i.e. $C(\not{z}) = 1$, this model has already been considered in [Espinoza et al., 2005b]. There the authors suggest that if $C(\not{z})$ and $D(\not{z})$ are not known a priori, they can be seen as additional hyper-parameters to the problem. As such, the coefficients of C and D then have to be selected for example by cross-validation. Due to the associated computational cost this is only feasible for very low order noise models. An alternative more efficient but restricted approach is proposed in the next section. The remainder of this section keeps on assuming full knowledge of the polynomials C and D. Note that in case $C(\not{z})$ and $D(\not{z})$ correspond to the true noise model, the residual e_t is white and formulation (8.2) becomes optimal.

For the sake of simplicity assume that $n_c = n_d = P$, then the constraint $D(\not{z})r_t = C(\not{z})e_t$ for $t = P+1, \dots, N$ can be written in matrix notation as $\boldsymbol{D}\boldsymbol{r} = \boldsymbol{C}\boldsymbol{e}$ with

$$\boldsymbol{C} = \begin{bmatrix} c_P & \cdots & c_1 & 1 & & \\ & \ddots & & & \ddots & \\ & & c_P & \cdots & c_1 & 1 \end{bmatrix}, \quad \boldsymbol{D} = \begin{bmatrix} d_P & \cdots & d_1 & 1 & & \\ & \ddots & & & \ddots & \\ & & d_P & \cdots & d_1 & 1 \end{bmatrix},$$

with $\boldsymbol{C}, \boldsymbol{D} \in \mathbb{R}^{(N-P)\times N}$, $\boldsymbol{e} = [e_1, \dots, e_N]^T$ and $\boldsymbol{r} = [r_1, \dots, r_N]^T$. For zero initial conditions and $\tau = 1$ these matrices can be extended to square matrices. Therefore define

$$\boldsymbol{C}_0 = \begin{bmatrix} 1 & & & \\ c_1 & 1 & & \\ \vdots & & \ddots & \\ c_{n_c-1} & \cdots & c_1 & 1 \end{bmatrix}, \quad \boldsymbol{D}_0 = \begin{bmatrix} 1 & & & \\ d_1 & 1 & & \\ \vdots & & \ddots & \\ d_{n_d-1} & \cdots & d_1 & 1 \end{bmatrix}.$$

Then

$$\bar{\boldsymbol{C}}_0 = \begin{bmatrix} \boldsymbol{C}_0 & \boldsymbol{0}_{\boxtimes} \\ - & \boldsymbol{C} & - \end{bmatrix}, \quad \bar{\boldsymbol{D}}_0 = \begin{bmatrix} \boldsymbol{D}_0 & \boldsymbol{0}_{\boxtimes} \\ - & \boldsymbol{D} & - \end{bmatrix}$$

are square. Hence, for zero initial conditions, the constraint reads $\bar{\boldsymbol{D}}_0\boldsymbol{r} = \bar{\boldsymbol{C}}_0\boldsymbol{e}$. Using this, it is straightforward to show that (8.2) can be solved by reweighting the residuals in the standard LS-SVM formulation (4.2). Note that reweighting the residuals can also be used for various other purposes, such as obtaining a robust or a sparse solution as shown in [Suykens, Van Gestel, et al., 2002].

Proposition 8.1. *Solving the standard LS-SVM problem* (4.2) *with weighted residuals* $\boldsymbol{e}^T\boldsymbol{R}\boldsymbol{e}$ *and* $\boldsymbol{R} = \bar{\boldsymbol{D}}_0^T\bar{\boldsymbol{C}}_0^{-T}\bar{\boldsymbol{C}}_0^{-1}\bar{\boldsymbol{D}}_0$ *in place of* $\boldsymbol{e}^T\boldsymbol{e}$ *is equivalent to solving* (8.2) *with* $\tau = 1$ *and zero initial conditions for* r_t *and* e_t*.*

The solution to the weighted problem is given by the linear system

$$\begin{bmatrix} \boldsymbol{\Omega} + \gamma^{-1}\boldsymbol{R}^{-1} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ 0 \end{bmatrix} \tag{8.3}$$

in terms of the dual variables $\boldsymbol{\alpha}$ *and with* $\Omega_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$.

Proof. The noise model can be rewritten as $\boldsymbol{e} = \bar{\boldsymbol{C}}_0^{-1}\bar{\boldsymbol{D}}_0\boldsymbol{r}$ as the matrix $\bar{\boldsymbol{C}}_0$ is invertible. Substitution of this relation into the objective function of (8.2) yields the weighting matrix $\boldsymbol{R}$. Note that $\bar{\boldsymbol{D}}_0$ is also invertible and thus $\boldsymbol{R}$ as well. This is needed for the solution in the dual domain.

Deriving the dual system relies on Lagrangian duality and the kernel trick $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\varphi}(\boldsymbol{x}_i)^T\boldsymbol{\varphi}(\boldsymbol{x}_j)$. The details are given in [Suykens, De Brabanter, et al., 2002]. □

As the procedure outlined above requires the invertibility of $\bar{\boldsymbol{C}}_0$ and $\bar{\boldsymbol{D}}_0$ it does not apply to the case with unknown initial conditions. To nevertheless obtain the optimal solution one could just write down the Lagrangian and solve it, but in this case it is easier to solve the nonlinear model $y_t = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b + r_t$ for r_t. Then substitution into $D(\mathcal{z})r_t = C(\mathcal{z})e_t$ yields a new combined modeling equation

$$y_t = \boldsymbol{w}^T \sum_{k=0}^{n_d} d_k\boldsymbol{\varphi}(\boldsymbol{x}_{t-k}) + b\sum_{k=1}^{n_d} d_k - \sum_{k=1}^{n_d} d_k y_{t-k} + C(\mathcal{z})e_t \tag{8.4}$$

with $d_0 := 1$. This relation can also be written more compactly using the matrix notation introduced above as $\boldsymbol{D}\boldsymbol{y} = \boldsymbol{D}\boldsymbol{\Phi}^T\boldsymbol{w} + b\boldsymbol{D}\mathbf{1} + \boldsymbol{C}\boldsymbol{e}$ where $\boldsymbol{\Phi} = [\boldsymbol{\varphi}(\boldsymbol{x}_1), \dots, \boldsymbol{\varphi}(\boldsymbol{x}_N)]$.

Then an alternative to the dual system (8.3) for Problem (8.2), using the combined modeling equation, is

$$\begin{bmatrix} \boldsymbol{D}\boldsymbol{\Omega}\boldsymbol{D}^T + \gamma^{-1}\boldsymbol{C}\boldsymbol{C}^T & \boldsymbol{D}\mathbf{1} \\ \mathbf{1}^T\boldsymbol{D}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\eta} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{D}\boldsymbol{y} \\ 0 \end{bmatrix}. \tag{8.5}$$

This formulation has the advantage that no explicit inverse has to be computed. Of course it can also be used in case zero initial conditions are assumed, in that case one can relate the dual variables in (8.3) and (8.5) (with $\bar{\boldsymbol{C}}_0$ and $\bar{\boldsymbol{D}}_0$ in place of $\boldsymbol{C}$ and $\boldsymbol{D}$ respectively) through $\boldsymbol{\alpha} = \bar{\boldsymbol{D}}_0^T\boldsymbol{\eta}$.

The system in (8.5) not only has a modified regularization term $\boldsymbol{C}\boldsymbol{C}^T$ but also a different kernel matrix, i.e. $\boldsymbol{D}\boldsymbol{\Omega}\boldsymbol{D}^T$. Showing that this matrix is positive semidefinite is straightforward. Let $\boldsymbol{z}$ be an arbitrary vector in $\mathbb{R}^{N-P}$ and define $z' = D^T z$. Then $z^T D\Omega D^T z = z'^T \Omega z'$. Therefore the modified matrix is positive semidefinite as long as the original kernel matrix is. Given that the problem (8.2) can be solved by considering a modified kernel matrix, an important question is whether the information of the noise model can be directly embedded into the kernel function. The answer is given by Espinoza et al. [2005b] where an equivalent kernel is defined as $K_{\mathrm{eq}}(\bar{x}_i, \bar{x}_j) = K(x_i, x_j) + \sum_{k,l=1}^{P} d_k d_l K(x_{i-k}, x_{j-l})$ with $\bar{x}_t = (x_t, \dots, x_{t-P})$. The equivalent kernel can be extracted from (8.4) by defining an equivalent feature map $\varphi_{\mathrm{eq}}(\bar{x}_t) = \varphi(x_t) + \sum_{k=1}^{P} d_k \varphi(x_{t-k})$ and forming the inner product $\varphi_{\mathrm{eq}}(\bar{x}_i)^T \varphi_{\mathrm{eq}}(\bar{x}_j)$.

The combination of the dual problem (8.5) and the equivalent kernel function yields the one-step-ahead predictor for a new point $(z_t, \dots, z_{t-n_d}, y_{t-1}, \dots, y_{t-n_d})$ at time t. It is given by

$$\hat{y}_t = \sum_{n=1}^{N} \eta_n K_{\mathrm{eq}}(\bar{x}_n, z_t) + b \sum_{k=1}^{n_d} d_k - \sum_{k=1}^{n_d} d_k y_{t-k}. \tag{8.6}$$

To obtain a model with good generalization performance model selection is needed. If the parameters c_k and d_k of the noise model are not known a priori, they have to be included in the model selection. In that case the regularization parameter γ, parameters of the kernel function and the noise model coefficients have to be tuned according to a validation scheme. This is computationally very demanding for all but very low order noise models. Therefore the next sections propose a convex relaxation that is able to estimate noise model coefficients jointly with the parameters of the nonlinear model w and b.

8.2 Estimation of parametric noise models

In the following, only purely autoregressive noise models will be considered, i.e. $C(\mathcal{z}) = 1$. The studied model structure is denoted AR(P)-NARX where P is the model order, i.e. $n_d = P$. This simplifies the estimation problem as the nonconvex product of unknowns $c_k e_{t-k}$ in $C(\mathcal{z})e_t$ does not have to be considered. It also simplifies the prediction because the sequence e_t does not have to be estimated.

8.2.1 Primal model

The problem of jointly estimating the nonlinear model with its parameters w and b and a linear parametric noise model, defined by the coefficients $\{d_k\}_{k=1}^{P}$, is formalized in the following nonconvex optimization problem

$$\begin{aligned} \min_{w,b,d_k,e_t,r_t} \quad & \frac{1}{2}w^T w + \frac{1}{2}\gamma \sum_{t=P+1}^{N} e_t^2 \\ \text{subject to} \quad & y_t = w^T \varphi(x_t) + b + r_t, \quad t = 1, \dots, N, \\ & r_t = e_t - \sum_{k=1}^{P} d_k r_{t-k}, \quad t = P+1, \dots, N. \end{aligned} \tag{8.7}$$

The nonconvexity is due to the bilinear term $d_k r_{t-k}$. Based on the idea of overparametrization as in Chapter 7 the problem can be cast into a form where the nonconvexity is concentrated in a rank constraint. Therefore in analogy to Subsection 7.2.2 one can introduce new variables $w_k = d_k w$ for $k = 0, \dots, P$. These can also be written in matrix form as $W = [w_0, \dots, w_P] = wd^T$ with $d = [1, d_0, \dots, d_P]^T$. Then, as in (7.8), an equivalent optimization problem can be stated based on a rank-1 constraint on W. In contrast to (7.8) for Hammerstein problems where all references to the variables $b_{H,k}$ have been absorbed by the new variables w_k, here references to the variables d_k are still present in the problem. Therefore one has to ensure collinearity between d and the columns of W which can be achieved by augmenting the rank constraint to $\text{rank}([W^T, d]) = 1$. Similar to Subsection 7.2.2 one can state an equivalent optimization problem with all nonconvexity concentrated in a rank constraint,

$$\begin{aligned} \min_{w_k,\bar{b},d_k,e_t} \quad & \frac{1}{2}\sum_{k=0}^{P} w_k^T w_k + \frac{1}{2}\gamma \sum_{t=P+1}^{N} e_t^2 \\ \text{subject to} \quad & y_t + \sum_{k=1}^{P} d_k y_{t-k} = \sum_{k=0}^{P} w_k^T \varphi(x_{t-k}) + \bar{b} + e_t, \\ & \qquad t = P+1, \dots, N, \\ & \text{rank}([W^T, d]) = 1. \end{aligned} \tag{8.8}$$

Based on this equivalent problem formulation a convex approximation is straightforwardly obtained by dropping the rank constraint. Note that in the formulation above the expression $b \sum_{k=0}^{P} d_k$ has been — without loss of generality — replaced by $\bar{b}$.

8.2.2 Solution in dual domain

Due to the often implicit definition of the feature map φ in LS-SVMs, the solution has to be obtained in the dual domain for which only the kernel function needs to be known. To facilitate a compact notation the effective number of constraints $N - P$ is denoted as $\tilde{N}$. The dual solution of the approximation of (8.8) is formalized in the following Lemma.

Lemma 8.2. *The solution of* (8.8) *without rank constraint is given in the dual domain by*

$$\begin{bmatrix} \sum_{k=0}^{P} \boldsymbol{\Omega}_k + \gamma^{-1}\boldsymbol{I}_{\tilde{N}} & \boldsymbol{Y}^T & \boldsymbol{1}_{\tilde{N}} \\ \boldsymbol{Y} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{1}_{\tilde{N}}^T & \boldsymbol{0} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{d} \\ \bar{b} \end{bmatrix} = \begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{0} \\ 0 \end{bmatrix} \tag{8.9}$$

with $(\boldsymbol{\Omega}_k)_{ij} = K(\boldsymbol{x}_{i-k}, \boldsymbol{x}_{j-k})$, $P+1 \leq i, j \leq N$ *where* $(\boldsymbol{\Omega}_k)_{ij}$ *is the ij-th element of* $\boldsymbol{\Omega}_k$. *Furthermore* $\boldsymbol{\alpha}$ *are the Lagrange multipliers corresponding to the equality constraints,* $\boldsymbol{y}_0 = [y_{P+1}, \dots, y_N]^T$, $\boldsymbol{y}_t = [y_{t-1}, \dots, y_{t-P}]^T$ *for* $t = P+1, \dots, N$ *and* $\boldsymbol{Y} = [\boldsymbol{y}_{P+1}, \dots, \boldsymbol{y}_N]$.

Proof. The Lagrangian for (8.8) without the rank constraint is

$$\begin{aligned} \mathscr{L}(\boldsymbol{w}_k, \bar{b}, d_k, e_t, \boldsymbol{\alpha}) = \frac{1}{2} \sum_{k=0}^{P} \boldsymbol{w}_k^T \boldsymbol{w}_k + \frac{1}{2}\gamma \sum_{t=P+1}^{N} e_t^2 \\ - \sum_{t=P+1}^{N} \alpha_t \left(\sum_{k=0}^{P} \boldsymbol{w}_k^T \boldsymbol{\varphi}(\boldsymbol{x}_{t-k}) + \bar{b} + e_t - y_t - \sum_{k=1}^{P} d_k y_{t-k} \right). \end{aligned} \tag{8.10}$$

The KKT conditions are similar to those of Lemma 7.1 and also extensions of the ones found in Lemma 4.2. The most important conditions are

$$\boldsymbol{0}_{n_h} = \frac{\partial \mathscr{L}}{\partial \boldsymbol{w}_k} = \boldsymbol{w}_k - \sum_{t=P+1}^{N} \alpha_t \boldsymbol{\varphi}(\boldsymbol{x}_{t-k}), \quad k = 0, \dots, P, \tag{8.11a}$$

$$0 = \frac{\partial \mathscr{L}}{\partial d_k} = \sum_{t=P+1}^{N} \alpha_t y_{t-k}, \quad k = 1, \dots, P, \tag{8.11b}$$

Substituting the above relations for $\boldsymbol{w}_k$ and $e_k = \gamma^{-1}\alpha_k$ into the modeling constraint yields

$$\sum_{k=0}^{P} \sum_{n=P+1}^{N} \alpha_n K(\boldsymbol{x}_{n-k}, \boldsymbol{x}_{t-k}) + \bar{b} + \gamma^{-1}\alpha_t = \textstyle\sum_{k=0}^{P} d_k y_{t-k}$$

after applying the kernel trick $K(\boldsymbol{x}_{n-k}, \boldsymbol{x}_{t-k}) = \boldsymbol{\varphi}(\boldsymbol{x}_{n-k})^T \boldsymbol{\varphi}(\boldsymbol{x}_{t-k})$. Expressing this, $\partial\mathscr{L}/\partial\bar{b} = \sum_{t=P+1}^{N} \alpha_t = 0$ and $\partial\mathscr{L}/\partial d_k = 0$ in matrix notation yields (8.9). □

Evaluating the overparametrized model at a new point $(y_{t-1}, \dots, y_{t-P}, \boldsymbol{x}_t, \dots, \boldsymbol{x}_{t-P})$ in terms of the dual variables $\boldsymbol{\alpha}$ and primal variables $\bar{b}$ and $\{d_k\}_{k=1}^{P}$ is done using the one step ahead predictor

$$\hat{y}_t = \sum_{n=P+1}^{N} \alpha_n \sum_{k=0}^{P} K(\boldsymbol{x}_{n-k}^{train}, \boldsymbol{x}_{t-k}) + \bar{b} - \sum_{k=1}^{P} a_k y_{t-k}. \tag{8.12}$$

Note that the solution obtained from (8.9) contains one direct estimate of $\boldsymbol{d}$. However embedded into $\boldsymbol{\alpha}$ is information that is capable of providing a second estimate for $\boldsymbol{d}$. The procedure to exploit this information is closely related to Subsection 7.2.4. However, in the setting here there is an additional complication. The two estimates for $\boldsymbol{d}$ are independent of each other as the collinearity constraint implemented by the rank constraint has been dropped from the problem.

8.2.3 Projection onto original class

The model obtained from (8.9) is only an approximation for the AR(P)-LS-SVM model stated in (8.7) and its rewritten form (8.8). This is due to neglecting the rank constraint which results in the relation $\boldsymbol{W} = \boldsymbol{w}\boldsymbol{d}^T$ not to be satisfied anymore. Hence the model structure is not AR(P)-NARX anymore but a relaxation. Furthermore, as the model has been dualized there is no direct access to $\boldsymbol{W}$ anymore. Note that the Hammerstein systems discussed in Chapter 7 face the same problem. In their case the recovery of the original model structure has been addressed in Subsection 7.2.4. Here the first method described in that subsection is adapted. It is used to obtain a second estimate for $\boldsymbol{d}$ that gives rise to the best (in a least squares sense) rank-1 factorization of $\boldsymbol{W}$.

To recover a rank-1 factorization of $\boldsymbol{W}$ one can make use of its SVD $\boldsymbol{U\Sigma V}^T$ and only consider the dominant singular vectors. The columns of $\boldsymbol{W}$ are given by (8.11a), but involve the usually implicitly defined feature map $\boldsymbol{\varphi}$. As the objective is to estimate a value for $\boldsymbol{d}$ it is however sufficient to consider the eigenvalue decomposition $\boldsymbol{V\Sigma}^2\boldsymbol{V}^T$ of $\boldsymbol{W}^T\boldsymbol{W}$. Scaling the dominant eigenvector such that its first component is one, gives an estimate for $\boldsymbol{d}$. Applying the kernel trick on $\boldsymbol{W}^T\boldsymbol{W}$ one obtains

$$\begin{bmatrix} \boldsymbol{\alpha}^T\boldsymbol{\Omega}_{0,0}\boldsymbol{\alpha} & \cdots & \boldsymbol{\alpha}^T\boldsymbol{\Omega}_{0,P}\boldsymbol{\alpha} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\alpha}^T\boldsymbol{\Omega}_{P,0}\boldsymbol{\alpha} & \cdots & \boldsymbol{\alpha}^T\boldsymbol{\Omega}_{P,P}\boldsymbol{\alpha} \end{bmatrix}$$

with $(\boldsymbol{\Omega}_{k,l})_{ij} = K(\boldsymbol{x}_{i-k}, \boldsymbol{x}_{j-l})$ for $k, l = 0, \dots, P$ and $i, j = P+1, \dots, N$.

Remark 8.1. The projection onto the class AR(P)-LS-SVM is incomplete as two independent estimates for d_k are obtained, one following directly from the solution of the dual system (8.9) and the other from the rank-1 approximation of $\boldsymbol{W}$ as outlined in this section. Therefore both estimates will be compared with respected to their predictive performance in the experimental section.

Algorithm 8.1 (Overparametrized model (OVER))**.**
TRAINING:
1. compute kernel matrix $\boldsymbol{\Omega} = \sum_{k=0}^{P} \boldsymbol{\Omega}_k$
2. solve (8.9) to obtain estimates for $\boldsymbol{\alpha}$, b and $\boldsymbol{a}$

PREDICTION:
Generate estimates with the predictor given by (8.12).

Algorithm 8.2 (Model with direct estimate for the noise model (DIRECT))**.**
TRAINING:
1. compute kernel matrix $\boldsymbol{\Omega} = \sum_{k=0}^{P} \boldsymbol{\Omega}_k$
2. solve (8.9) to obtain estimates for $\boldsymbol{\alpha}$, b and $\boldsymbol{a}$, denote the estimate for $\boldsymbol{a}$ by $\hat{\boldsymbol{a}}_{LS}$
3. compute final model by solving (8.2) given $\hat{\boldsymbol{a}}_{LS}$

PREDICTION:
Estimates are generated according to (8.6)

Algorithm 8.3 (Model with projection based estimate for the noise model (SVD))**.**
TRAINING:
1. compute kernel matrix $\boldsymbol{\Omega} = \sum_{k=0}^{P} \boldsymbol{\Omega}_k$
2. solve (8.9) to obtain estimates for $\boldsymbol{\alpha}$, b and $\boldsymbol{a}$
3. compute $\boldsymbol{W}^T\boldsymbol{W}$ as in Section 8.2.3
4. $\sigma_0, \boldsymbol{v}_0 \leftarrow$ largest eigenvalue and eigenvector of $\boldsymbol{W}^T\boldsymbol{W}$
5. $\hat{\boldsymbol{a}}_{SVD} \leftarrow \boldsymbol{v}_0/(\boldsymbol{v}_0)_0$
6. compute final model by solving (8.2) given $\hat{\boldsymbol{a}}_{SVD}$

PREDICTION:
Estimates are generated according to (8.6)

This results in three possible algorithms to obtain a predictive model. The first possibility is described in Algorithm 8.1 and uses the overparametrized model for projections. The second model uses the direct estimate for $\boldsymbol{a}$ to estimate an AR(P) model as explained in Algorithm 8.2. Finally another AR(P) model can be obtained by using the estimate for $\boldsymbol{a}$ obtained from the projection. This is outlined in Algorithm 8.3.

8.3 Numerical experiments

The RBF kernel is used for all considered models. Model selection is performed using an independent validation set. The regularization parameter γ and the kernel bandwidth σ are selected using grid search. Performance measures are reported on independent test sets in both cases.

As synthetic examples the nonlinear systems given in [Espinoza et al., 2005b] are considered,

1. $y_t = f_1(u_t) = 0.2(1 - 6u_t + 36u_t^2 - 53u_t^3 + 22u_t^5) + r_t$ with u_t uniformly distributed on $[-0.5, 1.3]$ and
2. $y_t = f_2(y_{t-1}) = \text{sinc}(y_{t-1}) + r_t$.

The noise term r_t is generated with a linear AR(P) noise model according to $r_t = \sum_{p=1}^{P} a_k r_{t-k} + e_t$. For the experiments models of order $P = 2p$ are used, with p pairs of conjugate complex poles on the unit disc and gain one. The excitation signal e_t is white Gaussian noise with standard deviation $\sigma_r = 0.3$.

8.3.1 Model order selection

In Figure 8.2 the validation performance of an overparametrized model is shown as a function of the model order P. The two particular examples are generated for f_1 and show that a model order can be selected based on the validation performance. Yet it is not necessarily the case that the true model order is revealed. From these simple experiments it appears that the model order tends to be underestimated.

An alternative approach to the method outlined here is presented in [De Brabanter et al., 2011]. There a cross-validation scheme is modified such that good hyper-parameters are selected for a standard LS-SVM formulation in the presence of correlated errors. The advantage is that no knowledge about the noise structure or its order is necessary a priori. The disadvantage is that no information on the noise process is gathered and as such the noise characteristic can for example not be taken into account in simulated predictions.

8.3.2 Correlation of estimated parameters with true noise model

Solving (8.9) one obtains $\hat{\boldsymbol{a}}_{LS}$ as an estimate for a_k. Projecting the model as described in Section 8.2.3 yields a second estimate for a_k which is denoted by $\hat{\boldsymbol{a}}_{SVD}$. To assess the quality of the overparametrized model, several quantities are investigated

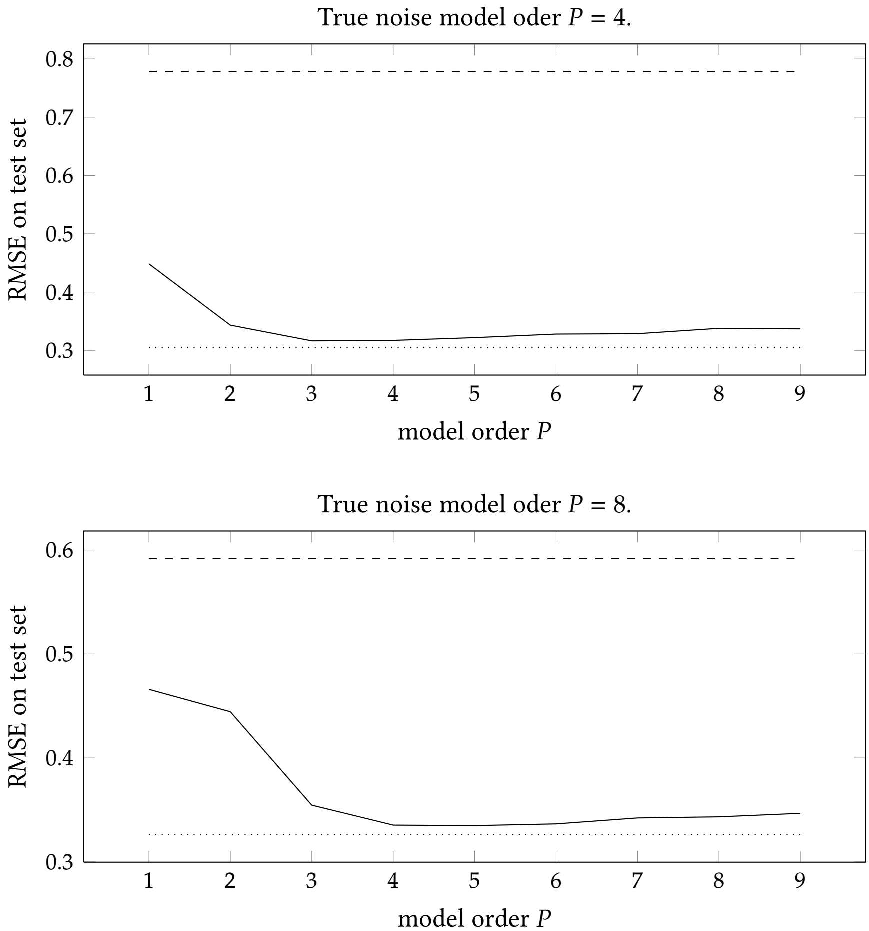

Figure 8.2: Validation performance as a function of the noise model order P. Tested for f_1. The solid line is the validation performance of an overparametrized model of order P. The dotted line gives the performance of an AR(P)-LS-SVM model with the true noise model while the dashed line indicates the performance of a standard LS-SVM model.

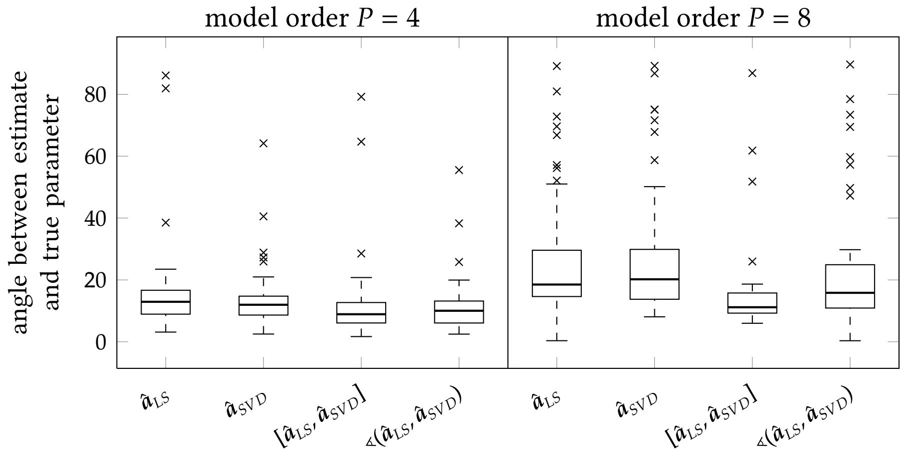

Figure 8.3: Correlation of true noise model parameters $\boldsymbol{a}$ with estimates $\hat{\boldsymbol{a}}_{LS}$ and $\hat{\boldsymbol{a}}_{SVD}$ based on 50 Monte Carlo simulations for f_1.

1. the agreement of the two different estimates, given by the angle between them $\sphericalangle(\hat{\boldsymbol{a}}_{LS}, \hat{\boldsymbol{a}}_{SVD})$,
2. the distance of true parameters to the plane spanned by the estimates $\sphericalangle(\boldsymbol{a}, [\hat{\boldsymbol{a}}_{LS}, \hat{\boldsymbol{a}}_{SVD}])$ and
3. the individual agreements between the true noise model and its estimates $\sphericalangle(\boldsymbol{a}, \hat{\boldsymbol{a}}_{LS})$, $\sphericalangle(\boldsymbol{a}, \hat{\boldsymbol{a}}_{SVD})$.

Figures 8.3 and 8.4 show the result of 50 Monte Carlo simulations with different realizations of the noise model for orders $P = 4$ and $P = 8$. Figure 8.3 depicts results for f_1 while Figure 8.4 shows results obtained with f_2. Especially for the lower order models the correlation of the different quantities are mostly below 10 degrees. Even for a model order of $P = 8$ the median angle is about 20 degrees which corresponds to a correlation coefficient of about 0.94. It seems that with the overparametrized formulation, the true noise model coefficients cannot be recovered. Yet the approximation is good enough to obtain predictive models that significantly outperform standard LS-SVM as shown in the next section.

8.3.3 Performance of projected models

The same experiments as in the previous section are performed but now the prediction performance is analyzed. The compared projection schemes are:

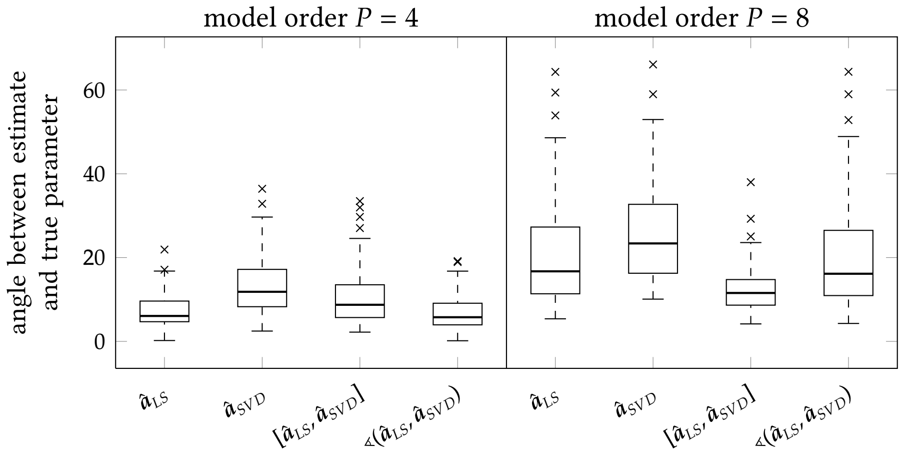

Figure 8.4: Correlation of true noise model parameters $\boldsymbol{a}$ with estimates $\hat{\boldsymbol{a}}_{LS}$ and $\hat{\boldsymbol{a}}_{SVD}$ based on 50 Monte Carlo simulations for f_2.

LS-SVM standard LS-SVM without noise model,
AR(P) AR(P)-LS-SVM given the true noise model,
OVER overparametrized LS-SVM (Algorithm 8.1),
DIRECT AR(P)-LS-SVM with $\hat{\boldsymbol{a}}_{LS}$ estimate (Algorithm 8.2) and
SVD AR(P)-LS-SVM with $\hat{\boldsymbol{a}}_{SVD}$ estimate (Algorithm 8.3).

Results for Monte Carlo simulations are shown in Figure 8.5.

One can observe that the AR(P)-LS-SVM significantly outperforms standard LS-SVMs in a lot of cases. The overparametrized model is much better than LS-SVMs but does not perform as well as AR(P)-LS-SVM with the true parameters. For the projected model the estimate obtained by (8.9) is much more reliable than the one obtained by the rank one approximation. In most cases the projected model slightly outperforms the overparametrized model.

8.3.4 Projection quality

For the models evaluated in the previous sections, one can also analyze the quality of the projection step. As a measure to assess how close $\boldsymbol{W}$ is to rank-1 the ratio of the largest singular value to the power in the whole matrix is used, i.e. $\|\boldsymbol{W}\|_2/\|\boldsymbol{W}\|_F$. Thus a value close to one in Figure 8.6 corresponds to a matrix that is close to rank one. Based on that figure, one can conclude that most of the energy is successfully concentrated in the largest singular value.

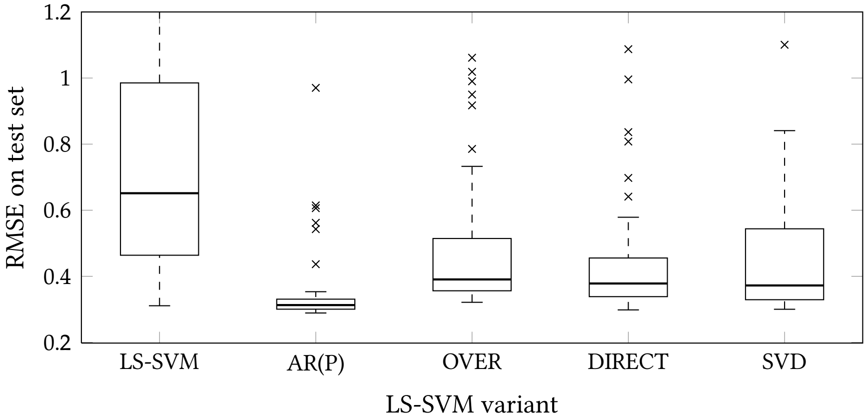

Figure 8.5: Performance of different model structures (cf. Section 8.3.3) evaluated for different nonlinearities in 50 Monte Carlo runs. The true noise model order for the used nonlinearity f_2 is $P = 8$.

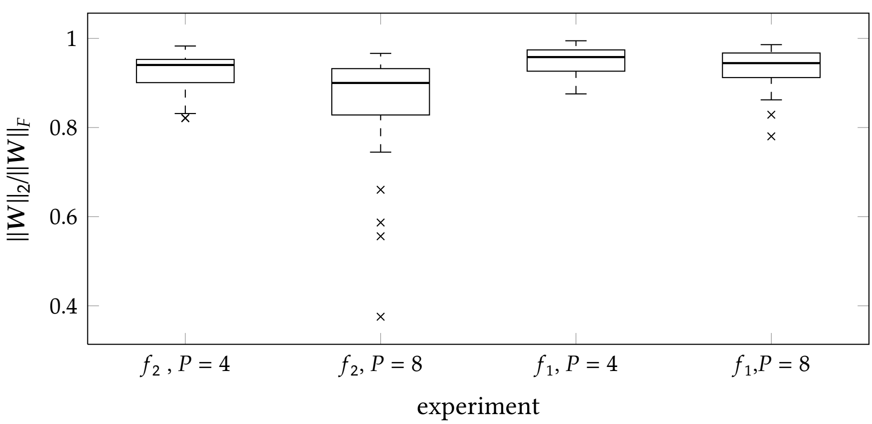

Figure 8.6: Quality measure for the rank of W. Values close to one indicate a solution dominated by the largest singular value. Results are given for both nonlinear models and different noise model orders and compared for 50 Monte Carlo simulations.

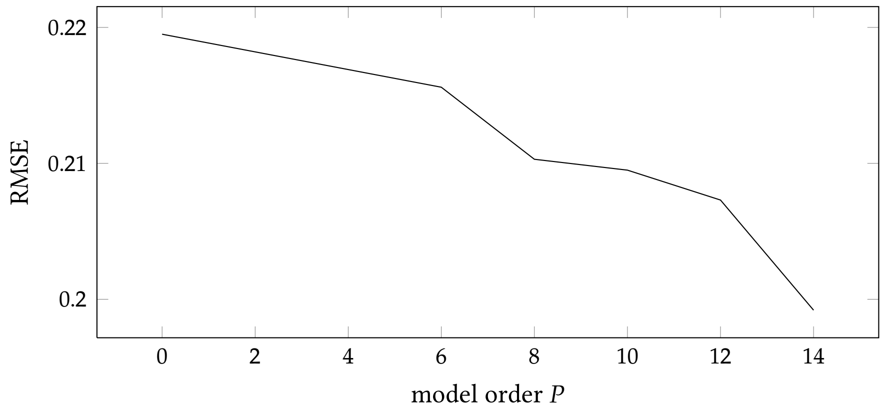

Figure 8.7: Test performance for ESTSP08 [Lendasse et al., 2010] data set 2. The RMSE is reported on an independent test set. The model order 0 corresponds to a LS-SVM model.

8.3.5 Real data

For a test on real data the second data set from the ESTSP08 benchmark [Lendasse et al., 2010] is used. The data set has one variable and contains 1300 hourly measurements of internet traffic in an academic network. To select the model order, a standard LS-SVM model with $\boldsymbol{x}_t = [y_{t-1}, \dots, y_{t-m}]^T$ for $m = 1, \dots, 30$ is trained. The model with order $m = 17$ has the smallest validation error and is used as basis for tests with additional noise models. Figure 8.7 shows the performance on the last 10% of the data. These have not been used for estimating or selecting the model. It can be seen that the performance on this independent test set can be improved by considering a noise model.

8.4 Conclusions

It has been shown how to integrate a noise model with LS-SVM based models and that doing so is beneficial in presence of colored noise. For the case that the noise model is not known a priori a novel convex relaxation, based on overparametrization to solve the otherwise nonconvex problem, has been proposed. This makes it viable to identify high order noise models without a significant increase in computational complexity. The identified coefficients of the noise model clearly deviate from the true parameters. Nevertheless the prediction capability of the identified models is superior to standard LS-SVM and can, in some cases, come close to the performance of a model given the true noise model parameters. Finally the applicability on a real world data set has been demonstrated.

9 Sensitivity of kernel based models

Based on the publication Falck, T., Pelckmans, K., Suykens, J. A. K., and De Moor, B. (July 2009). "Identification of Wiener-Hammerstein Systems using LS-SVMs". In: *Proceedings of the 15th IFAC Symposium on System Identification*. (Saint-Malo, France, July 6–8, 2009), pp. 820–825.

This thesis primarily discusses nonlinear, implicitly defined and nonparametric models. An inherent problem of these black-box models is that they offer only very limited insight about their properties besides predictive performance. This chapter attempts to gather some more information of a model in terms of its sensitivity to certain input variables. Due to the complex nature of the problem, even this limited analysis relies on several approximations. One advantage of the methodology described here is that it will result in a predictive model that is able to generate predictions for the case where the input data is not known exactly but only known to be within an interval.

The work is inspired by results in robust regression [El Ghaoui and Lebret, 1997; Chandrasekaran et al., 1999] and is in essence an approximate extension of these methods to nonlinear and nonparametric models. In case of the linear models considered in these references there is a close relation to classical regularization schemes as used in Total Least Squares (TLS) [Van Huffel and Vandewalle, 1991]. An iterative algorithm to solve TLS like problems in a nonlinear setting has for example been proposed by Rosen et al. [1998]. This chapter is however more similar to [Watson, 2003, 2007] and differs mainly by its use of kernel based models. The primary new contribution is to show that the primal SOCP can be recast as another cone optimization problem in the dual that depends only on the kernel function.

Further related work with regard to robust solutions for SVMs in terms of SOCP problems are given in [Shivaswamy et al., 2006; Huang et al., 2012] in a probabilistic setting while Trafalis and Gilbert [2006] look at the deterministic case. A result for the related TLS problem in a LS-SVM context can be found in [Renault et al., 2005].

In addition to stating the computationally intensive SOCP problem, this chapter shows how a related least squares problem can be constructed and how it is connected to the original SOCP formulation. This allows for a more efficient solution of the problem and avoids the need to investigate advanced optimization schemes like those considered in some of the previous chapters.

Structure of the chapter The following section briefly restates LS-SVMs as an equivalent SOCP problem. This enables Section 9.2 to derive a kernel based regression model that is robust with respect to bounded perturbations. To facilitate an efficient solution Section 9.3 reverts the robustified SOCP problem back to a least squares estimation problem. Due to the large size of the problem, some approximations to cope with large amounts of data are tailored to the specific needs in Section 9.4. Before concluding remarks in the last section, Section 9.5 presents results on several numerical examples.

9.1 LS-SVM models in SOCP form

In this section several results relevant later on in this chapter are derived. They establish an equivalence between the solutions of standard LS-SVMs on the one hand and a modified version on the other hand. Here standard LS-SVM denotes the formulation introduced in Chapter 4 which is in QP form and in particular a least squares problem. The modified formulation in essence changes the objective function from squared norms to their unsquared counterparts, yielding a SOCP problem. For reference the dual problem as well as the form of the predictive model in terms of the dual variables are given for the SOCP based LS-SVM model.

Lemma 9.1 (Kernel based model in SOCP form)**.** *The kernel based estimation problem*

$$\begin{aligned} &\min_{w,b,e_t} \quad \|\boldsymbol{w}\|_2 + \gamma' \|\boldsymbol{e}\|_2 \\ &\text{subject to} \quad y_t = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t, \quad t = 1, \dots, N, \end{aligned} \tag{9.1}$$

with $\mathbf{e} = [e_1, \dots, e_N]^T$ *is equivalent to a LS-SVM model* (4.2) *for* $\gamma' = \gamma\|\mathbf{e}\|_2/\|\mathbf{w}\|_2$.

Proof. Substitution of γ' into (9.1) and scaling the objective by $\frac{1}{2}\|\mathbf{w}\|_2$ yields (4.2). □

Lemma 9.2 (Dual of kernel based model in SOCP form). *The dual of the kernel based model introduced in Lemma 9.1 is*

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^T \mathbf{y} \\ \text{subject to} \quad & \mathbf{1}_N^T \boldsymbol{\alpha} = 0, \\ & \|\mathbf{G}\boldsymbol{\alpha}\|_2 \leq 1, \|\boldsymbol{\alpha}\|_2 \leq \gamma', \end{aligned} \tag{9.2}$$

where $\mathbf{y} = [y_1, \dots, y_N]^T$ *and* $\boldsymbol{\alpha} \in \mathbb{R}^N$ *contains the Lagrange multipliers of the equality constraints in* (9.1). *Furthermore* $\mathbf{G}$ *is a matrix square root of the kernel matrix* $\boldsymbol{\Omega}$ *such that* $\boldsymbol{\Omega} = \mathbf{G}^T\mathbf{G}$.

Proof. Using conic constraints (9.1) can be written as

$$\begin{aligned} \min_{\mathbf{w}, b, e_t, v_1, v_2} \quad & v_1 + \gamma' v_2 \\ \text{subject to} \quad & y_t = \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_t) + b + e_t, \quad t = 1, \dots, N, \\ & \|\mathbf{w}\|_2 \leq v_1, \|\mathbf{e}\|_2 \leq v_2. \end{aligned} \tag{9.3}$$

Let $\boldsymbol{\Phi} = [\boldsymbol{\varphi}(\mathbf{x}_1), \dots, \boldsymbol{\varphi}(\mathbf{x}_N)]$, then the Lagrangian of the above optimization problem can be written as

$$\begin{aligned} \mathscr{L}(\mathbf{w}, b, e_t, v_1, v_2, \mathbf{u}_1, \mathbf{u}_2, v_1', v_2', \boldsymbol{\alpha}) &= v_1 + \gamma' v_2 \\ &+ \mathbf{u}_1^T \mathbf{w} - v_1 v_1' + \mathbf{u}_2^T \mathbf{e} - v_2 v_2' - \boldsymbol{\alpha}^T(\boldsymbol{\Phi}^T \mathbf{w} + b\mathbf{1}_N + \mathbf{e} - \mathbf{y}), \end{aligned} \tag{9.4}$$

with $\|\mathbf{u}_1\|_2 \leq v_1'$ and $\|\mathbf{u}_2\|_2 \leq v_2'$. Computing the KKT conditions for optimality one obtains

$$\begin{aligned} &\frac{\partial \mathscr{L}}{\partial \mathbf{w}} = \mathbf{0}_{n_h} \Rightarrow \mathbf{u}_1 = \boldsymbol{\Phi}\boldsymbol{\alpha}, \\ &\frac{\partial \mathscr{L}}{\partial \mathbf{e}} = \mathbf{0}_N \Rightarrow \mathbf{u}_2 = \boldsymbol{\alpha}, \\ &\frac{\partial \mathscr{L}}{\partial v_1} = 0 \Rightarrow v_1' = 1, \\ &\frac{\partial \mathscr{L}}{\partial v_2} = 0 \Rightarrow v_2' = \gamma', \\ &\frac{\partial \mathscr{L}}{\partial b} = 0 \Rightarrow \mathbf{1}_N^T \boldsymbol{\alpha} = 0. \end{aligned}$$

Substitution of these relations into the Lagrangian reduce it to $\boldsymbol{\alpha}^T\boldsymbol{y}$. This gives the objective of the dual problem. The equality constraint is directly taken from the KKT conditions. The norm $\|\boldsymbol{\alpha}\|_2 \leq \gamma'$ is a consequence of substituting the KKT conditions in the dual conic constraint $\|\boldsymbol{u}_2\|_2 \leq v_2'$. The second conic constraint follows in the same way and is $\|\boldsymbol{\Phi\alpha}\|_2 \leq 1$ after substitution. To obtain a kernel based expression one can first square the constraint to obtain $\boldsymbol{\alpha}^T\boldsymbol{\Omega\alpha} \leq 1$ where $\boldsymbol{\Omega} = \boldsymbol{\Phi}^T\boldsymbol{\Phi}$ is the kernel matrix. Finally using a matrix square root, the constraint can be converted back into a norm constraint. □

Corollary 9.3. *(Model representation for kernel model in SOCP form) Let $\boldsymbol{\alpha}$ denote the dual variables computed from Lemma 9.2 and assume that the constraints of* (9.2) *are active. Then the form of the predictive model for a kernel model in SOCP form as specified in Lemma 9.1 is given by*

$$\hat{y}(z) = v_1 \sum_{t=1}^{N} \alpha_t K(\boldsymbol{x}_t, \boldsymbol{z}) + b \tag{9.5}$$

where $b = N^{-1}(\boldsymbol{1}_N^T\boldsymbol{y} - v_1\boldsymbol{1}_N^T\boldsymbol{\Omega\alpha})$, $v_1 = \boldsymbol{\alpha}^T\boldsymbol{y} - \gamma'^2 v_2$ and v_2 can be determined from $\boldsymbol{y} - \boldsymbol{P}_\alpha\boldsymbol{y} - \boldsymbol{P}_1\boldsymbol{y} = v_2(\boldsymbol{\Omega} - \frac{1}{\gamma'^2}\boldsymbol{I}_N - \boldsymbol{P}_1\boldsymbol{\Omega})\boldsymbol{\alpha}$, where $\boldsymbol{P}_\alpha = \frac{\boldsymbol{\alpha\alpha}^T}{\boldsymbol{\alpha}^T\boldsymbol{\alpha}}$ and $\boldsymbol{P}_1 = \frac{1}{N}\boldsymbol{1}_N\boldsymbol{1}_N^T$.

Proof. Note that one has $\boldsymbol{u}_1 = \boldsymbol{\Phi\alpha}$ and $\boldsymbol{u}_2 = \boldsymbol{\alpha}$. Due to complementary slackness and since the constraints are active, it holds that $\boldsymbol{u}_1^T\boldsymbol{w} - v_1v_1' = 0$ and $\boldsymbol{u}_2^T\boldsymbol{e} - v_2v_2' = 0$. Combining both relations one obtains $\boldsymbol{w} = v_1\boldsymbol{\Phi\alpha} + \boldsymbol{w}_\perp$ and $\boldsymbol{e} = v_2\boldsymbol{\alpha} + \boldsymbol{e}_\perp$, where $\boldsymbol{w}_\perp$ and $\boldsymbol{e}_\perp$ are arbitrary vectors that satisfy $\boldsymbol{u}_1^T\boldsymbol{w}_\perp = 0$ and $\boldsymbol{u}_2^T\boldsymbol{e}_\perp = 0$, respectively. Hence, the primal objective can be written as $v_1\|\boldsymbol{\Phi\alpha}\|_2 + \gamma' v_2\|\boldsymbol{\alpha}\|_2 + \|\boldsymbol{w}_\perp\|_2 + \|\boldsymbol{e}_\perp\|_2 = v_1 + \gamma'^2 v_2 + \|\boldsymbol{w}_\perp\|_2 + \|\boldsymbol{e}_\perp\|_2$. Similarly the equality constraint becomes $\boldsymbol{y} = v_1\boldsymbol{\Omega\alpha} + \boldsymbol{1}_N b + v_2\boldsymbol{\alpha} + \boldsymbol{\Phi}^T\boldsymbol{w}_\perp + \boldsymbol{e}_\perp$. In particular this also has to hold if both sides are multiplied by $\boldsymbol{\alpha}^T$. Therefore $\boldsymbol{\alpha}^T\boldsymbol{y} = v_1 + \gamma'^2 v_2$. Assuming that the duality gap is zero, one obtains that $\boldsymbol{w}_\perp$ and $\boldsymbol{e}_\perp$ are equal to zero.

Furthermore multiplying the equality constraint by $\boldsymbol{1}_N^T$ one obtains $\boldsymbol{1}_N^T\boldsymbol{y} = v_1\boldsymbol{1}_N^T\boldsymbol{\Omega\alpha} + Nb$. Solving this relation for b and $\boldsymbol{\alpha}^T\boldsymbol{y} = v_1 + \gamma'^2 v_2$ for v_2 and substituting those values into the equality constraint, one obtains the relations for b and v_1. □

9.2 Robust kernel based regression

The material in this chapter is different from the rest of this thesis in that it considers a setting in which the regression variable $\boldsymbol{x}$ is not known exactly.

Instead it is assumed that the regression variable is within a simple set, in particular a norm ball. Then the objective is to find the best model when always picking the worst possible element.

9.2.1 Problem setting

Based on a set of measurements $\{(\tilde{x}_t, y_t)\}_{t=1}^N$, where the inputs $\tilde{\boldsymbol{x}}_t = \boldsymbol{x}_t + \boldsymbol{\delta}_t \in \mathbb{R}^d$ are corrupted by unstructured perturbations $\boldsymbol{\delta}_t$ and the outputs $y_t \in \mathbb{R}$ are subject to additive noise e_t with bounded variance, a nonlinear model shall be estimated. Within the scope of this chapter it is assumed that these perturbations are bounded $\|\boldsymbol{\delta}_t\|_2 \leq \varrho$ for all t where ϱ is a given value. The form of the estimator $\hat{y} : \mathbb{R}^d \to \mathbb{R}$ is given by

$$\hat{y}(\tilde{x}) = \boldsymbol{w}^T \boldsymbol{\varphi}(\tilde{x}) + b \tag{9.6}$$

with the model parameters $\boldsymbol{w} \in \mathbb{R}^{n_h}$ and $b \in \mathbb{R}$ and the feature map $\boldsymbol{\varphi} : \mathbb{R}^d \to \mathbb{R}^{n_h}$.

The estimation objective is to fit the model in (9.6) to the perturbed data as formalized by the following optimization problem

$$\begin{aligned} \min_{\boldsymbol{w}, b, e_t, \boldsymbol{\delta}_t} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{t=1}^N e_t^2 + \sum_{t=1}^N l(\boldsymbol{\delta}_t) \\ \text{subject to} \quad & y_t = \boldsymbol{w}^T \boldsymbol{\varphi}(\tilde{x}_t - \boldsymbol{\delta}_t) + b + e_t, \quad t = 1, \dots, N. \end{aligned} \tag{9.7}$$

In case of a TLS-like setting the loss $l(\boldsymbol{\delta})$ would be $\frac{1}{2}\|\boldsymbol{\delta}\|_2^2$. Solving this problem is difficult as the constraints in (9.7) are nonlinear in $\boldsymbol{\delta}_t$ and possibly nonconvex. If the $\boldsymbol{x}_t$ were unperturbed, i.e. $\boldsymbol{\delta}_t = \boldsymbol{0}$, this problem corresponds to standard LS-SVM regression, c.f. Chapter 4.

9.2.2 Linearization

To make the problem tractable it has to be simplified. In the scope of this chapter the goal is to solve a related convex problem. Therefore the feature map $\boldsymbol{\varphi}$ is linearized with respect to the perturbations $\boldsymbol{\delta}$ on its argument $\tilde{\boldsymbol{x}}$

$$\boldsymbol{\varphi}(\tilde{\boldsymbol{x}} - \boldsymbol{\delta}) \simeq \boldsymbol{\varphi}(\tilde{\boldsymbol{x}}) - \sum_{i=1}^d (\partial_{x_i}\boldsymbol{\varphi})(\tilde{\boldsymbol{x}})\delta_i \tag{9.8}$$

using the shorthand notation $(\partial_x f)(x_0, y_0) = \left.\frac{\partial f(x,y)}{\partial x}\right|_{x_0,y_0}$. Also define $\boldsymbol{\Phi}_t' = [(\partial_{x_1}\boldsymbol{\varphi})(\tilde{\boldsymbol{x}}_t), \dots, (\partial_{x_d}\boldsymbol{\varphi})(\tilde{\boldsymbol{x}}_t)] \in \mathbb{R}^{n_h \times d}$. Then the modeling constraint in (9.7) can be compactly approximated as

$$y_t \simeq \boldsymbol{w}^T \boldsymbol{\varphi}(\tilde{\boldsymbol{x}}_t) - \boldsymbol{w}^T \boldsymbol{\Phi}_t' \boldsymbol{\delta}_t + b + e_t. \tag{9.9}$$

The term $\boldsymbol{w}^T \boldsymbol{\Phi}_t' \boldsymbol{\delta}_t$ is bilinear in the unknowns $\boldsymbol{w}$ and $\boldsymbol{\delta}_t$ and thus the corresponding optimization problem is still nonconvex.

For the integration in a kernel based model the derivatives of the feature map have to be expressed in terms of the kernel function [Lázaro et al., 2005]. The theoretical foundation as well as an example for the Gaussian RBF kernel are given in the following.

Lemma 9.4 (Kernelizing derivatives of the feature map). *For any positive definite kernel* $K : \mathbb{R}^D \times [a,b]^D \to [a,b]^D$ *and* $\boldsymbol{x}, \boldsymbol{y} \in [a,b]^D$ *for* $a, b \in \mathbb{R}$ *and* $a < b$ *one has*

$$\frac{\partial}{\partial x_i} K(\boldsymbol{x}, \boldsymbol{y}) = \left\langle \frac{\partial}{\partial x_i} \boldsymbol{\varphi}(\boldsymbol{x}), \boldsymbol{\varphi}(\boldsymbol{y}) \right\rangle, \tag{9.10}$$

$$\frac{\partial^2}{\partial x_i \partial y_j} K(\boldsymbol{x}, \boldsymbol{y}) = \left\langle \frac{\partial}{\partial x_i} \boldsymbol{\varphi}(\boldsymbol{x}), \frac{\partial}{\partial y_j} \boldsymbol{\varphi}(\boldsymbol{y}) \right\rangle. \tag{9.11}$$

Proof. According to Mercer's theorem a positive definite kernel can be written as $K(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{\infty} \varphi_i(\boldsymbol{x}) \varphi_j(\boldsymbol{y})$. Furthermore the series is uniformly and absolutely convergent. Therefore the differentiation can be performed elementwise. □

Remark 9.1 (Derivative of Gaussian RBF kernel). The Gaussian RBF kernel has the following derivatives:

$$\frac{\partial}{\partial x_i} K(\boldsymbol{x}, \boldsymbol{y}) = -\frac{2}{\sigma^2}(x_i - y_i) K(\boldsymbol{x}, \boldsymbol{y}), \tag{9.12}$$

$$\frac{\partial}{\partial x_i} \frac{\partial}{\partial y_j} K(\boldsymbol{x}, \boldsymbol{y}) = \begin{cases} -\frac{4}{\sigma^4}(x_i - y_i)(x_j - y_j) K(\boldsymbol{x}, \boldsymbol{y}), & i \neq j \\ -\frac{4}{\sigma^4}(x_i - y_i)^2 K(\boldsymbol{x}, \boldsymbol{y}) + \frac{2}{\sigma^2} K(\boldsymbol{x}, \boldsymbol{y}), & i = j \end{cases} \tag{9.13}$$

9.2.3 Convexification

In the literature for robust solutions of linear systems [El Ghaoui and Lebret, 1997] similar problems are solved by robustifying with respect to bounded perturbations $\|\boldsymbol{\delta}_k\|_2 \leq \varrho$. Based on the equivalence established in Section 9.1 and for the sake of this argument, a related ℓ_2-problem will be considered instead of the least squares problem (9.7) .

For the worst case scenario over all $\|\boldsymbol{\delta}_k\| \leq \varrho$, the ℓ_2-formulation of (9.7) with the linearized modeling constraint (9.9) is given by

$$\begin{aligned}
&\min_{w,b,e_k} \max_{\|\boldsymbol{\delta}_k\|_2 \leq \varrho} \quad \|\boldsymbol{w}\|_2 + \gamma\|\boldsymbol{e}\|_2 \\
&\text{subject to} \quad y_k = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_k) - \boldsymbol{w}^T \boldsymbol{\Phi}'_k \boldsymbol{\delta}_k + b + e_k, \quad k = 1, \dots, N.
\end{aligned} \tag{9.14}$$

Based on El Ghaoui and Lebret [1997, Theorem 3.1] the following lemma states a convex SOCP problem that approximates (9.14). In the original result by El Ghaoui and Lebret for linear systems it can even be shown that the relation is exact.

Lemma 9.5 (Robust SOCP). *An upper bound for the solution of* (9.14) *can be obtained by adding an additional regularization term to the objective function*

$$\begin{aligned}
&\min_{w,b,e_t} \quad \|\boldsymbol{w}\|_2 + \gamma\|\boldsymbol{e}\|_2 + \gamma\varrho\|\boldsymbol{\Phi}'^T \boldsymbol{w}\|_2 \\
&\text{subject to} \quad y_t = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t, \quad t = 1, \dots, N.
\end{aligned} \tag{9.15}$$

Proof. Consider the subproblem

$$\begin{aligned}
&\max_{\|\boldsymbol{\delta}_t\|_2 \leq \varrho} \quad \|\boldsymbol{e}\|_2 \\
&\text{subject to} \quad e_t = y_t - \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + \boldsymbol{w}^T \boldsymbol{\Phi}'_t \boldsymbol{\delta}_t - b, \quad t = 1, \dots, N
\end{aligned} \tag{9.16}$$

and define the matrices $\boldsymbol{\Phi} = [\boldsymbol{\varphi}(\boldsymbol{x}_1), \dots, \boldsymbol{\varphi}(\boldsymbol{x}_N)]$ and $\boldsymbol{\Phi}' = [\boldsymbol{\Phi}'_1, \dots, \boldsymbol{\Phi}'_N]$. Then an upper bound for the maximum in (9.16) can be computed. Therefore let $[x_t]_{t=1}^N$ define a N dimensional column vector such that the t-th element of the vector is given by x_t. Splitting off contributions independent of $\boldsymbol{\delta}_k$ yields

$$\begin{aligned}
\max_{\|\boldsymbol{\delta}_t\|_2 \leq \varrho} \|\boldsymbol{e}\|_2 &= \max_{\|\boldsymbol{\delta}_t\|_2 \leq \varrho} \left\| \boldsymbol{y} - \boldsymbol{\Phi}^T \boldsymbol{w} - b\mathbf{1} + \left[\boldsymbol{w}^T \boldsymbol{\Phi}'_t \boldsymbol{\delta}_t\right]_{t=1}^N \right\|_2 \\
&\leq \|\boldsymbol{y} - \boldsymbol{\Phi}^T \boldsymbol{w} - b\mathbf{1}\|_2 + \max_{\|\boldsymbol{\delta}_t\|_2 \leq \varrho} \left\| \left[\boldsymbol{w}^T \boldsymbol{\Phi}'_t \boldsymbol{\delta}_t\right]_{t=1}^N \right\|_2 .
\end{aligned}$$

The remainder can be simplified as follows

$$\begin{aligned}
\max_{\|\boldsymbol{\delta}_t\|_2 \leq \varrho} \left\| \left[\boldsymbol{w}^T \boldsymbol{\Phi}'_t \boldsymbol{\delta}_t\right]_{t=1}^N \right\|_2^2 &= \max_{\|\boldsymbol{\delta}_t\|_2 \leq \varrho} \sum_{t=1}^N \left(\boldsymbol{w}^T \boldsymbol{\Phi}'_t \boldsymbol{\delta}_t\right)^2 \\
&\leq \max_{\|\boldsymbol{\delta}_t\|_2 \leq \varrho} \sum_{t=1}^N \|\boldsymbol{w}^T \boldsymbol{\Phi}'_t\|_2^2 \|\boldsymbol{\delta}_t\|_2^2 \leq \varrho^2 \sum_{t=1}^N \|\boldsymbol{w}^T \boldsymbol{\Phi}'_t\|_2^2 = \varrho^2 \|\boldsymbol{\Phi}'^T \boldsymbol{w}\|_2^2.
\end{aligned}$$

Thus an upper bound for (9.16) is given by $||e||_2 + \varrho||\boldsymbol{\Phi}^T \boldsymbol{w}||_2$ subject to $e_t = y_t - \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) - b$. Substitution shows that second chain of inequalities is an equality for $\boldsymbol{\delta}_t = \varrho \operatorname{sign}(\boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b - y_t) \left(\boldsymbol{w}^T \boldsymbol{\Phi}'_t\right) / ||\boldsymbol{w}^T \boldsymbol{\Phi}'_t||_2$. However, the triangle inequality used in the first part remains an upper bound. □

Remark 9.2. It is straightforward to generalize this technique to incorporate simple prior knowledge on the perturbations $\boldsymbol{\delta}_t$. If known a priori that only some components of $\boldsymbol{x}_t$ are perturbed, i.e. that $\boldsymbol{\delta}_t$ is sparse, one should modify the approximation in (9.8) to consider only derivatives in non-sparse components $\sum_{i=1,\delta_i \neq 0}^{d} (\partial_{x_i} \boldsymbol{\varphi})(\boldsymbol{x})\delta_i$. The bounded perturbations may also be used to specify some belief in the accuracy of the components of $\boldsymbol{x}_t$. If known a priori that some components are much more precise than others, define a nonsingular matrix $\boldsymbol{D}$ and maximize (9.14) with respect to $||\boldsymbol{D}\boldsymbol{\delta}_t||_2 \leq \varrho$ instead of the unweighted norm. This is equivalent to solving the original problem with modified derivative information $\overline{\boldsymbol{\Phi}}'_t = \boldsymbol{\Phi}'_t \boldsymbol{D}^{-1}$.

9.3 Least squares kernel based model

9.3.1 Problem statement & solution

The SOCP in (9.15) gives a robust convex approximation to the original problem in (9.7). In kernel based regression, the solution is usually not obtained in the parametric primal formulation but in the nonparametric dual where the kernel function takes the place of the feature map. In that way the very high or even infinite dimensional estimation problem is cast into a problem of estimating a finite number of parameters. Although it is possible to derive a dual formulation of (9.15) in terms of the kernel it is not advisable as the computational costs with solving an SOCP are quite high. Therefore consider the following lemma.

Lemma 9.6. *For*

$$\gamma' = \frac{||\boldsymbol{w}||_2}{||e||_2}\gamma \quad \textit{and} \quad \varrho' = \frac{||e||_2}{||\boldsymbol{\Phi}'^T \boldsymbol{w}||_2}\varrho \tag{9.17}$$

the solution of the SOCP problem (9.15) *coincides with the solution of the following LS-SVM problem*

$$\begin{aligned} \min_{\boldsymbol{w},b,e_k} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma'\varrho'\boldsymbol{w}^T\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T\boldsymbol{w} + \frac{1}{2}\gamma'\sum_{k=1}^{N} e_k^2 \\ \text{subject to} \quad & y_k = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_k) + b + e_k, \qquad k = 1, \dots, N. \end{aligned} \tag{9.18}$$

Proof. Substituting (9.17) into (9.18) yields (9.15). □

Define $\boldsymbol{\Omega}_x = \boldsymbol{\Phi}'^T \boldsymbol{\Phi}$ and $\boldsymbol{\Omega}_{xy} = \boldsymbol{\Phi}'^T \boldsymbol{\Phi}'$. Note that these matrices can then be directly computed using the kernel function and Lemma 9.4. $\boldsymbol{\Omega}_x \in \mathbb{R}^{(Nd)\times N}$ has block structure with blocks defined by $(\boldsymbol{\Omega}_x)_k = \boldsymbol{\omega}_x^{kl} \in \mathbb{R}^{d\times 1}$ where $(\boldsymbol{\omega}_x^{kl})_i = (\partial/\partial x_i K)(\boldsymbol{x}_k, \boldsymbol{x}_l)$. $\boldsymbol{\Omega}_{xy} \in \mathbb{R}^{(Nd)\times(Nd)}$ is also block structured and its blocks are defined by $(\boldsymbol{\Omega}_{xy})_{kl} = \boldsymbol{\Omega}_{xy}^{kl} \in \mathbb{R}^{d\times d}$ where $(\boldsymbol{\Omega}_{xy}^{kl})_{ij} = (\partial^2/(\partial x_i \partial y_j)K)(\boldsymbol{x}_k, \boldsymbol{x}_l)$. Using these definitions the solution for (9.18) is stated in the following lemma.

Lemma 9.7. *For the estimation problem described by* (9.18) *the solution is given in the dual by*

$$\begin{bmatrix} \boldsymbol{\Omega}' + \boldsymbol{I}_N/\gamma' & \boldsymbol{1} \\ \boldsymbol{1}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ 0 \end{bmatrix} \tag{9.19}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^N$ are the dual variables. $\boldsymbol{\Omega}'$ is positive semidefinite and defined as

$$\boldsymbol{\Omega}' = \boldsymbol{\Omega} - \boldsymbol{\Omega}_x^T \left(\boldsymbol{\Omega}_{xy} + (\gamma'\varrho')^{-1} \boldsymbol{I}_{(Nd)}\right)^{-1} \boldsymbol{\Omega}_x. \tag{9.20}$$

Proof. The Lagrangian for (9.18) is

$$\begin{aligned} \mathscr{L}(\boldsymbol{w}, b, e_k, \alpha_k) = {} & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma'\varrho'\boldsymbol{w}^T\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T\boldsymbol{w} \\ & + \frac{1}{2}\gamma' \sum_{k=1}^N e_k^2 - \sum_{k=1}^N \alpha_k \left(\boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_k) + b + e_k - y_k\right) \end{aligned} \tag{9.21}$$

and the corresponding KKT conditions for optimality are

$$\begin{aligned} \boldsymbol{0}_{n_h} &= \frac{\partial \mathscr{L}}{\partial \boldsymbol{w}} \Rightarrow \boldsymbol{\Phi}\boldsymbol{\alpha} = \left(\boldsymbol{I}_{n_h} + \gamma'\varrho'\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T\right)\boldsymbol{w}, \\ 0 &= \frac{\partial \mathscr{L}}{\partial b} \Rightarrow \boldsymbol{1}^T\boldsymbol{\alpha} = 0, \\ 0 &= \frac{\partial \mathscr{L}}{\partial e_k} \Rightarrow \gamma' e_k = \alpha_k, \quad k = 1, \dots, N, \\ 0 &= \frac{\partial \mathscr{L}}{\partial \alpha_k} \Rightarrow \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_k) + b + e_k = y_k, \quad k = 1, \dots, N. \end{aligned}$$

The combination of $\nabla_{\boldsymbol{w}}\mathscr{L} = \boldsymbol{0}$, $\partial_{e_k}\mathscr{L} = 0$ and $\partial_{\alpha_k}\mathscr{L} = 0$ yields

$$\boldsymbol{y} = \boldsymbol{\Phi}^T \left(\gamma'\varrho'\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T + \boldsymbol{I}_{n_h}\right)^{-1} \boldsymbol{\Phi}\boldsymbol{\alpha} + \frac{1}{\gamma'}\boldsymbol{\alpha} + b\boldsymbol{1}. \tag{9.22}$$

The modified kernel matrix (9.20) is a consequence of the matrix inversion lemma applied to the term $\boldsymbol{\Phi}^T \left(\gamma' \varrho' \boldsymbol{\Phi}' \boldsymbol{\Phi}'^T + \boldsymbol{I}_{n_h}\right)^{-1} \boldsymbol{\Phi}$. To show that $\boldsymbol{\Omega}'$ is positive semidefinite, note that $\boldsymbol{\Phi}' \boldsymbol{\Phi}'^T$ is positive semidefinite (psd) as $\boldsymbol{\Omega}_{xy} = \boldsymbol{\Phi}'^T \boldsymbol{\Phi}'$ is psd. Furthermore as $\boldsymbol{\Phi}' \boldsymbol{\Phi}'^T$ is psd, the regularized matrix $\gamma' \varrho' \boldsymbol{\Phi}' \boldsymbol{\Phi}'^T + \boldsymbol{I}_{n_h}$ is positive definite and so is its inverse. Then with $\boldsymbol{z} := \boldsymbol{\Phi} \boldsymbol{x}$ and for any $\boldsymbol{x} \in \mathbb{R}^N$ the following holds

$$\boldsymbol{x}^T \boldsymbol{\Omega}' \boldsymbol{x} = \boldsymbol{x}^T \boldsymbol{\Phi}^T \left(\gamma' \varrho' \boldsymbol{\Phi} \boldsymbol{\Phi}^T + \boldsymbol{I}_{n_h}\right)^{-1} \boldsymbol{\Phi} \boldsymbol{x} = \boldsymbol{z}^T \left(\gamma' \varrho' \boldsymbol{\Phi} \boldsymbol{\Phi}^T + \boldsymbol{I}_{n_h}\right)^{-1} \boldsymbol{z} \geq 0. \tag{9.23}$$

Finally the linear system (9.19) follows from the combination of (9.22) and $\partial_b \mathscr{L} = 0$. □

Remark 9.3. It is possible to express the relations between the regularization constants in (9.17) in terms of the dual variables. The norms of $\boldsymbol{w}$ and $\boldsymbol{e}$ can be rewritten using the expansions following from $\nabla_{\boldsymbol{w}} \mathscr{L} = \boldsymbol{0}$ and $\partial_{e_k} \mathscr{L} = 0$ respectively. The resulting expressions then are

$$\|\boldsymbol{e}\|_2 = \frac{1}{\gamma'} \|\boldsymbol{\alpha}\|_2, \tag{9.24a}$$

$$\|\boldsymbol{\Phi}'^T \boldsymbol{w}\|_2 = \left\| \boldsymbol{\Omega}_{xy} \left(\boldsymbol{\Omega}_{xy} + (\gamma' \varrho')^{-1} \boldsymbol{I}_{(Nd)}\right)^{-1} \boldsymbol{\Omega}_x \boldsymbol{\alpha} - \boldsymbol{\Omega}_x \boldsymbol{\alpha} \right\|_2 \tag{9.24b}$$

and

$$\begin{aligned} \|\boldsymbol{w}\|_2^2 = {} & (\gamma' \varrho')^2 \boldsymbol{\alpha}^T \boldsymbol{\Omega}_x^T \boldsymbol{\Omega}_{xy} \left(\boldsymbol{\Omega}_{xy} + (\gamma' \varrho')^{-1} \boldsymbol{I}_{(Nd)}\right)^{-2} \boldsymbol{\Omega}_x \boldsymbol{\alpha} \\ & + \boldsymbol{\alpha}^T \boldsymbol{\Omega} \boldsymbol{\alpha} - 2 \boldsymbol{\alpha}^T \boldsymbol{\Omega}_x^T \left(\boldsymbol{\Omega}_{xy} + (\gamma' \varrho')^{-1} \boldsymbol{I}_{(Nd)}\right)^{-1} \boldsymbol{\Omega}_x \boldsymbol{\alpha}. \end{aligned} \tag{9.24c}$$

Note that given a solution for a particular pair of γ', ϱ' these expressions only allow to recover the corresponding original constants γ and ϱ. Going from the original constants to the new ones is however not possible. Nevertheless especially the original constant ϱ is of great interest as it carries some direct information on the perturbations. It is the user defined bound on the perturbation in $\|\boldsymbol{\delta}_k\|_2 \leq \varrho$.

9.3.2 Predictive model

For out of sample extensions a predictive equation has to be derived. Therefore the value for $\boldsymbol{w}$ obtained from $\frac{\partial \mathscr{L}}{\partial \boldsymbol{w}} = \boldsymbol{0}_{n_h}$ is substituted into the predictive model

(9.6). For a new point z this yields

$$\hat{y}(z) = \sum_{k=1}^{N} \alpha_k K(\boldsymbol{x}_k, z) + b - \boldsymbol{k}_x^T(z)\left(\boldsymbol{\Omega}_{xy} + (\gamma'\varrho')^{-1}\boldsymbol{I}_{(Nd)}\right)^{-1}\boldsymbol{\Omega}_x\boldsymbol{\alpha} \tag{9.25}$$

where $\boldsymbol{k}_x : \mathbb{R}^d \to \mathbb{R}^{Nd}$ is defined as $\boldsymbol{k}_x(z) = [(\partial/\partial_{x_1}K)(\boldsymbol{x}_1, z), \dots, (\partial/\partial_{x_d}K)(\boldsymbol{x}_N, z)]^T$.

9.4 Numerical implementation

The computational burden of solving the SOCP problem (9.15) has been greatly reduced to solving a linear system (9.19). However, the solution still requires the solution of a very large linear system to compute the modified gram matrix (9.20). To be able to solve a moderately sized problem with for example $N = 1000$ data points and $d = 10$ dimensions, auxiliary linear systems in 10,000 variables have to be solved. Storing as well solving such systems is expensive. Therefore ways to approximate the solution of (9.20) are needed. Note that the matrix $\boldsymbol{\Omega}_{xy}$ is a gram matrix i.e. it contains inner products and is therefore at least positive semidefinite. In the literature several approximation techniques have been proposed for kernel based learning settings like the Nyström approximation [Williams and Seeger, 2001], discussed in Section 4.3.1, or the incomplete Cholesky decomposition [Fine and Scheinberg, 2002]. In the following the Nyström approximation is applied to $\boldsymbol{\Omega}_{xy}$ to reduce the computational complexity.

9.4.1 Optimizations

Given an (approximate) factorization $\boldsymbol{\Omega}_{xy} \simeq \boldsymbol{FDF}^T$ with $\boldsymbol{D}$ diagonal and of size $a \times a$ and $\boldsymbol{F}$ of size $(Nd) \times a$ one can simplify the computation in (9.20) using the matrix inversion lemma, namely $\boldsymbol{\Omega}_x^T(\boldsymbol{\Omega}_{xy} + (\gamma'\varrho')^{-1}\boldsymbol{I}_{(Nd)})^{-1}\boldsymbol{\Omega}_x = \gamma'\varrho'(\boldsymbol{\Omega}_x^T\boldsymbol{\Omega}_x - \boldsymbol{\Omega}_x^T\boldsymbol{F}((\gamma'\varrho')^{-1}\boldsymbol{D}^{-1} + \boldsymbol{F}^T\boldsymbol{F})^{-1}\boldsymbol{F}^T\boldsymbol{\Omega}_x)$.

Depending on the application, the modified Gram matrix in (9.20) has to be computed for many different values of $\gamma'\varrho'$. In that case it is beneficial to transform the factorization into a real eigenvalue decomposition. Therefore the matrix $\boldsymbol{F}$ can be orthogonalized by means of a QR factorization $\boldsymbol{F} = \boldsymbol{QR}$. Then the eigenvalue decomposition is computed of $\boldsymbol{RDR}^T = \tilde{\boldsymbol{U}}\tilde{\boldsymbol{V}}\tilde{\boldsymbol{U}}^T$. Based on this define a new factorization as $\boldsymbol{\Omega}_{xy} = \boldsymbol{UVU}^T$ with $\boldsymbol{V} = \tilde{\boldsymbol{V}}$ and $\boldsymbol{U} = \boldsymbol{Q}\tilde{\boldsymbol{U}}$. Using this factorization the expression for (9.20) simplifies much better to $\boldsymbol{\Omega}_x^T\boldsymbol{U}(\boldsymbol{V} + (\gamma'\varrho')^{-1}\boldsymbol{I}_a)^{-1}\boldsymbol{U}^T\boldsymbol{\Omega}_x$. The remaining matrix inversion is trivial as the matrix is diagonal. The cost of repeated evaluations of $\boldsymbol{\Omega}'$ for different values of $\gamma'\varrho'$ is now reduced to a single matrix multiplication.

9.5 Numerical experiments

In the following section the proposed method is evaluated on several artificial data sets. In total six aspects are analyzed. One subsection is dedicated to the sensitivity of a given kernel based model with respect to its inputs and with respect to the utilized kernel, respectively. One experiment looks at the implications of model sensitivity at a point estimate. The remaining subsections consider more technical aspects. Subsection 9.5.4 shows an example for the dependency between the regularization parameters of the model in SOCP and in LS forms. Finally the proposed approximation scheme is analyzed, once as is and once integrated into the estimation problem.

Model selection The setting outlined above exhibits three parameters that define the model, the bound on the perturbations ϱ, the level of regularization γ and the choice of the kernel K (and its parameters). These parameters have to be selected in some optimal way. The perturbation level ϱ is considered a user defined variable that is given and does not have to be selected. The regularization parameter γ is selected with cross-validation.

To reduce the computational burden, the model selection is carried out on a LS-SVM model (corresponding to $\varrho' = 0$). Then, based on these parameters the effects when increasing ϱ are studied. The procedure is outlined in Algorithm 9.1.

Algorithm 9.1 (Experimental procedure for Section 9.5)**.**

1. estimate a LS-SVM model: solve (9.19) with $\varrho' = 0$
   - select model with best prediction performance on the training set based on cross validation
   - this yields a regularization parameter γ' and a bandwidth σ
2. translate γ' to γ using (9.17) and fix it
3. for ϱ in a set of candidates
   a) determine γ' and ϱ' corresponding to the chosen γ and ϱ using the relations in (9.17), see Remark 9.3 and Section 9.5.4
   b) solve (9.19)
   c) compute predictions using (9.25)

Data generation Let $\{x_k\}_{k=1}^{N+d-1}$ be independent draws from a normal distribution $\mathcal{N}(0,1)$. Then define regressors $\boldsymbol{x}_k = [x_k, \dots, x_{k+d-1}]^T$ with FIR structure and a training set $\mathbb{T} = \{\boldsymbol{x}_k\}_{k=1}^{N}$. Unless stated otherwise the experiments use a

Gaussian RBF kernel. For the matrix approximation experiments in Sections 9.5.5 and 9.5.6 a fixed bandwidth $\sigma = \sqrt{d}$ is used.

9.5.1 Sensitivity of inputs

For experimental design it can be interesting to analyze a given model with respect to which of its input variables are most important to its prediction performance. This can support the decision where to concentrate the effort to acquire good data or aid variable selection. This is done by considering perturbations only on a single input at a time using Remark 9.2. Consider the following two toy systems

(a) $f(\boldsymbol{x}) = \sum_{k=1}^{5}(k-1)\,\mathrm{sinc}(x_k)$ with $N = 1000$ and

(b) $y_l = \mathrm{sinc}(\sum_{k=1}^{7} a_k x_{l-k}) + \sum_{k=1}^{3} b_k y_{l-k}$ with $\boldsymbol{a} = [-6.30, 8.66, 4.15, -6.40, 1.46, 0.77, -2.19]^T$, $\boldsymbol{b} = [-1.54, 1.43, -0.78]^T$ and $N = 1000$.

The importance of the inputs of function (a) is clearly increasing with their index, this is consistent with the analysis as shown in Figure 9.1's top panel. In the bottom panel the dynamical model (b) is analyzed. It can indeed be seen that the model performance depends nonuniformly on the used variables, the nonlinear variables have a much higher weight than the linear ones.

9.5.2 Sensitivity of kernels

In model selection one often selects the model achieving the best prediction performance. In case one has several models achieving a similar performance one can consider trading off performance versus robustness. Therefore sacrificing a bit of performance one can gain stability with a slightly more conservative model. This is demonstrated with the following system $y_l = h(\sum_{k=1}^{5} c_k x_{l-k})$ with $\boldsymbol{c} = [-0.48, 1.67, -1.14, -0.13, 1.87]^T$, $h(x) = x^3 - 3x^2 + x - 5$ and $N = 500$. The nonlinearity is clearly polynomial. Therefore the performance of the polynomial kernel is compared with a Gaussian RBF kernel. Figure 9.2 shows that the best model with a polynomial kernel outperforms the RBF based model, yet the model employing the Gaussian kernel is more robust to perturbation. Depending on the application one might select one or the other.

9.5.3 Confidence of point estimates

Consider the evolution of point estimates $\hat{y}(x)$ for a simple one-dimensional sinc function as a function of ϱ. For ϱ sufficiently small the solution is that

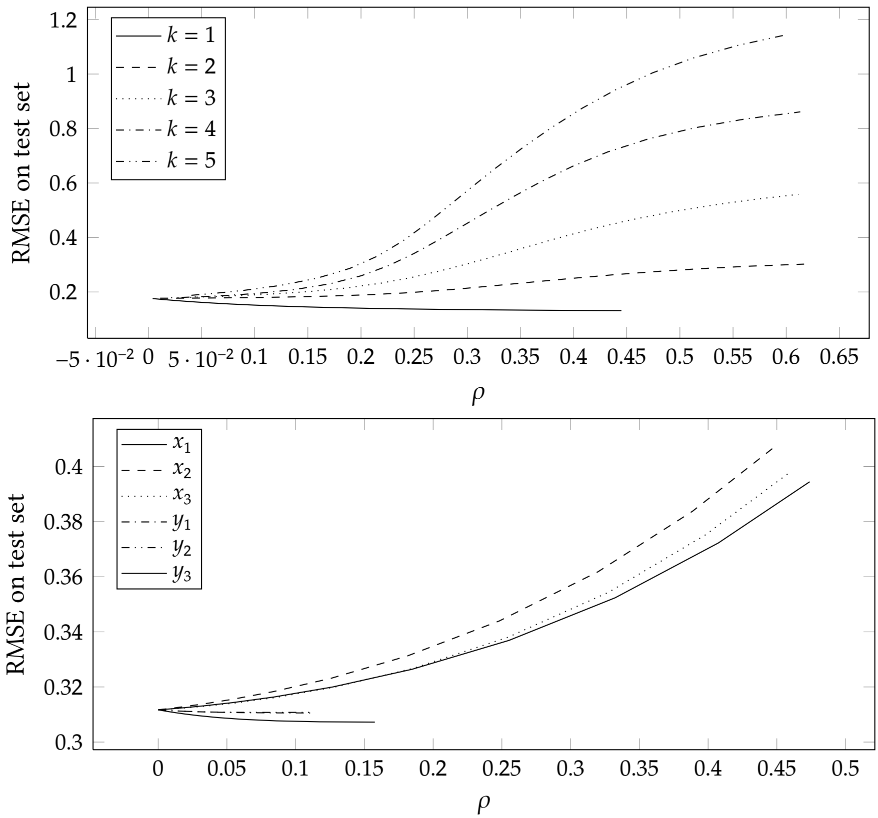

Figure 9.1: Sensitivity of the prediction performance of model (a) — top panel — and model (b) — bottom panel — in Section 9.5.1 with respect to their input variables.

of a LS-SVM estimator. For large ϱ the function will be driven to have zero derivatives at the training points i.e. the constant function. Figure 9.3 shows the sensitivity of point estimates for increasing levels of potential perturbations. If a point estimate stays close to the initial LS-SVM estimate for a wide range of ϱ, two conclusions can be drawn. Firstly the point possesses only little new information about the function and secondly with high confidence the predicted value is reliable. Both conclusions follow from the fact that even if the precise location of $\boldsymbol{x}_k$ is unknown, the estimate will not be affected. Figure 9.4 depicts the estimated function over the domain $[-4, 4]$ showing the interpolation behavior as a function of the perturbation parameter ϱ.

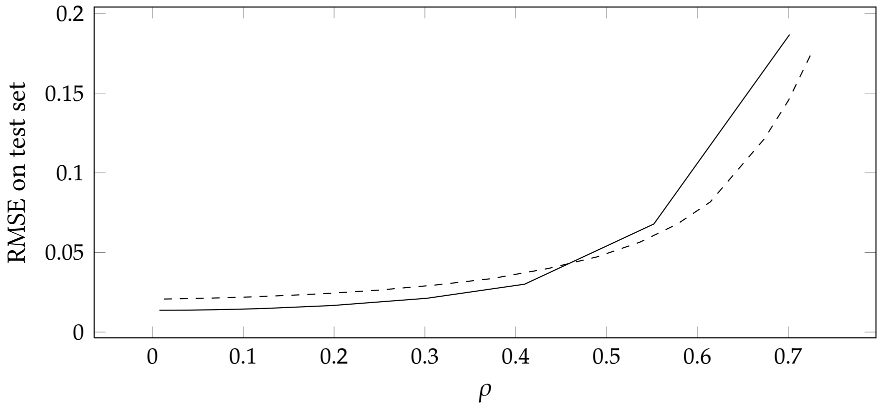

Figure 9.2: Sensitivity of the NARX model in Section 9.5.2 with respect to chosen kernel function. The dashed and the solid lines mark the polynomial kernel of order 4 ($c_2 = 4$) and the Gaussian RBF respectively.

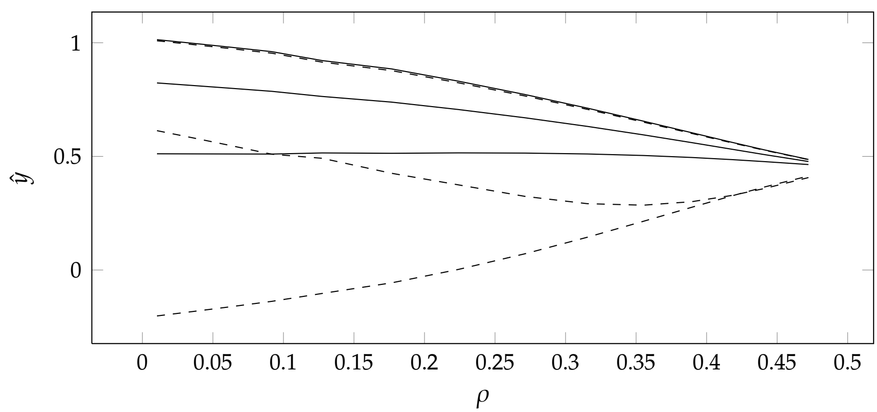

Figure 9.3: Examples for sensitivity of pointwise predictions $\hat{y}(x)$ of a simple one-dimensional sinc for $N = 80$ training data. The predictions for particular samples x_k are shown as a function of ϱ. The dashed lines represent points not taken from the training sample whereas the solid lines correspond to predictions for samples utilized during model estimate.

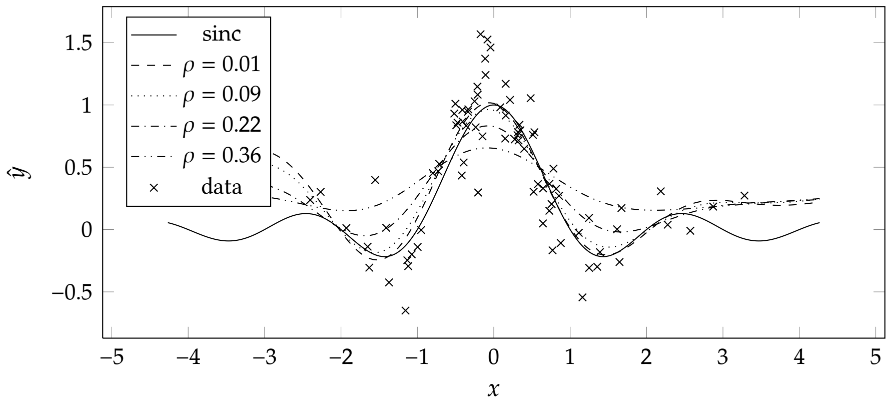

Figure 9.4: The estimate of a simple one-dimensional sinc function estimated using $N = 80$ training data. The shape of the function is shown for different values of ϱ.

9.5.4 Relation between regularization parameters

The parameter ϱ has a direct physical interpretation as the bound on the perturbations and therefore one would like to fix it. Yet the relation between the regularization parameters for the initial SOCP problem (9.14) and the LS problem (9.18) as given by (9.17) is only explicit once a solution has been computed. Therefore it is only possible to choose a pair ϱ', γ', solve (9.18) and then map these parameters onto γ and ϱ. An example is shown in Figure 9.5 (top panel).

While the parameter ϱ' varies on a logarithmic scale the parameters ϱ and γ are well represented on a linear scale. The variation of γ with respect to ϱ' has to be considered significant therefore it is necessary to search for a pair of γ', ϱ' that results in a constant γ and a desired ϱ. This nonlinear optimization problem in two variables is quite sensitive. Figure 9.5 (bottom panel) shows the result of such a conversion. Such a mapping is only possible for reasonable ranges of $10^{-4} \leq \gamma', \varrho' \leq 10^4$, otherwise the problem gets numerically unstable.

9.5.5 Approximation performance of $\boldsymbol{\Omega}_{xy}$

The matrix $\boldsymbol{\Omega}_{xy}$ is approximated using the Nyström approximation. Therefore a randomly chosen subset $\mathbb{S} \subset \mathbb{T}$ of size $s = |\mathbb{S}|$ is used. Table 9.1 reports the performances for different approximation dimensions $a = s \cdot d$. The approximation performance is measured as $\mathrm{PERF}(\boldsymbol{\Omega}_{xy}, s) = 100\, \|\boldsymbol{\Omega}_{xy} - \boldsymbol{\Omega}_{xy}^{(s)}\|_F / \|\boldsymbol{\Omega}_{xy}\|_F$.

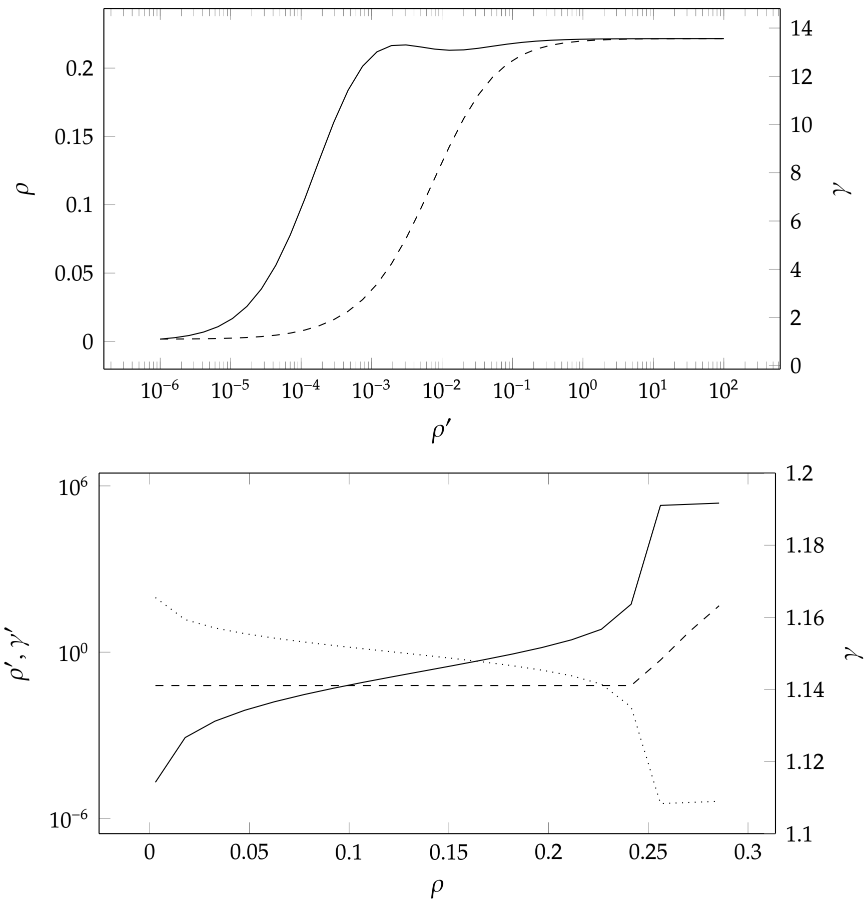

Figure 9.5: Relation of regularization parameters γ (dashed lines), ϱ (solid line), γ' (dotted line) and ϱ' (solid line). The upper panel shows the relations as a function of ϱ' with γ' fixed, whereas in the lower panel ϱ is varied for γ as constant as possible. The analyzed function is $y(\boldsymbol{x}) = \sum_{i=1}^{d} \operatorname{sinc}(x_i)$, $N = 500$.

Table 9.1: Performance of Nyström approximation $\boldsymbol{\Omega}_{xy}^{(s)}$ for $N = 1000$ and $d = 10$ and different subsample sizes s. The fit is defined as $100\|\boldsymbol{\Omega}_{xy} - \boldsymbol{\Omega}_{xy}^{(s)}\|_F / \|\boldsymbol{\Omega}_{xy}\|_F$. The values in parenthesis indicate standard deviations computed for different draws of the subsample.

SUBSAMPLE SIZE	FIT	RUNTIME
100	7.67 (0.67)	4.0 s
250	2.92 (0.30)	36.2 s
500	1.16 (0.12)	205.6 s
750	0.56 (0.15)	621.0 s

Each approximation is carried out for five different initial sets $\mathbb{T}$ and each one of them for three different subset selections.

From the numerical data it can be seen that the approximation performance is quite good even for low order approximations and gets better if larger subsamples are used. The Nyström approximation is based on an eigenvalue decomposition of a gram matrix for the subset $\mathbb{S}$. This matrix is of dimension $(s \cdot d)^2$. Therefore the approximation dimensions may not be too large as otherwise the computation time gets increasingly long.

9.5.6 Composite approximation performance

The general setting is identical to before but now approximation performance of $\boldsymbol{\Omega}_x^T(\boldsymbol{\Omega}_{xy} + (\gamma'\varrho')^{-1}\boldsymbol{I}_{Nd})^{-1}\boldsymbol{\Omega}_x$ is studied as a function of $\gamma'\varrho'$. The results are illustrated in Figure 9.6 for different subset sizes. As can be expected for small values of $\gamma'\varrho'$ the approximation dimension is secondary and even small scale approximations do well. Yet for very large values of $\gamma'\varrho'$ it becomes necessary to use very accurate approximations to obtain low approximation errors. In the transition phase the user is able to trade off accuracy versus admissible speed.

9.6 Conclusions

Based on the assumption of bounded unstructured perturbations on the inputs, a scheme for robustness analysis of nonlinear black box models has been derived. The estimation problem is convex and can be recast from a SOCP problem into a linear system. For this large linear system an approximate

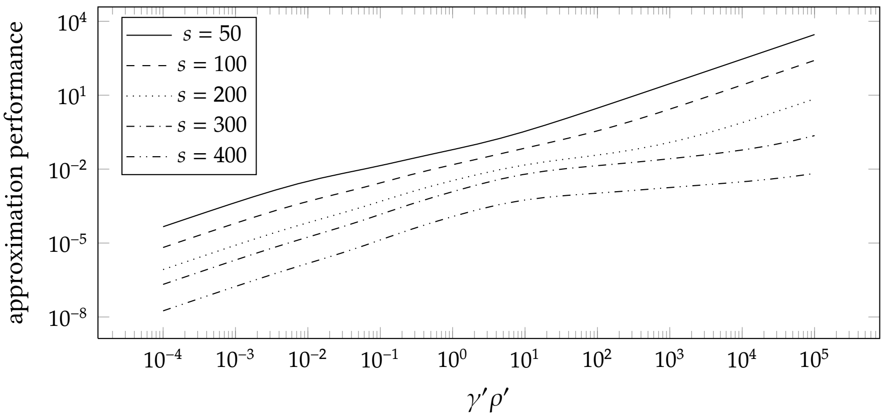

Figure 9.6: Composite approximation performance for $N = 500$ and $d = 5$ of $\boldsymbol{\Omega}_x^T(\boldsymbol{\Omega}_{xy} + (\gamma'\varrho')^{-1}\boldsymbol{I}_{Nd})^{-1}\boldsymbol{\Omega}_x$.

solution has been outlined. It has been shown on simple examples that existing models, based on the LS-SVM estimator, can be analyzed employing this methodology.

Segmentation of nonlinear time series 10

Based on the publication Falck, T., Ohlsson, H., Ljung, L., Suykens, J. A. K., and De Moor, B. (Aug. 2011). "Segmentation of time series from nonlinear dynamical systems". In: *Proceedings of the 18th IFAC World Congress.* (Milan, Italy, Aug. 28–11, 2011), pp. 13209–13214.

All systems studied in the previous chapters have in common that they are time invariant. In practice however it is possible that system dynamics change subject to auxiliary influence. Gradual changes can for example be effects of temperature or time dependent degradation. More abrupt changes in behavior can be caused by faults in sensors, actuators or the system itself. Depending on the application one has information on the origin of the time dependence and can directly use this for the identification of the model. This is for example considered in linear parameter varying (LPV) system identification. Another possibility is that there are a limited number of known behaviors that should be detected. In that case one can identify a time invariant model for each behavior and then use these to detect which regime is active [Bodenstein and Praetorius, 1977; Lindgren, 1978; Tugnait, 1982; Andersson, 1985]. The studied setting is the generalization of both. The main three assumptions are that i) there is no direct information about the times at which the system changes its behavior, ii) there is no a priori known model for the system and iii) the change is large enough to be reflected in the parameters of the to be identified model. In summary the objective is, given just measured data, to estimate times at which the model changes. As a side product one obtains approximate models that describe the system in between the change points. In contrast to all other chapters the objective of these approximate models is not the best predictive performance, but only a predictive performance that is good enough to detect changes in the system dynamics.

The goal to detect changes in the system dynamics without explicit knowledge of the system, as in e.g. a priori known models, is in general a combinatorial problem. Therefore any polynomial time algorithm aimed at solving this problem is a heuristic. The method proposed in this chapter is based on convex optimization and LS-SVM core models. The cornerstone for segmentation is given by a particular group ℓ_1-regularization while the support of nonlinear dynamics is contributed by LS-SVMs.

The key idea is to exploit the fact that the LS-SVM primal formulation is linear-in-parameters. This allows the introduction of new a parameter vector at each time t. In the next step the difference between two adjacent parameter vectors can then be penalized. This concept as heuristic for segmentation problems has initially been proposed by [Kim et al., 2009] for static linear regression to detect trends in economic data. It has recently been extended by Ohlsson et al. [2010] to also handle piecewise constant linear dynamical systems. An alternative name for the group ℓ_1-regularization technique used in this chapter as well as in the two references is sum-of-norms regularization. This regularization scheme can also be seen as an application of total-variation known from image processing [Rudin et al., 1992; Candès et al., 2006a] in high dimensional parameter instead of signal spaces. The main contribution of this chapter is the extension of the convex heuristic introduced in the two references to nonlinear systems in a kernel based framework.

Structure of the chapter The next section will state the general setting while Section 10.2 gives further information on the sum-of-norms regularization. A kernel based model is derived in Section 10.3 while model selection is briefly discussed in Section 10.4. A short overview on algorithmic considerations is given in Section 10.5. The chapter ends with an application to two motivational data sets in Section 10.7 and concludes in the last section.

10.1 Problem Formulation

The objective in this chapter is to segment time series generated by piecewise constant nonlinear dynamical systems. The only given information is measured data $\{(\boldsymbol{x}_t, y_t)\}_{t=1}^N$ where $\boldsymbol{x}_t$ will usually have ARX structure. In absence of

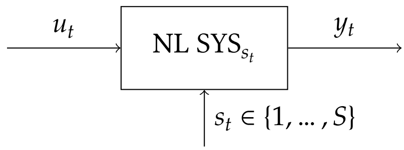

Figure 10.1: Nonlinear dynamical system with inputs u_t, outputs y_t and (unknown) scheduling variable s_t.

a priori known models they have to be estimated from the data. Therefore the starting point is once more the LS-SVM core model given by (4.1). For the problem considered here it will be of advantage that all model parameters are subject to regularization. Note that the offset parameter b is usually not regularized. As it will be easier to drop the offset b than to extend the regularization to it, the former will be assumed for the remainder of this chapter. To better motivate the approach, assume that the time instances $\{t_c\}_{c=1}^{C}$ at which the system changes its dynamics are given. Then using the LS-SVM core model in each segment one has

$$f_c(\boldsymbol{x}_t) = \boldsymbol{w}_c^T \boldsymbol{\varphi}_c(\boldsymbol{x}_t), \quad t = t_{c-1}, \dots, t_c - 1, \tag{10.1}$$

where $\boldsymbol{x}_t$ are ARX like regressors such that $\boldsymbol{x}_t = [y_{t-1}, \dots, y_{t-p}, u_t, \dots, u_{t-q}]$. This extends the linear ARX model

$$f_c(\boldsymbol{x}_t) = \boldsymbol{w}_c^T \boldsymbol{x}_t, \quad t = t_{c-1}, \dots, t_c - 1, \tag{10.2}$$

in [Ohlsson et al., 2010] by considering nonlinear basis functions $\boldsymbol{\varphi}_c$. The model parameters are $\boldsymbol{w}_c \in \mathbb{R}^{n_h}$ and the components of the nonlinear maps $\boldsymbol{\varphi}_c(\cdot) = [\varphi_c^1(\cdot), \dots, \varphi_c^{n_h}(\cdot)]^T : \mathbb{R}^D \to \mathbb{R}^{n_h}$ are the corresponding basis functions. Both are defined for $c = 1, \dots, C$ where $C < N$ and $t_0 = 1$ is assumed throughout this chapter without loss of generality. In this case a separate model can be estimated for each of the piecewise constant segments by solving

$$\begin{aligned} \min_{\boldsymbol{w}_c, e_t} \quad & \frac{1}{2}\boldsymbol{w}_c^T \boldsymbol{w}_c + \frac{1}{2}\gamma \sum_{t=t_{c-1}}^{t_c-1} e_t^2 \\ \text{subject to} \quad & y_t = \boldsymbol{w}_c^T \boldsymbol{\varphi}_c(\boldsymbol{x}_t) + e_t, \quad t = t_{c-1}, \dots, t_c - 1. \end{aligned} \tag{10.3}$$

Note that, as in most of the preceding chapters, the regularization parameter γ trades off the model fit, measured by the squared residuals, versus the model complexity, quantified using the quadratic penalty term $\boldsymbol{w}_c^T \boldsymbol{w}_c$. While the C estimation problems given by (10.3) above give rise to models for the system in each of the segments, their solution neither estimates the number of change points C nor their positions t_c. However, for the extension introduced in the following section that tackles these problems it is a necessary condition that the simpler problem given by (10.3) gives (reasonably) good results.

10.2 Piecewise Nonlinear Modeling

In this section an approach is introduced that is able to estimate the number of change points C as well as their positions t_c by means of a convex relaxation. The methodology followed here is based on [Ohlsson et al., 2010], which uses the same convex relaxation technique to segment linear ARX type models. In contrast to [Ohlsson et al., 2010] the objective here is to segment a nonlinear model of the form given by (10.1). First, to make the problem tractable, the basis functions are fixed across all segments, namely $\boldsymbol{\varphi}_c = \boldsymbol{\varphi} \forall c$. Therefore the basis functions have to be chosen rich enough to represent the dynamics of all segments. Then, in order to deal with the unknown change points t_c, $c = 1, \dots, C$, the parameter vectors $\boldsymbol{w}_c$ are overparametrized by introducing a new parameter vector $\boldsymbol{w}_t$ for each time instant t. Hence, one assumes a model of the form

$$f_t(\boldsymbol{x}_t) = \boldsymbol{w}_t^T \boldsymbol{\varphi}(\boldsymbol{x}_t), \quad t = 1, \dots, N. \tag{10.4}$$

For an exact representation of the modeling goal, the segmentation of the analyzed time series, one can define a vector $\boldsymbol{\nu}$ whose elements ν_t quantify the model change between two adjacent time instances, i.e. $\nu_t = \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|_2$ for $t = 2, \dots, N$. It is convenient to also define $\nu_1 = \|\boldsymbol{w}_1\|_2$. In principle any norm could be used to quantify the change in the model, however, as kernel based models are considered the ℓ_2-norm is the most natural choice and later on crucial to obtain a fully kernelized problem. With the vector of changes $\boldsymbol{\nu}$ one can augment the problem with an additional penalty $\|\boldsymbol{\nu}\|_0$ to minimize the cardinality of the change vector and therefore model changes. Depending on the size of the corresponding regularization parameter "many" of the regularized variables come out as zero. On the one hand this avoids a severe overfit and on the other hand this solves the overall goal of the proposed methodology, namely the segmentation of a time series based on a series of constant models.

As mentioned in Chapter 3 the cardinality function leads to combinatorial optimization problems. However, the ℓ_1-heuristic has been shown to be a powerful surrogate in many applications. Replacing the "ℓ_0-norm" by the

ℓ_1-norm one arrives at a sum-of-norms regularization scheme $\sum_{t=2}^{N} \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|_2$. Combining this with the overparametrized model (10.4) a convex relaxation for the segmentation problem is

$$\begin{aligned} \min_{\boldsymbol{w}_t, e_t} \quad & \lambda_m \|\boldsymbol{w}_1\|_2 + \lambda_s \sum_{t=2}^{N} \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|_2 + \frac{1}{2} \sum_{t=1}^{N} e_t^2 \\ \text{subject to} \quad & y_t = \boldsymbol{w}_t^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + e_t, \quad t = 1, \dots, N. \end{aligned} \tag{10.5}$$

In this problem the number of changes is roughly controlled by the regularization parameter λ_s, while λ_m defines the (initial) model complexity.

10.3 Nonparametric kernel based formulation

In the framework of LS-SVMs one can identify (10.5) as a combination of a LS-SVM core model with a special regularization scheme. This allows one to utilize the power of support vector machines for the modeling part of the problem. One key advantage is that the usually difficult choice of a good set of basis functions $\boldsymbol{\varphi}$ to model all different segments is simplified by this kernel based method.

10.3.1 Dual formulation

Due to the ℓ_2-norms (which are not squared) in (10.5), the overparametrized problem has to be solved as a second order cone programming problem (SOCP) instead of a simple linear system as for (10.3). Therefore the results obtained in Section 9.1 can be exploited which yield the following dual problem.

Lemma 10.1. *Let G be a matrix square root of the kernel matrix $\boldsymbol{\Omega} \in \mathbb{R}^{N\times N}$ with $\Omega_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Furthermore for $\boldsymbol{\alpha} \in \mathbb{R}^N$ define $\boldsymbol{\alpha}_{[t]} \in \mathbb{R}^N$ such that $(\boldsymbol{\alpha}_{[t]})_i = \alpha_i$ for $i \geq t$ and zero otherwise. Then the dual problem of* (10.5) *is*

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^T \boldsymbol{y} - \frac{1}{2} \boldsymbol{\alpha}^T \boldsymbol{\alpha} \\ \text{subject to} \quad & \|\boldsymbol{G}\boldsymbol{\alpha}_{[t]}\|_2 \leq \lambda_t, \quad t = 1, \dots, N, \end{aligned} \tag{10.6}$$

where $\boldsymbol{y} = [y_1, \dots, y_N]^T$, $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_N]^T$ are the Lagrange multipliers corresponding to the equality constraints in (10.5), *$\lambda_1 = \lambda_m$ and $\lambda_t = \lambda_s$ for $t = 2, \dots, N$.*

Proof. Let $\mathcal{K} = \{(\boldsymbol{x}, s) \mid \|\boldsymbol{x}\|_2 \leq s\}$ be the second order cone [Boyd and Vandenberghe, 2004] then the objective function (10.5) can be reformulated as $\sum_{t=1}^{N} \lambda_t \omega_t + \frac{1}{2} \sum_{t=1}^{N} e_t^2$ with additional constraints $(\boldsymbol{w}_1, \omega_1) \in \mathcal{K}$ and $(\boldsymbol{w}_t - \boldsymbol{w}_{t-1}, \omega_t) \in \mathcal{K}$ for $t = 2, \ldots, N$. Using conic duality and noting that the second order cone is self-dual, the corresponding Lagrangian is

$$\begin{aligned}\mathcal{L}(\boldsymbol{w}_t, \boldsymbol{v}_t, e_t, \alpha_t, \omega_t, \tau_t) = &\sum_{t=1}^{N} \lambda_t \omega_t + \frac{1}{2} \sum_{t=1}^{N} e_t^2 - \sum_{t=1}^{N} \omega_t \tau_t \\ &- \boldsymbol{v}_1^T \boldsymbol{w}_1 - \sum_{t=2}^{N} \boldsymbol{v}_t^T (\boldsymbol{w}_t - \boldsymbol{w}_{t-1}) + \sum_{t=1}^{N} \alpha_t (y_t - \boldsymbol{w}_t^T \boldsymbol{\varphi}(\boldsymbol{x}_t) - e_t) \end{aligned} \quad (10.7)$$

with $(\boldsymbol{v}_t, \tau_t) \in \mathcal{K}$ for $t = 1, \ldots, N$. The corresponding KKT conditions for optimality [see e.g. Boyd and Vandenberghe, 2004] are

$$\begin{aligned} &\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}_t} = \mathbf{0} : \ \boldsymbol{v}_t - \boldsymbol{v}_{t+1} = -\alpha_k \boldsymbol{\varphi}(\boldsymbol{x}_t), \ \ t = 1, \ldots, N-1, \\ &\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}_N} = \mathbf{0} : \ \boldsymbol{v}_N = -\alpha_N \boldsymbol{\varphi}(\boldsymbol{x}_N), \\ &\frac{\partial \mathcal{L}}{\partial e_t} = 0 : \ e_t = \alpha_t, \ \ t = 1, \ldots, N, \\ &\frac{\partial \mathcal{L}}{\partial \omega_t} = 0 : \ \tau_t = \lambda_t, \ \ t = 1, \ldots, N. \end{aligned}$$

Note that applying the first KKT condition recursively to the second one yields $\boldsymbol{v}_t = -\sum_{k=t}^{N} \alpha_k \boldsymbol{\varphi}(\boldsymbol{x}_k) = -\boldsymbol{\Phi}\boldsymbol{\alpha}_{[t]}$ where $\boldsymbol{\Phi} = [\boldsymbol{\varphi}(\boldsymbol{x}_1), \ldots, \boldsymbol{\varphi}(\boldsymbol{x}_N)]$. Then substitution of the KKT conditions into the Lagrangian yields the dual optimization problem

$$\begin{aligned} &\max_{\boldsymbol{v}_t, \boldsymbol{\alpha}} \quad \boldsymbol{\alpha}^T \boldsymbol{y} - \frac{1}{2} \boldsymbol{\alpha}^T \boldsymbol{\alpha} \\ &\text{subject to} \quad \|\boldsymbol{\Phi}\boldsymbol{\alpha}_{[t]}\|_2 \leq \lambda_t, \quad t = 1, \ldots, N. \end{aligned} \quad (10.8)$$

Depending on the feature map this problem may still be infinite dimensional. To obtain a finite dimensional problem, note that the constraints of (10.8) in squared form are $\boldsymbol{\alpha}_{[t]}^T \boldsymbol{\Omega} \boldsymbol{\alpha}_{[t]} \leq \lambda_t^2$ as $\boldsymbol{\Phi}^T \boldsymbol{\Phi} = \boldsymbol{\Omega}$. This allows writing the possibly infinite dimensional problem (10.8) in terms of the finite number of Lagrange multipliers $\boldsymbol{\alpha}$ and a matrix square root $\boldsymbol{G}$ of the kernel matrix $\boldsymbol{\Omega}$ as in (10.6). □

10.3.2 Recovery of sparsity pattern and predictive model

Instead of a problem in $N \cdot n_h$ variables $\boldsymbol{w}_t$ as in (10.5) the problem has been reduced to just N variables in (10.6). Yet to use the solution for prediction, one also needs to rewrite the model (10.1) in terms of the dual variables. As the primal problem is sparse one would also like to recover its sparsity pattern.

Lemma 10.2. *Let $\mathcal{A} = \{t \mid \|\boldsymbol{G}\boldsymbol{\alpha}_{[t]}\|_2 = \lambda_t, t = 1, \dots, N\}$ denote the ordered set of active constraints and $\mathcal{A}_c$ its complement. Denote the k-th element of $\mathcal{A}$ by t_k where k runs from 1 to $|\mathcal{A}|$, the cardinality of $\mathcal{A}$. For ease of notation also define $t_{|\mathcal{A}|+1} = N + 1$. Finally define the function*

$$\operatorname{idx}(t) = \arg\min_k \quad k - 1 \quad \text{subject to} \quad t < t_k$$

and denote the value of $\|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|_2$ by $\rho_t \lambda_t$.

Then the sparsity pattern is given by

1. *$\rho_t = 0$ for $t \in \mathcal{A}_c$ and*

2. *the solution of the linear system in ρ_t for $t \in \mathcal{A}$:*

$$y_t - \alpha_t = \boldsymbol{k}(\boldsymbol{x}_t)^T \sum_{k=1}^{\operatorname{idx}(t)} \boldsymbol{\alpha}_{[t_k]} \rho_{t_k}, \quad t = 1, \dots, N, \tag{10.9a}$$

$$\boldsymbol{\alpha}^T \boldsymbol{y} - \boldsymbol{\alpha}^T \boldsymbol{\alpha} = \sum_{k=1}^{|\mathcal{A}|} \lambda_{t_k}^2 \rho_{t_k}, \tag{10.9b}$$

where $\boldsymbol{k}(\boldsymbol{z}) = [K(\boldsymbol{x}_1, \boldsymbol{z}), \dots, K(\boldsymbol{x}_N, \boldsymbol{z})]^T$.

Proof. The fact that $\rho_t = 0$ for $t \in \mathcal{A}_c$ is a direct result from complimentary slackness. To obtain the linear system let $\boldsymbol{\delta}_t = \boldsymbol{w}_{t-1} - \boldsymbol{w}_t$ for $t = 2, \dots, N$ and $\boldsymbol{\delta}_1 = \boldsymbol{w}_1$. Then by applying the definition of complementary slackness one has $\boldsymbol{v}_t^T \boldsymbol{\delta}_t + \tau_t \omega_t = 0$. After substitution of the optimal values for τ_t and $\boldsymbol{v}_t$ into this relation, it is straightforward to check that it is only satisfied by $\boldsymbol{\delta}_t = -\rho_t \boldsymbol{v}_t$ for $(\boldsymbol{\delta}_t, \omega_t) \in \mathcal{K}$.

To recover $\boldsymbol{w}_t$ one has $\boldsymbol{w}_t = \sum_{k=1}^{N} \boldsymbol{\delta}_t$. Due to the sparsity pattern it is sufficient to sum over the nonzero differences which correspond to the time instances in $\mathcal{A}$. Substituting the optimal parameters into the constraint of (10.5) then yields (10.9a). Equation (10.9b) follows from plugging the optimal parameters into the objective functions of primal (10.5) and dual (10.6) and exploiting that the duality gap is zero. □

Finally one can state a predictive equation in terms of the dual variables.

Corollary 10.3. *The prediction at a new point z_t at time $t \in \{1, \ldots, N\}$ is obtained as*

$$f_t(z_t) = \boldsymbol{k}(z_t)^T \sum_{k=1}^{\mathrm{idx}(t)} \rho_{t_k} \boldsymbol{\alpha}_{[t_k]}. \tag{10.10}$$

Proof. First substitute the expressions for $\boldsymbol{w}_t$ obtained in the proof of Lemma 10.2 into the model equation (10.1). Then the dual model (10.10) follows from replacing the inner products of the feature map by the kernel function. □

Remark 10.1. Note that the predictions obtained from (10.10) depend on the time instant t at which a new point z_t is acquired. This requirement could be relaxed if the operation region c that generated the new point would be known. In general this will not be the case, therefore the primary use of this model is in validation schemes for model selection. This is in contrast to all other models in this thesis whose predictions are independent of time or system state.

In the following the steps needed to obtain a predictive model in the dual are summarized.

Algorithm 10.1 (Model estimation)**.**

1. Choose regularization constants λ_m and λ_s.
2. Compute the kernel matrix $\Omega_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and its matrix square root $\boldsymbol{G}$ such that $\boldsymbol{\Omega} = \boldsymbol{G}^T \boldsymbol{G}$.
3. Solve the dual estimation problem (10.6) for the optimal Lagrange multipliers $\boldsymbol{\alpha}$.
4. Solve (10.9) for ρ_t with $t \in \mathcal{A}$ to recover the sparsity pattern of the primal problem.
5. Evaluate (10.10) to obtain predictions.

10.4 Model selection

To obtain a good segmentation the model complexity as well as its sparsity need to be tuned. Therefore suitable values for the regularization constants λ_m and λ_s have to be selected. Additionally in the primal formulation (10.5) the basis functions need to be fixed or in the dual (10.6) a kernel function needs to be chosen. In the nonlinear setting described here the regularization parameter λ_m is crucial to control the complexity of the initial nonlinear model whereas λ_s controls the sparsity such that the change points $\{t_c\}_{c=1}^C$ can be discovered.

The main objective within this chapter is to identify the correct change points. Estimating a single model on a known segment using (10.3) will always be able to outperform the joint model obtained from (10.5). A model specific to a single segment has better control over the complexity versus fit trade-off and is also able to select a better suited set of basis functions (or kernel in the dual). Therefore, if predictive performance is important, it is suggested to re-estimate models on the individual segments using (10.3) once the change points are identified.

The modeling power of the global description (10.5) needs to be powerful enough to capture the nonlinear dynamics to a large extent to be able to detect changes. Therefore the selection λ_m and λ_s is performed according to generalization performance of the models using a validation scheme [see e.g. Hastie et al., 2009]. Once model parameters have been estimated they can be visualized as $\|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|_2$ over t (see Figure 10.2). In case sparsity is not perfect one can either perform thresholding, clustering [Alzate, 2009] or a combination thereof.

To help with the parameter selection process consider the following lemma and its corollary.

Lemma 10.4. *Let $\boldsymbol{\alpha}$ correspond to the solution without any changes. It can be obtained from* (10.5) *without the sparsity inducing term $\sum_{t=2}^N \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|_2$ and the corresponding variables $\boldsymbol{w}_t$ for $t = 2, \dots, N$. Alternatively* (10.6) *can be solved with a single constraint for $t = 1$. Then for $\lambda_s \geq \lambda_{s,\max}$ where*

$$\lambda_{s,\max} = \max_{t=2,\dots,N} \sqrt{\boldsymbol{\alpha}_{[t]}^T \boldsymbol{\Omega} \boldsymbol{\alpha}_{[t]}}, \tag{10.11}$$

the solution of (10.5) *is also constant, i.e. $\boldsymbol{w}_1 = \cdots = \boldsymbol{w}_N$.*

Proof. For a nonsmooth problem a necessary condition for optimality is that $\mathbf{0}$ is in the subgradient at the solution. Eliminating e_t from (10.5) one obtains $\mathcal{J}(\boldsymbol{\delta}_1, \dots, \boldsymbol{\delta}_N) = \sum_{t=1}^N \lambda_t \|\boldsymbol{\delta}_t\|_2 + \frac{1}{2} \sum_{t=1}^N (y_t - \sum_{k=1}^t \boldsymbol{\delta}_k^T \boldsymbol{\varphi}(\boldsymbol{x}_t))^2$. The subgradient of $\mathcal{J}$ at $\boldsymbol{\delta}_t$ is $\lambda_t \partial_{\boldsymbol{\delta}_t} \|\boldsymbol{\delta}_t\|_2 - \sum_{k=t}^N \boldsymbol{\varphi}(\boldsymbol{x}_k)(y_k - \sum_{l=1}^k \boldsymbol{\delta}_l^T \boldsymbol{\varphi}(\boldsymbol{x}_k))$. Note that the subgradient of the ℓ_2-norm at zero is the unit norm ball. Therefore one needs to satisfy that $\partial_{\boldsymbol{\delta}_t} \mathcal{J}|_{\boldsymbol{\delta}_2 = \dots \boldsymbol{\delta}_N = \mathbf{0}} = \lambda_s \boldsymbol{\mu}_t - \sum_{k=t}^N \boldsymbol{\varphi}(\boldsymbol{x}_k)(y_k - \boldsymbol{w}_1^T \boldsymbol{\varphi}(\boldsymbol{x}_k)) \in \mathbf{0}$ for $t = 2, \dots, N$ with $\|\boldsymbol{\mu}_t\|_2 \leq 1$. Note that $y_k - \boldsymbol{w}_1^T \boldsymbol{\varphi}(\boldsymbol{x}_k) = e_k = \alpha_k$. Then the last expression can be rewritten in matrix notation as $\|\boldsymbol{\Phi}\boldsymbol{\alpha}_{[t]}\|_2 \leq \lambda_s$. Squaring this relation one obtains the quadratic form $\boldsymbol{\alpha}_{[t]}^T \boldsymbol{\Omega} \boldsymbol{\alpha}_{[t]}$. As the relation has to hold for all $t = 2, \dots, N$ the condition stated in the Lemma follows. □

Corollary 10.5. *Let* $\lambda_s \geq \lambda_{s,\max}$ *and additionally* $\lambda_m \geq \lambda_{m,\max} = \sqrt{\boldsymbol{y}^T \boldsymbol{\Omega} \boldsymbol{y}}$ *then* $\boldsymbol{w}_1 = \boldsymbol{0}$.

Proof. The proof is analogous to the one of Lemma 10.4. □

10.5 Algorithm

The kernel based dual problem (10.8) can be solved using general purpose Second Order Cone Programming (SOCP) solvers like `Sedumi` [Sturm, 1999]. The same holds for the primal problem (10.5), if it is finite dimensional and an explicit expression for the feature map $\boldsymbol{\varphi}$ is given. This is especially straightforward with modeling tools like CVX [Grant and Boyd, 2011]. Yet, the large number of constraints of any of the two problems makes their solution slow or even infeasible. In the following several ways to accelerate the solution are discussed. This also makes the procedure applicable to larger problem sizes.

10.5.1 Active set strategy

The dual problem (10.6) suggests that depending on the value of the regularization constant λ_s, many of its constraints will not be active, i.e. $\|\boldsymbol{G}\boldsymbol{\alpha}_{[t]}\|_2 < \lambda_s$. Therefore omitting these constraints does not change the solution. This motivates the use of an active set strategy. Starting with a single constraint, the most violating constraint is successively added to the set of active constraints as formalized in the following procedure.

Algorithm 10.2 (Active set strategy)**.**

1. Initialize $\mathcal{A} = \{1\}$ and $k = 1$.
2. Solve

$$\min_{\boldsymbol{\alpha}} \; \boldsymbol{y}^T \boldsymbol{\alpha} - \frac{1}{2} \boldsymbol{\alpha}^T \boldsymbol{\alpha} \text{ subject to } \|\boldsymbol{G}\boldsymbol{\alpha}_{[t]}\| \leq \lambda_t, \; t \in \mathcal{A}. \tag{10.12}$$

3. Compute $t_k = \arg\max_{2 \leq t \leq N} \{\|\boldsymbol{G}\boldsymbol{\alpha}_{[t]}\|_2\}$.
4. If $\|\boldsymbol{G}\boldsymbol{\alpha}_{[t_k]}\| \leq \lambda_s$ then terminate.
5. Else $\mathcal{A} := \mathcal{A} \cup \{t_k\}$, $k := k + 1$ and goto (2).

In the experiments (for an example see Figure 10.7) it can observed that only a small fraction of the whole number of constraints is needed to define the final solution. A similar approach has been presented in [Jenatton et al., 2009; Bach et al., 2011].

10.5.2 First order algorithms

After an initial solution for (10.12) has been obtained, the optimal solution will likely change only gradually over the iterations needed to satisfy all constraints. Therefore a good initial guess for the new solution is given by the solution of the last iteration. Interior-point solvers like `Sedumi` are in general hard to warm start such that there is no benefit of having a good initial guess. In contrast first order schemes are very easy to warm start.

After a change of variables ($\boldsymbol{w}_t - \boldsymbol{w}_{t-1} = \boldsymbol{\delta}_t$) the primal problem (10.5) can be solved with efficient software like SPGL1 [Van Den Berg and Friedlander, 2008], NESTA [Becker et al., 2009] or SLEP [Liu et al., 2009]). However this is not true for the dual problem (10.12) as most first order schemes are based on the idea of projected gradients. The main requirement of these algorithms is that the projection onto the constraint set is cheap. In the current form of (10.12) this is not the case. There are at least three ways to treat this problem: (i) use a framework like TFOCS [Becker et al., 2011] which is able to compute approximate solutions for (10.12) using smoothing, (ii) augment the Lagrangian as described in [Falck et al., 2011] and (iii) solving the dual of (10.12) with the afore mentioned software. Here the last possibility is described.

Lemma 10.6. *The dual of* (10.12) *is*

$$\min_{\boldsymbol{s}_t \in \mathbb{R}^N} \quad \sum_{t \in \mathcal{A}} \lambda_t \|\boldsymbol{s}_t\|_2 + \frac{1}{2} \left\| \boldsymbol{y} - \sum_{t \in \mathcal{A}} (\boldsymbol{G}^T \boldsymbol{s}_t)_{[t]} \right\|_2^2 , \tag{10.13}$$

where the i-th element of $(\boldsymbol{G}^T \boldsymbol{s}_t)_{[t]}$ *is zero if* $i < t$ *and unchanged otherwise.*

The original variables can be recovered with $\boldsymbol{\alpha} = \boldsymbol{y} - \sum_{t \in \mathcal{A}} (\boldsymbol{G}^T \boldsymbol{s}_t)_{[t]}$.

Proof. In conic form the constraints of (10.12) can be written as $(\boldsymbol{G}\boldsymbol{\alpha}_{[t]}, \lambda_t) \in \mathcal{K}$. This gives rise to the Lagrangian $\mathscr{L}(\boldsymbol{\alpha}, \boldsymbol{s}_t, \zeta_t) = \boldsymbol{\alpha}^T \boldsymbol{y} - \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{\alpha} - \sum_{t \in \mathcal{A}} \boldsymbol{s}_t^T \boldsymbol{G}\boldsymbol{\alpha}_{[t]} - \sum_{t \in \mathcal{A}} \lambda_t \zeta_t$, with $(\boldsymbol{s}_t, \zeta_t) \in \mathcal{K}$ for $t \in \mathcal{A}$. Taking the condition for optimality for $\boldsymbol{\alpha}$ one obtains $\boldsymbol{\alpha} = \boldsymbol{y} - \sum_{t \in \mathcal{A}} (\boldsymbol{G}^T \boldsymbol{s}_t)_{[t]}$. Note that the conic constraints $\|\boldsymbol{s}_t\|_2 \leq \zeta_t$ are always active and therefore can be moved into the objective.

Now substitution of $\boldsymbol{\alpha}$ into the Lagrangian yields the dual problem given in (10.13). □

10.6 Extension to different loss functions

The modification of (10.5) to different convex loss functions such as ε-insensitive loss function is straightforward. For the ε-insensitive loss function $L_\varepsilon(x) = |x| - \varepsilon$ for $|x| \geq \varepsilon$ and zero otherwise as used in Support Vector Machines [Vapnik, 1998; Schölkopf and Smola, 2002], the modified primal problem is given by

$$\begin{aligned}
\min_{\boldsymbol{w}_t, \xi_t^\pm} \quad & \lambda_m \|\boldsymbol{w}_1\|_2 + \lambda_s \sum_{t=2}^{N} \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|_2 + \sum_{t=1}^{N} (\xi_t^+ - \xi_t^-) \\
\text{subject to} \quad & y_t - \boldsymbol{w}_t^T \boldsymbol{\varphi}(\boldsymbol{x}_t) \leq \varepsilon + \xi_t^+, \quad t = 1, \dots, N \\
& \boldsymbol{w}_t^T \boldsymbol{\varphi}(\boldsymbol{x}_t) - y_t \leq \varepsilon + \xi_t^-, \quad t = 1, \dots, N \\
& \xi_t^\pm \geq 0, \quad t = 1, \dots, N.
\end{aligned}$$

Lemma 10.7. *The solution of the primal problem with ε-insensitive loss can be obtained from the kernel based dual*

$$\begin{aligned}
\max_{\alpha_t} \quad & \sum_{t=1}^{N} \alpha_t y_t - \varepsilon \sum_{t=1}^{N} |\alpha_t| \\
\text{subject to} \quad & \|\boldsymbol{G}\boldsymbol{\alpha}_{[t]}\|_2 \leq \lambda_t, \quad t = 1, \dots, N \\
& -1 \leq \alpha_t \leq 1, \quad t = 1, \dots, N.
\end{aligned}$$

Proof. The proof of the ε-insensitive loss is a straightforward extension of the one of Lemma 10.1 for the ℓ_2-loss. The corresponding Lagrangian is

$$\begin{aligned}
\mathscr{L}(\boldsymbol{w}_t, \boldsymbol{v}_t, \xi_t^\pm, \alpha_t^\pm, \eta_t^\pm, \omega_t, \tau_t) = \sum_{t=1}^{N} \lambda_t \omega_t + \sum_{t=1}^{N} (\xi_t^+ + \xi_t^-) \\
- \sum_{t=1}^{N} \omega_t \tau_t - \boldsymbol{v}_1^T \boldsymbol{w}_1 - \sum_{t=2}^{N} \boldsymbol{v}_t^T (\boldsymbol{w}_t - \boldsymbol{w}_{t-1}) - \sum_{t=1}^{N} \xi_t^+ \eta_t^+ - \sum_{t=1}^{N} \xi_t^- \eta_t^- \\
+ \sum_{t=1}^{N} \alpha_t^+ (y_t - \boldsymbol{w}_t^T \boldsymbol{\varphi}(\boldsymbol{x}_t) - \varepsilon - \xi_t^+) + \sum_{t=1}^{N} \alpha_t^- (\boldsymbol{w}_t^T \boldsymbol{\varphi}(\boldsymbol{x}_t) - y_t - \varepsilon - \xi_t^-),
\end{aligned}$$

with $\alpha_t^\pm, \eta_t^\pm \geq 0$, $(\boldsymbol{w}_t, \omega_t) \in \mathcal{K}$ and $(\boldsymbol{v}_t, \tau_t) \in \mathcal{K}$ for $t = 1, \dots, N$. Let $\alpha_t = \alpha_t^+ - \alpha_t^-$ then KKT conditions for $\boldsymbol{w}_t$ are the same as in Lemma 10.1 and the kernelization can be performed as before. The KKT conditions for the residuals $\xi_t^{\pm}$ are $0 = \partial\mathscr{L}/\partial\xi_t^{\pm} = 1 - \alpha_t^{\pm} - \eta_t^{\pm}$. Exploiting the positivity constraints for the Lagrange multipliers $\alpha_t^{\pm}$ and $\eta_t^{\pm}$ this can be simplified to $0 \leq \alpha_t^{\pm} \leq 1$. Finally substitution of the KKT conditions into the Lagrangian then yields the objective function of the dual optimization problem. □

10.7 Experiments

10.7.1 NFIR Hammerstein system

As first example a simple Hammerstein type system with $y_t = [b_{1,t}\, b_{2,t}]\, \mathrm{sinc}(\boldsymbol{x}_t) + e_t$, $\boldsymbol{x}_t = [u_t, u_{t-1}]^T$ with $\mathrm{sinc}(\cdot)$ applied element-wise is considered. The input signal u_t and the noise e_t are white and Gaussian. The noise is scaled such that the data has a signal to noise ratio of 10 dB, while the input signal has unit variance. The parameters are chosen as

$$(b_{1,t}, b_{2,t}) = \begin{cases} (5, -2), & 100 < t \leq 200, \\ (1, 2), & \text{otherwise.} \end{cases}$$

In total 900 equally spaced samples in the time interval $1 \leq t \leq 300$ are generated. These are split into three parts by taking every third sample, one for estimation, one for model selection and one for the final evaluation of the model performance. The model is used in its dual formulation (10.6) and a RBF kernel is applied with the bandwidth σ fixed to 1. The regularization parameter γ is selected according to prediction performance on the validation set. The obtained sparsity pattern is shown in Figure 10.2. Note that the procedure correctly isolated the two change points and the initial model. However, especially close to the change points, there are some small spurious components. In Figure 10.3 the pairwise norms $\|\boldsymbol{w}_k - \boldsymbol{w}_l\|_2$ for all differences $\|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|_2$ that are at least as big as 10^{-3} times the largest one are shown. One can clearly see only three segments are really significant and that the first and the third segment share the same dynamics.

The predictions on the independent test data as well as the residuals are shown in Figure 10.4. The root mean squared error on the whole test data is 0.1905 for the piecewise nonlinear model. As a reference a LS-SVM trained on the whole data (without knowledge of the segments) achieves a RMSE of 0.6638 on the test set.

To illustrate the advantage of using a nonlinear model, the results are compared with a segmented ARX model. Using the same scheme as proposed

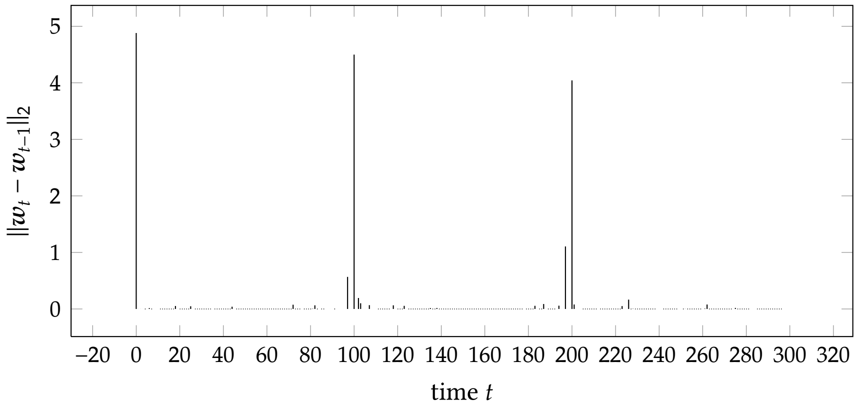

Figure 10.2: Sparsity pattern obtained from Lemma 10.2 for the nonlinear system described in Section 10.7.1 switching to a different dynamic at $t_1 = 100$ and switching back to the initial dynamics at $t_2 = 200$.

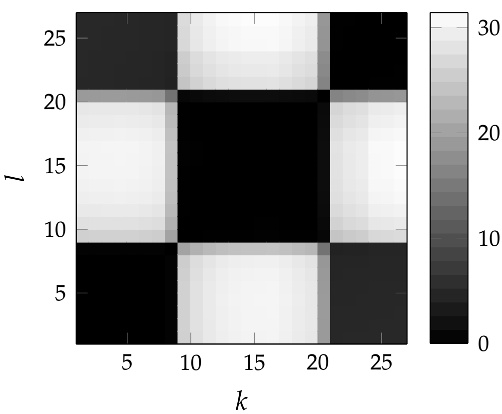

Figure 10.3: Norm $\|w_k - w_l\|_2$ for $k, l \in \{t : \|w_t - w_{t-1}\|_2 \geq 10^{-3} \max_t \|w_t - w_{t-1}\|_2\}$. The corresponding nonlinear system is specified in Section 10.7.1.

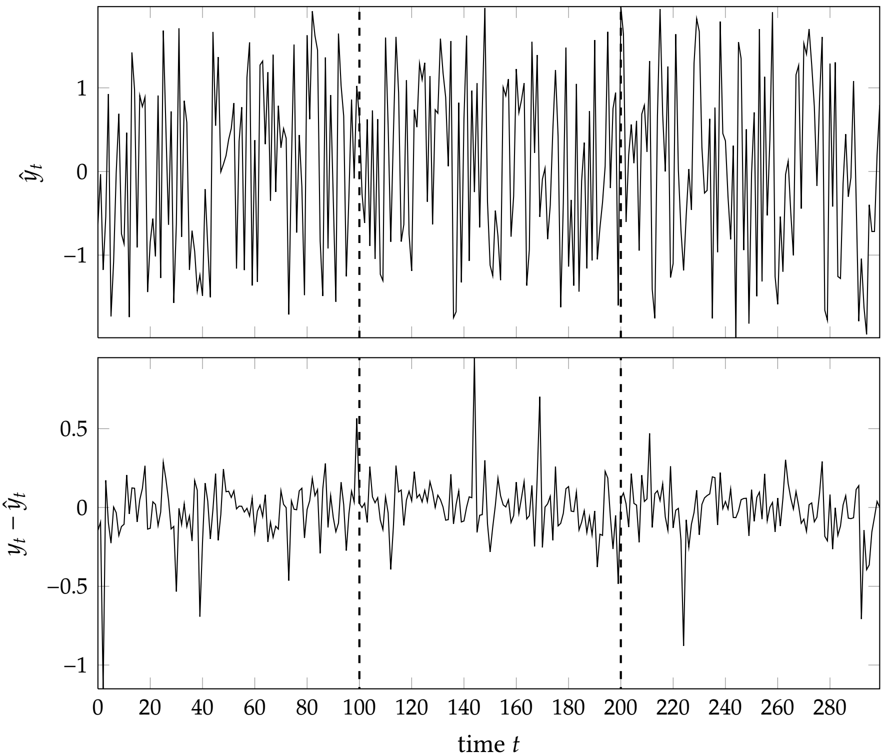

Figure 10.4: Predictions (top panel) of the model described in Section 10.7.1 on independent test data and the corresponding modeling errors (bottom panel). The vertical dashed lines indicate the positions of the change points.

in [Ohlsson et al., 2010], a model (10.2) with $x_t = [u_t \, u_{t-1}]^T$, is estimated. If the prediction performance on the validation set is used to find the number of segments, no change points are found. The estimated ARX parameters are therefore equivalent to those of a least squares estimate on the whole estimation data set. The prediction performance on the test data yields a root mean squared error of 1.060.

To obtain an indication of the best case performance, a LS-SVM (10.3) model given the true segmentation is trained. For optimal performance full model selection is carried out, i.e. the bandwidth σ as well as the regularization parameter γ are selected based on prediction performance. The implementation trains one model for the second segment and another model with the

Table 10.1: Root mean squared error (RMSE) on independent test data for different sections (I: $1 \leq t < 100$, II: $100 \leq t < 200$, III: $200 \leq t < 300$) of Example 1. Piecewise nonlinear model Algorithm 10.1 (PW NL), piecewise ARX model (PW ARX), LS-SVM given the true change points (10.3).

	PW NL (Algorithm 10.1) RMSE	$\sigma = 1, \gamma$	PW ARX RMSE	LS-SVM (Eq. 10.3) RMSE	(σ, γ)
I	0.206	0.468	1.045	0.163	(1.274, 33.6)
II	0.185	0.468	1.003	0.140	(1.274, 297.6)
III	0.180	0.468	1.118	0.126	(1.274, 33.6)

combined data of the remaining two. The results of the piecewise nonlinear model, the piecewise ARX model and the segment-wise LS-SVM are reported in Table 10.1.

10.7.2 NARX Wiener system

As second example a Wiener type system with ARX structure is considered. It is defined by $y_t = \sin(\frac{\pi}{2}\, \boldsymbol{\theta}_t^T \boldsymbol{x}_t) + e_t$ and $\boldsymbol{x}_t = [y_{t-1}, y_{t-2}, u_t, u_{t-1}, u_{t-2}]^T$. The input signal u_t and the noise term e_t are zero mean white Gaussian. The noise has variance 0.1^2 and the input is scaled such that it is in the interval $[-1, 1]$. The parameter vector $\boldsymbol{\theta}_t$ is scaled to unit mean and chosen as $\boldsymbol{\theta}_t|_{t=1}^{400} = [-0.525, 0.096, 01.585, -0.562, 0.542, -0.135]^T$ and $\boldsymbol{\theta}_t|_{t=401}^{600} = [-1.168, -1.401, 2.178, 1.334, 0.247, -0.190]^T$. Again the data is split into three parts by taking every third sample. Therefore the estimation data at position 134 ($t = 400$) is governed by a different system than the corresponding sample ($t = 401$) in the validation data. The kernel function is again a RBF kernel with fixed bandwidth $\sigma = 1$ and the regularization parameter γ is selected based on validation performance. The resulting sparsity pattern is depicted in Figure 10.5. The initial model at position 1 is clearly visible. Around position 134 one can observe two significant peaks. The energy of this change is spread over two positions as there is a mismatch of dynamics in the estimation and the validation data sets at position 134. This can also be seen from the pairwise differences in Figure 10.6. One can clearly see two blocks that share the same dynamics, but the model at one position correlates well with the models before and after it. The root mean squared errors on an independent test set are for segment I: 0.223 (0.190), segment II: 0.592 (0.485) and total: 0.390 (0.319). The

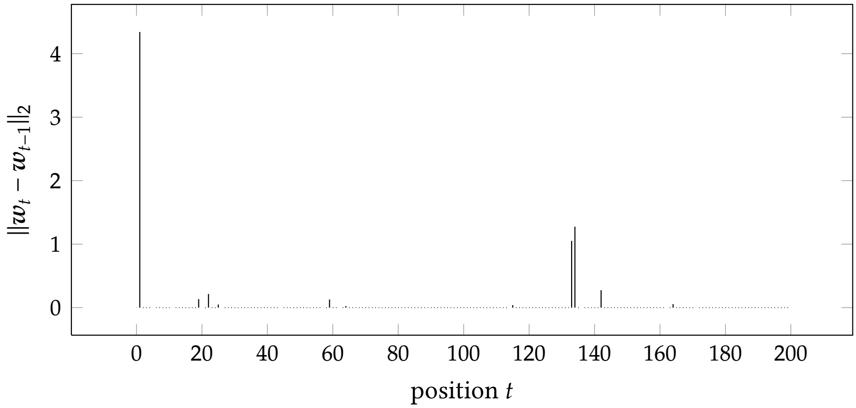

Figure 10.5: Sparsity pattern obtained from Lemma 10.2 for the nonlinear system described in Section 10.7.2 switching to a different dynamic between positions 133 and 135.

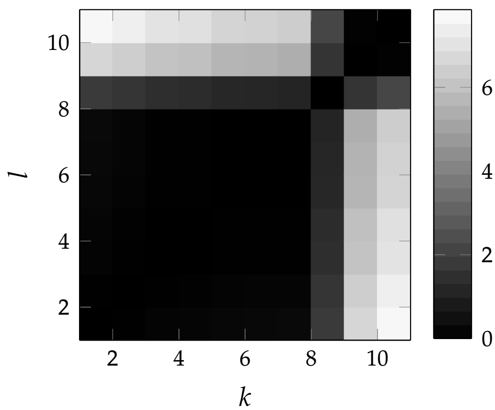

Figure 10.6: Norm $\|w_k - w_l\|_2$ for $k, l \in \{t : \|w_t - w_{t-1}\|_2 \geq 10^{-3} \max_t \|w_t - w_{t-1}\|_2\}$. The corresponding nonlinear system is specified in Section 10.7.2.

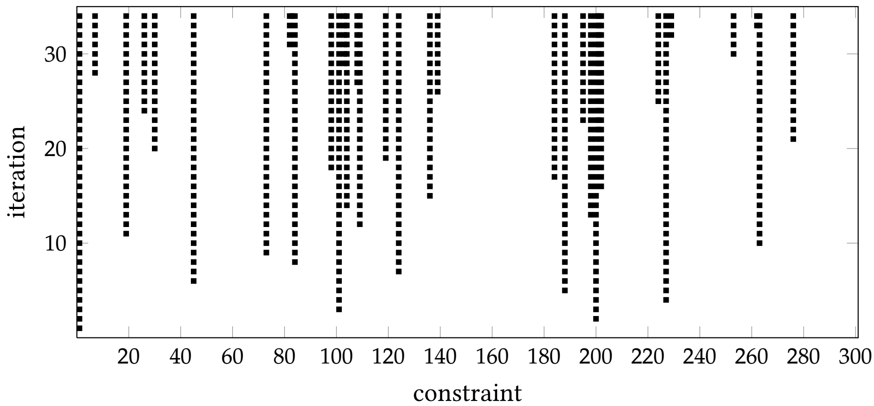

Figure 10.7: Active constraints as a function of iterations in the active set scheme described in Section 10.5.1. Black pixels indicate that a constraint belongs to the active set.

values given in parenthesis are the performances of a LS-SVM model given the true segmentation (10.3) and using a full model selection.

Once more the model is compared to a segmented ARX model. The scheme proposed by Ohlsson et al. [2010] is used to estimate a piecewise ARX model (10.2) with $x_t = [y_{t-1}, y_{t-2}, u_t, u_{t-1}, u_{t-2}]$. The change point at 134 is correctly detected and the root mean squared errors on an independent test set is for segment I: 0.310, segment II: 0.600 and total: 0.428, which is slightly worse than the proposed method.

10.7.3 Algorithm

For the example in Subsection 10.7.1 and the optimal value for γ the evolution of the active set along the iterations is shown in Figure 10.7. It can be observed that only a fraction of all constraints determine the optimal solution, in this case 34 out of 300. Also note that the first constraints that are included in the active set are the ones at $t = 100$ and $t = 200$ namely the positions of the two change points.

10.8 Conclusions

This chapter introduced a novel method for segmenting time-series from nonlinear dynamical systems. The proposed method uses sum-of-norms regularization to trade-off the number of segments and fit. Two examples motivate the use of a nonlinear underlying model instead of a linear one used in previous work. The fact that the method only has one design parameter, the regularization parameter, makes it user friendly and attractive for e.g. change detection, diagnosis and fault detection.

11 Conclusions

The three main themes of this thesis are nonlinear system identification, kernel based modeling and convex optimization. The goal was to explore how far the combination of the latter two can be taken to tackle problems posed by the first. From the modeling perspective a lot of work has been invested into ways to incorporate prior knowledge into the estimation problem. Shifting the perspective to an optimization point of view the foremost objective was to find convex approximations to nonconvex problems.

Overparametrization and regularization

Condensing this thesis to two main conclusions, one obtains i) that the concept of overparametrization can be applied to a large number of problems and ii) the possibilities are even larger once overparametrization is combined with new regularization schemes. To support these claims note that the relevant chapters of this text are just examples and many more applications can be found. However, some major challenges need to be solved within this context to make it even more effective.

Numerical complexity As soon as more elaborate regularization schemes are incorporated one faces two problems. On the one hand it is much more involved to obtain a dual finite dimensional estimation problem and the corresponding predictive model than it is for standard LS-SVMs. On the other hand, there is the associated numerical complexity. Besides being relatively straightforward to work with, LS-SVMs have the major advantage that the basic formulation "just" needs to solve a linear system. A method for their numerical solution is therefore usually of no concern. For many of the more interesting, and usually more powerful, optimization problems, the numerical solution is however still a very active and dynamic field. This provides confidence that all presented techniques can scale to real world problems in the future, even though they might not do so right now.

To put these statements into perspective, in classical parametric identification it often can be feasible to handle millions or tens of millions of data points while the order of computational effort for a single optimization problem can still be measured in minutes. In case of kernel based models for NARX models, estimation problems with several 100,000 data points can be solved in a similar time frame when solved with the fixed size approach. The exact limit depends on the amount of memory at one's disposal, the implementation and most importantly the number of support vectors one considers. In case of overparametrized models this limit is a bit lower as the number of data points is effectively multiplied by the amount of overparametrization. In practice this means one to two orders of magnitude compared to plain fixed size models. In the scope of the Wiener-Hammerstein benchmark, overparametrized models with 50,000 data points and 2,500 support vectors were estimated. Note however that while a single estimation problem is usually solved in minutes, the model selection process as well as tuning the model hyper-parameters often requires the estimation of hundreds of individual models. However, this applies to all identification techniques at least to some extent.

The scale of problems is much more limiting for the advanced regularization techniques. In case of the sum-of-norms formulation used to segment time series, the relevant optimization problem is a second order cone programming problem. The main complexity here is in the number of constraints. Another important factor is the number of detected segments as this determines how many of the constraints are active at the solution. Note that the model selection process will also evaluate models that give rise to far too many segments, which has to be taken into account. With general purpose solvers a reasonable limit for the number of data points is a few thousand at most. Finally the nuclear norm regularized models pose the biggest challenge in terms of computation. When solving these problems with general purpose SDP solvers, the practical limit is on the order of 1,000 data points. In this case some other crucial factors are the decision to solve the primal or the dual problem and in case of the primal the number of basis functions as well as the number of target variables. The computational complexity for this problem is briefly studied for Wiener-Hammerstein identification.

Model representation The second methodological challenge is finding the form of the kernel based estimation problem as well as the corresponding predictive model. This thesis provides several examples how the dual optimization problem can be derived in a straightforward manner. Although the presentation is different, these examples can serve as templates for finding dual problems for most convex, norm based regularization schemes. Solutions are also given for the predictive model. However, these have not yet reached the desired simplicity in all cases. Therefore, the results cannot necessarily be transferred to new problems in a straightforward manner. Improvements in this area would certainly improve the generality and adoption of these ideas.

Relatively straightforward cases that are already well studied are for instance the orthogonality constraint in case of partially linear models which gives rise to an equivalent kernel, embedding the additional information. Another slightly more complex example are the overparametrized models for Wiener-Hammerstein systems as well as the nonlinear models complemented by a linear noise model. For these models several representations are available, each having its own advantages and disadvantages. The original formulations are the most truthful representations of the underlying system, however they are difficult to identify numerically. The overparametrized formulations are suitable for numerical solution and often have good predictive performance but lack the direct physical interpretation of the original models. In between are the projected models that try to combine as many advantages as possible.

The most complicated as well as most novel results are with respect to models with nonquadratic regularization. The LS-SVM formulation based on the ℓ_2-norm as used for the sensitivity analysis already provides good insight how a kernel based optimization problem can be derived. An interesting open question in this context, which applies to all nonquadratic regularization schemes considered in this thesis, is whether it is possible to avoid factorizing the kernel matrix as this imposes an additional computational burden. Some simple answers to this exist, but these relate to solutions that are not any more attractive computationally. In case of the sensitivity analysis, the recovery of a model representation in terms of the kernel function is still mostly straightforward. This changes significantly for the sum-of-norms formulation and even more so for the nuclear norm regularization. In these cases in depth analysis of the relations between primal and dual variables is required to establish a link between the parametric and the kernel based model representations. A reassuring result in this context is that the final model representations are simple linear combinations of kernel functions. Therefore they are no more complex than a basic LS-SVM model. The whole complexity is in establishing the correct weights for each of the kernel functions.

Knowledge from other domains

As outlined above, one key point of this thesis is the application of a particular set of methods to a variety of problems. Next to this, knowledge is taken from the well-established linear identification domain and integrated with nonlinear concepts. In one way or another, this is part of most chapters. The most prominent example is the chapter on partially linear systems. Here a complete estimate obtained using linear methods is used to improve a nonlinear model. Even though the predictive performance of the model is not improved over classical partially linear models, it is assured that an interpretation attached to it remains valid. This can be achieved with minimal additional computational cost.

The chapter on block structured systems provides valuable insight into a particular application. To succeed, it uses information on the partial linearity of the studied system. By doing so, the described method is able to improve the quality of the nonlinear model. The main new contributions here are the systematic comparison of different approaches on a large scale real world data set. On this data it can be shown that taking prior information into account can reduce the error by more than 50%. This comparison has only been possible as the scope of the basic idea is extended to include a more general class of systems, from Hammerstein to Wiener-Hammerstein. Furthermore, to handle large datasets, additional techniques are incorporated and everything has been implemented in an efficient manner.

Handling broader model classes

Incorporation of existing knowledge into kernel based models is only one of the applications considered in this thesis. The second area of applications is the treatment of broader model classes. The most general example in this direction is the work on models with multiple outputs. While handling multiple outputs can be straightforwardly done in a kernel based framework, however most approaches can be reduced to estimating independent models for each output. In a lot of cases such an approach clearly does not facilitate all available information. With the introduction of nuclear norm regularization it could be illustrated that the predictive performance can be improved by considering the information from all outputs. Due to the theoretical complexity of this model formulation, the main contributions of this part are in the derivation of the model representation.

In cases where the noise corrupting a measurement is not a white process, it is necessary to also model the noise dynamics. In a kernel based setting that so far meant that a nonconvex problem had to be solved. With the work presented in this text it is now possible to obtain noise model estimates directly from a convex optimization problem. Also the feasible order for the noise model increased by an order of magnitude compared to some previously proposed methods.

Furthermore, it could be demonstrated that even for time dependent systems, kernel based models can be estimated without specific knowledge on the time dependence. For an unknown nonlinear system that switches its dynamics at unknown instances of time, approximate models for the system dynamics as well as time instances at which this systems switches can now be estimated from a convex optimization problem.

Future work and final outlook

Extensions to more problem classes Interesting topics for future work are the application of the proposed techniques to more model classes or extending the existing methodologies. In case of partially linear systems, one could derive nonlinear models that are coupled to linear models identified in different working points. Such an approach could also utilize the idea of linearizing kernel based models as used in the chapter on model sensitivity. In case of block structured systems a natural extension would be parallel cascades of Hammerstein systems and the analysis of their modeling power with respect to general nonlinear systems. Such a structure would be especially intriguing in combination with nuclear norm regularization that could provide an automatic selection of the number of parallel branches. In case of linear noise models one could evaluate the advantages and disadvantages of using a nonparametric model for the noise instead of a parametric one. The sum-of-norms formulation was used only to segment a model in time. An interesting question could be which other segmentations would be useful and whether these could be mapped onto a similar formulation. For the nuclear norm regularization there is vast amount of possible applications. For once every application of overparametrization can probably be enhanced by the use of nuclear norm regularization. It might also give rise to interesting segmentations, where a model is able to choose a combination of several "submodels" at each time instance. Furthermore, LS-SVM based models have been used for a lot more than system identification, it would be interesting to evaluate whether some of these formulations would benefit from a low rank assumption.

Application on real world problems Besides the application of the techniques utilized within this thesis to more problems, the most interesting and challenging task is the evaluation of the proposed models on more real world data sets. The next step from a mere evaluation would then be the application on real world problems. This requires a further in-depth analysis of the strengths as well as the weaknesses of the proposed models. Furthermore, an open mind is needed when looking for suitable applications. The field of nonlinear identification is very broad and the techniques covered in this thesis only consider very small subsets. Also the application of nonlinear identification techniques in practical applications is still in its infancy. The adoption of state of the art techniques could most likely be improved if easy to use software were available that does not require a large amount of expert knowledge as well as tuning from the potential user.

Scaling to larger problem sizes A major effort is needed to work on the scalability of the proposed methods. Some of them are already applicable to realistic problem sizes, while others still have to gain orders of magnitude before they can be considered suitable for real world problems. All advances in this respect directly improve the chances for real world applications as discussed in the previous paragraph. There are many potential avenues in the direction of better scaling. One could study the performance of approximate solutions. Tailored algorithms could be written to replace general purpose software. With the rise of multi-core and in case of graphics cards many-core computational architectures, the investigation of parallel processing is of increasing importance. Work in this direction has already started and methods like the alternating direction method of multipliers show promising results. On a modeling level ideas from compressed sensing might be incorporated to solve problems with even sparser data representations than with the fixed size approach. This paragraph gives a few possible directions but most likely many more are at least as suitable to tackle large problem sizes as those mentioned here.

Improved model representations As last topic for more investigation the work on model representations shall be mentioned. Although model representations have been derived for all models presented in this thesis, the interpretation of these links is only at a very basic level. More insight into these links might provide interesting knowledge on the problems themselves. Maybe this work would then also provide more compact formulations that establish the link between primal and dual.

Final outlook With the increase of computational power and advances in the algorithms for numerical optimizations, the application of all proposed methods on large scale problems will get within reach. At that point, especially the models based on nuclear norms for multiple output systems can be applied in many scenarios. It very naturally generalizes the single output to the multiple output problem, which in a nonlinear setting is not possible for many alternatives. In the introduction, the study of nonlinear identification has been motivated by the fact that real world problems are nonlinear. However, the same can be said for the number of outputs of most real world systems, only few have just one.

Appendix A

A.1 Proof of Theorem 6.6

The theorem is stated on page 107 and characterizes the set of all matrices $\boldsymbol{Z}$ that satisfy the relation $\operatorname{tr}(\boldsymbol{Z}^T\boldsymbol{X}) = \|\boldsymbol{X}\|_*$.

Denote the thin SVD of $\boldsymbol{Z}$ by $\boldsymbol{U\Sigma V}^T$. Note that the diagonal matrix of singular values $\boldsymbol{\Sigma}$ is block structured such that $\boldsymbol{\Sigma} = \operatorname{diag}(\boldsymbol{I}_r, \boldsymbol{\Sigma}_2)$ where $\boldsymbol{\Sigma}_2$ contains all singular values strictly smaller than one. In a first step it will be shown that every element of the set given by the left side of 6.13 is contained in the set given by the right side. The second part then shows the reverse and establishes the proof.

Let $\boldsymbol{U}_X\boldsymbol{\Sigma}_X\boldsymbol{V}_X^T$ be the thin SVD of the matrix $\boldsymbol{X}$ and $\boldsymbol{U}_X = [\boldsymbol{u}_{X,1}, \dots, \boldsymbol{u}_{X,P}]$, $\boldsymbol{V}_X = [\boldsymbol{v}_{X,1}, \dots, \boldsymbol{v}_{X,P}]$ and $\boldsymbol{\Sigma}_X = \operatorname{diag}(\sigma_{X,1}, \dots, \sigma_{X,P})$ where $P = \max(M, N)$. Then, from the definition of the set $\{\boldsymbol{X} : \operatorname{tr}(\boldsymbol{X}^T\boldsymbol{Z}) = \xi, \|\boldsymbol{X}\|_* = \xi\}$, the following relations can be derived

$$
\begin{aligned}
\xi &= \operatorname{tr}(\boldsymbol{X}^T\boldsymbol{Z}) = \operatorname{tr}(\boldsymbol{V}_X\boldsymbol{\Sigma}_X\boldsymbol{U}_X^T\boldsymbol{Z}) \\
&= \sum_{i=1}^{P} \operatorname{tr}(\sigma_{X,i}\boldsymbol{v}_{X,i}\boldsymbol{u}_{X,i}^T\boldsymbol{Z}) && \text{(formulation using rank-1 products)} \\
&= \sum_{i=1}^{P} \sigma_{X,i} \operatorname{tr}(\boldsymbol{u}_{X,i}^T\boldsymbol{Z}\boldsymbol{v}_{X,i}) && \text{(using cyclic property and linearity of trace)} \\
&\leq \sum_{i=1}^{P} \sigma_{X,i} \max_{\|\boldsymbol{u}\|_2, \|\boldsymbol{v}\|_2 \leq 1} \operatorname{tr}(\boldsymbol{u}^T\boldsymbol{Z}\boldsymbol{v}) && \text{(largest singular value of } \boldsymbol{Z}\text{)} \\
&= \sum_{i=1}^{P} \sigma_{X,i}\|\boldsymbol{Z}\|_2 && \text{(definition of spectral norm)} \\
&= \operatorname{tr}(\boldsymbol{\Sigma}_X) = \|\boldsymbol{X}\|_* = \xi. && (\|\boldsymbol{Z}\|_2 = 1\text{; definition of trace norm)}
\end{aligned}
$$

The first and the last equalities are directly taken from the definition of the set. As both sides of the inequality are identical, the inequality has to be satisfied with equality. This proves that $\boldsymbol{u}_{X,i}$ and $\boldsymbol{v}_{X,i}$ corresponding to nonzero $\sigma_{X,i}$ are singular vectors of $\boldsymbol{Y}$ corresponding to its largest singular value, one. Hence, for these one has $\boldsymbol{u}_{X,i} \in \operatorname{span}(\boldsymbol{U}_1)$ and $\boldsymbol{v}_{X,i} \in \operatorname{span}(\boldsymbol{V}_1)$ respectively. Therefore one can construct $\boldsymbol{X}$ as $\boldsymbol{U}_1\boldsymbol{H}_1\boldsymbol{V}_1^T$ as desired.

To verify the reverse direction $\|\boldsymbol{U}_1\boldsymbol{H}_1\boldsymbol{V}_1^T\|_*$ as well as $\operatorname{tr}(\boldsymbol{V}_1\boldsymbol{H}_1\boldsymbol{U}_1^T\boldsymbol{Z})$ have to equal ξ. The former is established by the unitary invariance of the nuclear norm, which provides $\|\boldsymbol{U}_1\boldsymbol{H}_1\boldsymbol{V}_1^T\|_* = \|\boldsymbol{H}_1\|_*$. The nuclear norm of a positive semidefinite matrix like $\boldsymbol{H}_1$ is directly given by its trace, which per definition is equal to ξ. For the second condition note that $\operatorname{tr}(\boldsymbol{V}_1\boldsymbol{H}_1\boldsymbol{U}_1^T\boldsymbol{Z}) = \operatorname{tr}(\boldsymbol{V}_1\boldsymbol{H}_1\boldsymbol{V}_1^T) = \operatorname{tr}(\boldsymbol{H}_1) = \xi$. The second equation can for example be argued with the cyclic invariance of the trace, while the last just states the definition.

A.2 Proof of Theorem 6.26: Singular value clipping

In the following the characterization of the subdifferential of the spectral norm is needed. Therefore it is reproduced here for reference.

Theorem A.1. *Let X be a rank r matrix. Denote its largest singular value by σ_1 and let C denote the multiplicity of σ_1. Furthermore let $\boldsymbol{U}_1$ and $\boldsymbol{V}_1$ contain the corresponding left and right singular vectors. Then the subdifferential of $\|\boldsymbol{X}\|_2$, the spectral norm of $\boldsymbol{X}$, can be written as*

$$\partial\|\boldsymbol{X}\|_2 = \{\boldsymbol{U}_1\boldsymbol{H}_1\boldsymbol{V}_1^T : \boldsymbol{H}_1 \succeq \boldsymbol{0}_{C\times C}, \operatorname{tr}(\boldsymbol{H}_1) = 1\}. \tag{A.1}$$

Proof. The proof is given by Watson [1992]. □

Relying on this characterization of the spectral norm, it will be shown that

$$\boldsymbol{X}^* = \boldsymbol{U}(\boldsymbol{\Sigma})_-\boldsymbol{V}^T$$

with $\boldsymbol{U}$, $\boldsymbol{V}$ and $(\boldsymbol{\Sigma})_-$ defined as in Theorem 6.26, is the solution to the proximal problem

$$\min_{\boldsymbol{X}} \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{Y}\|_F^2 + \eta\|\boldsymbol{X}\|_2. \tag{A.2}$$

Note that the $\|\boldsymbol{X} - \boldsymbol{Y}\|_F^2$ is strongly convex. Therefore the optimization problem (A.2) has a unique solution. It remains to be shown that $\boldsymbol{X}^*$ is a solution of (A.2). According to [Nesterov, 2004] a necessary and sufficient condition is that the zero element is in the subdifferential of the objective function at $\boldsymbol{X}^*$ or

$$\mathbf{0} \in \boldsymbol{X}^* - \boldsymbol{Y} + \eta\partial\|\boldsymbol{X}^*\|_2.$$

Therefore it is sufficient to show that $-\eta^{-1}(\boldsymbol{X}^* - \boldsymbol{Y}) \in \partial\|\boldsymbol{X}^*\|_2$. Note that the largest singular value of $\boldsymbol{X}^*$ is σ_C and denote its multiplicity by r_C. Also note that all singular values less than or equal to σ_C are not affected by the clipping operation.

Hence, by Theorem A.1, the subdifferential of $\|\cdot\|_2$ at $\boldsymbol{X}^*$ is

$$\partial\|\boldsymbol{X}^*\|_2 = \{\boldsymbol{U}_1\boldsymbol{H}_1\boldsymbol{V}_1^T : \boldsymbol{H}_1 \succeq \mathbf{0}_{r_C \times r_C}, \operatorname{tr}(\boldsymbol{H}_1) = 1\},$$

where $\boldsymbol{U}_1$ and $\boldsymbol{V}_1$ contain the singular vectors corresponding to σ_C.

Now, evaluating $\boldsymbol{Y} - \boldsymbol{X}^*$ one obtains $\boldsymbol{U}(\boldsymbol{\Sigma} - (\boldsymbol{\Sigma})_-)\boldsymbol{V}^T$. As the singular values less than or equal to σ_C are not affected by the clipping operation, $\boldsymbol{\Sigma} - (\boldsymbol{\Sigma})_-$ will have at most r_C nonzero singular values, i.e. those singular values of $\boldsymbol{Y}$ that are strictly larger than σ_C. Therefore $\boldsymbol{Y} - \boldsymbol{X}^* = \boldsymbol{U}_1\tilde{\boldsymbol{\Sigma}}\boldsymbol{V}_1^T$ where $\tilde{\boldsymbol{\Sigma}}$ contains the first r_C singular values of $\boldsymbol{\Sigma} - (\boldsymbol{\Sigma})_-$.

What remains to be shown is that $\tilde{\boldsymbol{\Sigma}}$ satisfies the conditions on $\boldsymbol{H}_1$. By its construction it is straightforward to see that $\tilde{\boldsymbol{\Sigma}}$ is positive semidefinite. Finally its trace can be computed as

$$\operatorname{tr}(\tilde{\boldsymbol{\Sigma}}) = \sum_{i=1}^{r_C} \sigma_i - \sigma_C = m_\Sigma(\sigma_C) = \eta.$$

This concludes the proof.

Bibliography

Absil, P.-A., Mahony, R., and Sepulchre, R. (2008). *Optimization Algorithms on Matrix Manifolds.* Princeton, NJ: Princeton University Press.

Akaike, H. (Dec. 1974). "Stochastic theory of minimal realization". In: *IEEE Transactions on Automatic Control* 19(6), pp. 667–674.

Alzate, C. (May 2009). "Support Vector Methods for Unsupervised Learning". PhD thesis. Katholieke Universiteit Leuven, Belgium.

Alzate, C., Espinoza, M., De Moor, B., and Suykens, J. A. K. (2009). "Identifying Customer Profiles in Power Load Time Series Using Spectral Clustering". In: *Proceedings of the 19th International Conference on Artificial Neural Networks.* ICANN '09. Limassol, Cyprus: Springer-Verlag, pp. 315–324.

Andersson, P. (1985). "Adaptive Forgetting in Recursive Identification Through Multiple Models". In: *International Journal of Control* 42(5), pp. 1175–1193.

Argyriou, A., Evgeniou, T., and Pontil, M. A. (Jan. 2008). "Convex multi-task feature learning". In: *Machine Learning* 73(3), pp. 243–272.

Argyriou, A., Micchelli, C. A., and Pontil, M. A. (2009). "When Is There a Representer Theorem? Vector Versus Matrix Regularizers". In: *Journal of Machine Learning Research* 10, pp. 2507–2529.

Aronszajn, N. (1950). "Theory of reproducing kernels". In: *Transactions of the American Mathematical Society* 68, pp. 337–404.

Åström, K. J. and Bohlin, T. (1965). "Numerical identification of linear dynamic systems from normal operating records". In: *Proceedings of the 2nd IFAC Symposium on the Theory of Self-Adaptive Control Systems.* Teddington, England, pp. 96–111.

Bach, F. R., Jenatton, R., Mairal, J., and Obozinski, G. (2011). "Convex optimization with sparsity-inducing norms". In: *Optimization for Machine Learning.* Neural Information Processing. MIT Press, pp. 19–53.

Bai, E.-W. (1998). "An optimal two-stage identification algorithm for Hammerstein-Wiener nonlinear systems". In: *Automatica* 34(3), pp. 333–338.

Beck, A. and Teboulle, M. (Mar. 2009). "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems". In: *SIAM Journal on Imaging Science* 2 (1), pp. 183–202.

Becker, S. R., Bobin, J., and Candès, E. J. (2009). *Nesta: A fast and accurate first-order method for sparse recovery.* Tech. rep. Pasadena, CA, USA: California Institute of Technology, pp. 1–37.

Becker, S. R., Candès, E. J., and Grant, M. C. (Sept. 2011). *Templates for convex cone problems with applications to sparse signal recovery.* Tech. rep., pp. 165–218.

— (2012). *TFOCS v1.1 user guide.* URL: http://cvxr.com/tfocs/.

Berlinet, A. and Thomas-Agnan, C. (2004). *Reproducing Kernel Hilbert Spaces in Probability and Statistics.* Kluwer Academic Publishers.

Bershad, N. J., Celka, P., and Mclaughlin, S. (2001). "Analysis of stochastic gradient identification of Wiener-Hammerstein systems for nonlinearities with Hermite polynomial expansions". In: *IEEE Transactions on Signal Processing* 49(5), pp. 1060–1072.

Bertsekas, D. P. (1996). *Constrained Optimization and Lagrange Multiplier Methods.* Nashua, NH, USA: Athena Scientific.

— (1999). *Nonlinear Programming.* 2nd edition. Belmont, Massachusetts: Athena Scientific.

Billings, S. A. and Fakhouri, S. (1978). "Identification of a class of nonlinear systems using correlation analysis". In: *Proceedings of IEE* 125, pp. 691–697.

Bodenstein, G. and Praetorius, H. M. (1977). "Feature extraction from the electroencephalogram by adaptive segmentation". In: *Proceedings of the IEEE* 65, pp. 642–652.

Boyd, S. P. and Chua, L. (Sept. 1983). "Uniqueness of a basic nonlinear structure". In: *IEEE Transactions on Circuits and Systems* 30, pp. 648–651.

— (July 1985). "Uniqueness of circuits and systems containing one nonlinearity". In: *IEEE Transactions on Automatic Control* 30, pp. 674–681.

Boyd, S. P. and Vandenberghe, L. (2004). *Convex Optimization.* Cambridge University Press.

Cai, J.-F., Candès, E. J., and Shen, Z. (2010). "A Singular Value Thresholding Algorithm for Matrix Completion". In: *SIAM Journal on Optimization* 20(4), pp. 1956–1982.

Candès, E. J., Romberg, J. K., and Tao, T. (2006a). "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information". In: *IEEE Transactions on Information Theory* 52(2), pp. 489–509.

— (2006b). "Stable signal recovery from incomplete and inaccurate measurements". In: *Communications on Pure and Applied Mathematics* 59(8), pp. 1207–1223.

Chandrasekaran, S., Golub, G. H., Gu, M., and Sayed, A. H. (1999). "An efficient algorithm for a bounded errors-in-variables model". In: *SIAM Journal on Matrix Analysis and Applications* 20, pp. 839–859.

Chang, F. H. I. and Luus, R. (1971). "A noniterative method for identification using Hammerstein model". In: *IEEE Transactions on Automatic Control* 16(5), pp. 464–468.

Cristianini, N. and Shawe-Taylor, J. (2000). *An introduction to Support Vector Machines: and other kernel-based learning methods.* Cambridge University Press.

Dahl, J. and Vandenberghe, L. (2011). *Python Software for Convex Optimization (CVXOPT).* URL: http://abel.ee.ucla.edu/cvxopt.

Dantzig, G. B. (1963). *Linear Programming and Extensions.* Vol. R-366-PR. Reports. RAND Corporation.

De Brabanter, K., De Brabanter, J., Suykens, J. A. K., and De Moor, B. (2010). "Optimized fixed-size kernel models for large data sets". In: *Computational Statistics & Data Analysis* 54(6), pp. 1484–1504.

De Brabanter, K., De Brabanter, J., Suykens, J. A. K., and De Moor, B. (June 2011). "Kernel Regression in the Presence of Correlated Errors". In: *Journal of Machine Learning Research* 12, pp. 1955–1976.

De Vito, E., Rosasco, L., Caponnetto, A., Piana, M., and Verri, A. (2004). "Some properties of regularized kernel methods". In: *Journal of Machine Learning Research* 5, pp. 1363–1390.

Deistler, M. (2002). "System Identification and Time Series Analysis: Past, Present, and Future". In: *Stochastic Theory and Control.* Ed. by B. Pasik-Duncan. Vol. 280. Lecture Notes in Control and Information Sciences. Springer Berlin / Heidelberg, pp. 97–109.

Donoho, D. L. (Apr. 2006). "Compressed sensing". In: *IEEE Transactions on Information Theory* 52(4), pp. 1289–1306.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). "Least angle regression". In: *Annals of Statistics* 32(2), pp. 407–499.

El Ghaoui, L. and Lebret, H. (1997). "Robust solutions to least-squares problems with uncertain data". In: *SIAM Journal on Matrix Analysis and Applications* 18, pp. 1035–1064.

El Ghaoui, L. and Gahinet, P. (July 1993). "Rank Minimization under LMI constraints: A Framework for Output Feedback Problems". In: *Proceedings of the Second European Control Conference.* Groningen, The Netherlands.

Enqvist, M. and Ljung, L. (Mar. 2005). "Linear approximations of nonlinear FIR systems for separable input processes". In: *Automatica* 41, pp. 459–473.

Espinoza, M., Falck, T., Suykens, J. A. K., and De Moor, B. (Sept. 2008). "Time Series Prediction using LS-SVMs". In: *Proceedings of the European Symposium on Time Series Prediction.* (Porvoo, Finland, Sept. 17–19, 2008), pp. 159–168.

Espinoza, M., Suykens, J. A. K., Belmans, R., and De Moor, B. (Oct. 2007). "Electric Load Forecasting – Using kernel based modeling for nonlinear system identification". In: *IEEE Control Systems Magazine* 27, pp. 43–57.

Espinoza, M., Suykens, J. A. K., and De Moor, B. (Oct. 2005a). “Kernel Based Partially Linear Models and Nonlinear Identification”. In: *IEEE Transactions on Automatic Control* 50, pp. 1602–1606.

— (Mar. 2005b). “LS-SVM Regression with Autocorrelated Errors”. In: *Proceedings of the 14th IFAC Symposium on System Identification.* Newcastle, Australia, pp. 582–587.

Falck, T., Dreesen, P., De Brabanter, K., Pelckmans, K., De Moor, B., and Suykens, J. A. K. (Nov. 2012). “Least-Squares Support Vector Machines for the Identification of Wiener-Hammerstein Systems”. In: *Control Engineering Practice* 20(11), pp. 1165–1174.

Falck, T., Ohlsson, H., Ljung, L., Suykens, J. A. K., and De Moor, B. (Aug. 2011). “Segmentation of time series from nonlinear dynamical systems”. In: *Proceedings of the 18th IFAC World Congress.* (Milan, Italy, Aug. 28–11, 2011), pp. 13209–13214.

Falck, T., Pelckmans, K., Suykens, J. A. K., and De Moor, B. (July 2009). “Identification of Wiener-Hammerstein Systems using LS-SVMs”. In: *Proceedings of the 15th IFAC Symposium on System Identification.* (Saint-Malo, France, July 6–8, 2009), pp. 820–825.

Falck, T., Signoretto, M., Suykens, J. A. K., and De Moor, B. (2010). *A two stage algorithm for kernel based partially linear modeling with orthogonality constraints.* Tech. rep. 10-03. ESAT-SISTA, K.U. Leuven.

Falck, T., Suykens, J. A. K., and De Moor, B. (Dec. 2009). “Robustness analysis for Least Squares Kernel Based Regression: an Optimization Approach”. In: *Proceedings of the 48th IEEE Conference on Decision and Control.* (Shanghai, China, Dec. 16–18, 2009), pp. 6774–6779.

Falck, T., Suykens, J. A. K., and De Moor, B. (Dec. 2010). “Linear Parametric Noise Models for Least Squares Support Vector Machines”. In: *Proceedings of the 49th IEEE Conference on Decision and Control.* (Atlanta, GA, USA, Dec. 15–17, 2010), pp. 6389–6394.

Falck, T., Suykens, J. A. K., Schoukens, J., and De Moor, B. (Dec. 2010). “Nuclear Norm Regularization for Overparametrized Hammerstein Systems”. In: *Proceedings of the 49th IEEE Conference on Decision and Control.* (Atlanta, GA, USA, Dec. 15–17, 2010), pp. 7202–7207.

Fazel, M. (2002). “Matrix Rank Minimization with Applications”. PhD. Stanford.

Fazel, M., Hindi, H. A., and Boyd, S. P. (2001). “A rank minimization heuristic with application to minimum order system approximation”. In: *Proceedings of the American Control Conference.* Arlington, VA, USA.

Fine, S. and Scheinberg, K. (2002). “Efficient SVM training using low-rank kernel representations”. In: *Journal Machine Learning Research* 2, pp. 243–264.

Gevers, M. (2005). “Identification for Control: from the early Achievements to the Revival of Experiment Design”. In: *European Journal of Control* 11(4-5), p. 15.

— (Dec. 2006). “A personal view of the development of system identification: A 30-year journey through an exciting field”. In: *IEEE Control Systems Magazine* 26(6), pp. 93–105.

Gevers, M. and Ljung, L. (1986). “Optimal experiment designs with respect to the intended model application”. In: *Automatica* 22(5), pp. 543–554.

Giri, F. and Bai, E.-W., eds. (2010). *Block-oriented Nonlinear System Identification.* Vol. 404. Lecture notes in control and information sciences. Springer.

Girolami, M. (Mar. 2002). “Orthogonal series density estimation and the kernel eigenvalue problem”. In: *Neural Computation* 14, pp. 669–688.

Goemans, M. X. and Williamson, D. P. (Nov. 1995). “Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming”. In: *Journal of the ACM* 42 (6), pp. 1115–1145.

Goethals, I., Pelckmans, K., Falck, T., Suykens, J. A. K., and De Moor, B. (2010). “NARX Identification of Hammerstein Systems using Least-Squares Support Vector Machines”. In: *Block-oriented Nonlinear System Identification.* Ed. by F. Giri and E.-W. Bai. Vol. 404. Lecture notes in control and information sciences. Springer. Chap. 15, pp. 241–256.

Goethals, I., Pelckmans, K., Suykens, J. A. K., and De Moor, B. (2005a). “Identification of MIMO Hammerstein models using least squares support vector machines”. In: *Automatica* 41, pp. 1263–1272.

— (Oct. 2005b). “Subspace identification of Hammerstein systems using least squares support vector machines”. In: *IEEE Transactions on Automatic Control* 50, pp. 1509–1519.

Goldfarb, D. and Ma, S. (June 2009). *Convergence of fixed point continuation algorithms for matrix rank minimization.*

Golub, G. H. and Van Loan, C. V. (1996). *Matrix Computations.* 3rd edition. Baltimore, MD: Johns Hopkins University Press.

Grant, M. and Boyd, S. P. (Jan. 2011). *CVX: Matlab Software for Disciplined Convex Programming, version 1.21.* http://cvxr.com/cvx.

Greblicki, W. and Pawlak, M. (2008). *Non-Parametric System Identification.* Cambridge University Press.

Ha Quang, M., Pillonetto, G., and Chiuso, A. (Aug. 2009). “Nonlinear System Identification Via Gaussian Regression and Mixtures of Kernels”. In: *Proceedings of the 15th IFAC Symposium on System Identification.* Saint-Malo, France, pp. 528–533.

Härdle, W., Liang, H., and Gao, J. (2000). *Partially linear models.* Springer.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* 2nd edition. Springer.

Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation.* Prentice Hall PTR Upper Saddle River, NJ, USA.

Henrion, D. and Lasserre, J.-B. (June 2004). “Solving nonconvex optimization problems”. In: *IEEE Control Systems Magazine* 24(3), pp. 72–83.

Hjalmarsson, H., Rojas, C. R., and Rivera, D. E. (2012). “System identification: A Wiener-Hammerstein benchmark”. In: *Control Engineering Practice* 20(11), pp. 1095–1096.

Ho, B. L. and Kalman, R. E. (1966). “Effective construction of linear state-variable models from input/output functions (Algorithm for minimal realization of linear finite-dimensional dynamical system displayed by Markov parameters)”. In: *Regelungstechnik* 14(12), pp. 545–548.

Horn, R. A. and Johnson, C. R. (1990). *Matrix Analysis.* Cambridge University Press.

Huang, G., Song, S., Wu, C., and You, K. (Nov. 2012). "Robust Support Vector Regression for Uncertain Input and Output Data". In: *IEEE Transactions on Neural Networks and Learning Systems* 23(11), pp. 1690–1700.

Huber, P. J. and Ronchetti, E. (2009). *Robust statistics.* John Wiley and Sons.

International Business Machines Corporation (2010). *IBM ILOG CPLEX V12.1 – User's Manual for CPLEX.*

Jaggi, M. and Sulovský, M. (2010). "A Simple Algorithm for Nuclear Norm Regularized Problems". In: *Proceedings of the 27th International Conference on Machine Learning.* Haifa, Israel.

Jenatton, R., Audibert, J.-Y., and Bach, F. R. (2009). "Active Set Algorithm for Structured Sparsity-Inducing Norms". In: *Proceedings of the 2nd NIPS Workshop on Optimization for Machine Learning.* Whistler, Canada, p. 6.

Ji, S. and Ye, J. (2009). "Accelerated Gradient Method for Trace Norm Minimization". In: *Proceedings of the 26th International Conference on Machine Learning (ICML).* Montreal, Canada, pp. 457–464.

Juditsky, A., Hjalmarsson, H., Benveniste, A., Delyon, B., Ljung, L., Sjoberg, J., and Zhang, Q. (Dec. 1995). "Nonlinear black-box models in system identification: Mathematical foundations". In: *Automatica* 31, pp. 1725–1750.

Kailath, T. (1980). *Linear Systems.* Prentice-Hall.

Kalman, R. E. (1960a). "A new approach to linear filtering and prediction problems". In: *Journal of Basic Engineering.* D 82, pp. 35–45.

— (1960b). "Contributions to the theory of optimal control". In: *Boletín de la Sociedad Matemática Mexicana* 5, pp. 102–119.

Kantz, H. and Schreiber, T. (2003). *Nonlinear Time Series Analysis.* 2nd edition. Cambridge University Press.

Karger, D., Motwani, R., and Sudan, M. (Mar. 1998). "Approximate graph coloring by semidefinite programming". In: *Journal of the ACM* 45 (2), pp. 246–265.

Karmarkar, N. (1984). "A new polynomial-time algorithm for linear programming". In: *Combinatorica* 4(4) (4), pp. 373–395.

Katayama, T. (2005). *Subspace methods for system identification.* Communications and Control Engineering. Springer.

Khalil, H. K. (2002). *Nonlinear systems.* 3rd edition. Prentice Hall.

Kim, S.-J., Koh, K., Boyd, S. P., and Gorinevsky, D. (2009). “ℓ_1 Trend Filtering”. In: *SIAM Review* 51(2), pp. 339–360.

Kimeldorf, G. and Wahba, G. (1971). “Some results on Tchebycheffian spline functions”. In: *Journal of Mathematical Analysis and Applications* 33, pp. 82–95.

Krige, D. (1951). “A statistical approach to some mine valuations and allied problems at the Witwatersrand”. MA thesis. Johannesburg, South Africa: University of Witwatersrand.

Lasserre, J.-B. (2001). “Global Optimization with Polynomials and the Problem of Moments”. In: *SIAM Journal on Optimization* 11(3), pp. 796–817.

Lázaro, M., Santamaría, I., Pérez-Cruz, F., and Artés-Rodríguez, A. (Dec. 2005). “Support Vector Regression for the simultaneous learning of a multivariate function and its derivatives”. In: *Neurocomputing* 69(3), pp. 42–61.

Lendasse, A., Honkela, T., and Simula, O. (June 2010). “European symposium on time series prediction”. In: *Neurocomputing* 73 (10–12), pp. 1919–1922.

Li, Y.-F., Li, L.-J., Su, H.-Y., and Chu, J. (2006). “Least Squares Support Vector Machine Based Partially Linear Model Identification”. In: *Lecture Notes in Computer Science* 4113/2006, pp. 775–781.

Lindgren, G. (1978). “Markov regime models for mixed distributions and switching regressions”. In: *Scandinavian Journal of Statistics* 5, pp. 81–91.

Liu, J., Ji, S., and Ye, J. (2009). *SLEP: Sparse Learning with Efficient Projections.* Arizona State University. URL: http://www.public.asu.edu/~jye02/Software/SLEP.

Liu, Z. and Vandenberghe, L. (Jan. 2009). “Interior-Point Method for Nuclear Norm Approximation with Application to System Identification”. In: *SIAM Journal on Matrix Analysis and Applications* 31(3), pp. 1235–1256.

Ljung, L. (1999). *System identification: Theory for the User.* 2nd edition. Upper Saddle River, NJ, USA: Prentice Hall PTR.

Ljung, L. (2010). "Perspectives on system identification". In: *Annual Reviews in Control* 34(1), pp. 1–12.

Löfberg, J. (2004). "YALMIP : A Toolbox for Modeling and Optimization in MATLAB". In: *Proceedings of the 2004 IEEE International Symposium on Computer Aided Control Systems Design.* Taipei, Taiwan, pp. 284–289.

Luenberger, D. (1998). *Optimization by Vector Space Methods.* Wiley-IEEE.

Luo, Z.-q., Ma, W.-k., So, A.-C., Ye, Y., and Zhang, S. (May 2010). "Semidefinite Relaxation of Quadratic Optimization Problems". In: *IEEE Signal Processing Magazine* 27(3), pp. 20–34.

MacKay, D. J. C. (1999). "Comparison of approximate methods for handling hyperparameters". In: *Neural Computation* 11, pp. 1035–1068.

Martin, R. J. (2000). "A Metric for ARMA Processes". In: *IEEE Transactions on Signal Processing* 48(4), pp. 1164–1170.

Mehrkanoon, S., Falck, T., and Suykens, J. A. K. (Sept. 2012a). "Approximate Solutions to Ordinary Differential Equations Using Least Squares Support Vector Machines". In: *IEEE Transactions on Neural Networks and Learning Systems* 23(9), pp. 1356–1367.

— (July 2012b). "Parameter Estimation for Time Varying Dynamical Systems using Least Squares Support Vector Machines". In: *Proceedings of the 16th IFAC Symposium on System Identification.* (Brussels, Belgium, July 11–13, 2012), pp. 1300–1305.

Mercer, J. (1909). "Functions of positive and negative type, and their connection with the theory of integral equations". In: *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character* 209, pp. 415–446.

Mesbahi, M. (1998). "On the rank minimization problem and its control applications". In: *Systems & Control Letters* 33(1), pp. 31–36.

Mesbahi, M. and Papavassilopoulos, G. (Feb. 1997). "On the rank minimization problem over a positive semidefinite linear matrix inequality". In: *IEEE Transactions on Automatic Control* 42(2), pp. 239–243.

Mosek (2011). *The MOSEK optimization software.* URL: http://www.mosek.com.

Narendra, K. S. and Parthasarathy, K. (1990). "Identification and control of dynamical systems using neural networks". In: *IEEE Transactions on Neural Networks* 1(1), pp. 4–27.

Nelles, O. (2001). *Nonlinear system identification: from classical approaches to neural networks and fuzzy models.* Springer Verlag.

Nesterov, Yu. (1998). "Semidefinite relaxation and nonconvex quadratic optimization". In: *Optimization Methods and Software* 9(1), pp. 141–160.

— (2004). *Introductory Lectures on Convex Optimization: A Basic Course.* Vol. 87. Applied Optimization. Kluwer Academic Publishers.

— (2005). *Gradient Methods for Minimizing Composite Objective Function.* ECORE discussion paper 2007/96. ECORE.

Nesterov, Yu. and Nemirovskii, A. (1994). *Interior-Point Polynomial Algorithms in Convex Programming.* Studies in applied mathematics; 13. SIAM.

Nocedal, J. and Wright, S. J. (2006). *Numerical Optimization.* 2nd edition. New York, NY, USA: Springer.

Ohlsson, H., Ljung, L., and Boyd, S. P. (2010). "Segmentation of ARX-models using sum-of-norms regularization". In: *Automatica* 46(6), pp. 1107–1111.

Ojeda, F., Falck, T., De Moor, B., and Suykens, J. A. K. (July 2010). "Polynomial componentwise LS-SVM: fast variable selection using low rank updates". In: *Proceedings of the International Joint Conference on Neural Networks 2010.* (Barcelona, Spain, July 18–23, 2010), pp. 3291–3297.

Oppenheim, A. V., Willsky, A. S., and Nawab, S. H. (1997). *Signals and systems.* 2nd edition. Prentice-Hall.

Petersen, K. B. and Pedersen, M. S. (2008). *The Matrix Cookbook.* URL: http://www2.imm.dtu.dk/pubdb/p.php?3274.

Pintelon, R. and Schoukens, J. (2001). *System identification: a frequency domain approach.* John Wiley and Sons.

Poggio, T. and Girosi, F. (1990). "Networks for Approximation and Learning". In: *Proceedings of IEEE* 78(9), pp. 1481–1497.

Pong, T. K., Tseng, P., Ji, S., and Ye, J. (2010). "Trace Norm Regularization: Reformulations, Algorithms, and Multi-Task Learning". In: *SIAM Journal on Optimization* 20(6), pp. 3465–3489.

Rasmussen, C. E. and Ghahramani, Z. (Dec. 2001). "Occam's razor". In: *Advances in neural information processing systems 13.* Cambridge, MA, USA: The MIT Press, pp. 294–300.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian processes for machine learning.* Springer.

Recht, B., Fazel, M., and Parrilo, P. A. (Jan. 2010). "Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization". In: *SIAM Review* 52(3), pp. 471–501.

Renault, R. A., Guo, H., and Chen, W. J. (May 2005). *Regularised Total Least Squares Support Vector Machines.* Presentation. URL: http://math.asu.edu/~rosie/mypresentations/Rosie_talk_svmc.pdf.

Rosen, J. B., Park, H., and Glick, J. (Sept. 1998). "Structured total least norm for nonlinear problems". In: *SIAM Journal on Matrix Analysis and Applications* 20, pp. 14–30.

Rudin, L. I., Osher, S., and Fatemi, E. (1992). "Nonlinear total variation based noise removal algorithms". In: *Physica D: Nonlinear Phenomena* 60(1–4), pp. 259–268.

Schölkopf, B., Herbrich, R., and Smola, A. J. (2001). "A Generalized Representer Theorem". In: *Proceedings of the Annual Conference on Computational Learning Theory*, pp. 416–426.

Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels.* MIT Press Cambridge, Mass.

Schölkopf, B., Smola, A. J., Williamson, R. C., and Bartlett, P. L. (2000). "New Support Vector Algorithms". In: *Neural Computation* 12, pp. 1207–1245.

Schoukens, J., Nemeth, J. G., Crama, P., Rolain, Y., and Pintelon, R. (2003). "Fast approximate identification of nonlinear systems". In: *Automatica* 39, pp. 1267–1274.

Schoukens, J., Suykens, J. A. K., and Ljung, L. (2009). "Wiener-Hammerstein benchmark". In: *Proceedings of the 15th IFAC Symposium on System Identification.* Saint-Malo, France.

Shivaswamy, P. K., Bhattacharyya, C., and Smola, A. J. (2006). "Second Order Cone Programming Approaches for Handling Missing and Uncertain Data". In: *Journal Machine Learning Research* 7, pp. 1283–1314.

Shor, N. Z. (1987). "Quadratic optimization problems". In: *Tekhnicheskaya Kibernetika* 1, pp. 128–139.

Sjoberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, B., Glorennec, P.-Y., Hjalmarsson, H., and Juditsky, A. (Dec. 1995). "Nonlinear black-box modeling in system identification: a unified overview". In: *Automatica* 31, pp. 1691–1724.

Smola, A. J. and Schölkopf, B. (2004). "A tutorial on support vector regression". In: *Statistics and Computing* 14(3), pp. 199–222.

Söderström, T. and Stoica, P. (1989). *System Identification.* Prentice-Hall.

Speckman, P. (1988). "Kernel Smoothing in Partial Linear Models". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 50(3), pp. 413–436.

Steinwart, I. and Christmann, A. (2008). *Support Vector Machines.* Springer.

Sturm, J. (1999). "Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones". In: *Optimization Methods and Software* 11(1), pp. 625–653.

Suykens, J. A. K., Alzate, C., and Pelckmans, K. (Aug. 2010). "Primal and dual model representations in kernel-based learning". In: *Statistics Surveys* 4, pp. 148–183.

Suykens, J. A. K., De Brabanter, J., Lukas, and Vandewalle, J. (2002). "Weighted least squares support vector machines: robustness and sparse approximation". In: *Neurocomputing* 48(1-4), pp. 85–105.

Suykens, J. A. K., Van Gestel, T., De Brabanter, J., De Moor, B., and Vandewalle, J. (2002). *Least Squares Support Vector Machines.* World Scientific.

Suykens, J. A. K. and Vandewalle, J. (1999). "Least squares support vector machine classifiers". In: *Neural processing letters* 9(3), pp. 293–300.

Suykens, J. A. K. and Vandewalle, J. (2000). "Recurrent least squares support vector machines". In: *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 47, pp. 1109–1114.

Suykens, J. A. K., Vandewalle, J., and De Moor, B. (1995). *Artificial Neural Networks for Modelling and Control of Non-Linear Systems.* Springer.

Takens, F. (1981). "Detecting strange attractors in turbulence". In: *Dynamical Systems and Turbulence, Warwick 1980.* Ed. by D. Rand and L.-S. Young. Vol. 898. Lecture Notes in Mathematics. Springer, pp. 366–381.

Tan, A. H. and Godfrey, K. (2002). "Identification of Wiener-Hammerstein models using linear interpolation in the frequency domain (LIFRED)". In: *IEEE Transactions on Instrumentation and Measurement* 51, pp. 509–521.

Tibshirani, R. (1996). "Regression Shrinkage and Selection via the Lasso". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 58(1), pp. 267–288.

Toh, K.-C., Todd, M. J., and Tutuncu, R. H. (1999). "SDPT3 — a Matlab software package for semidefinite programming". In: *Optimization Methods and Software* 11, pp. 545–581.

Toh, K.-C. and Yun, S. (Sept. 2010). "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems". In: *Pacific Journal of Optimization* 6, pp. 615–640.

Trafalis, T. B. and Gilbert, R. C. (Sept. 2006). "Robust classification and regression using support vector machines". In: *European Journal of Operational Research* 173, pp. 893–909.

Tseng, P. (Aug. 2010). "Approximation accuracy, gradient methods, and error bound for structured convex optimization". In: *Mathematical Programming, Series B*, p. 33.

Tugnait, J. K. (1982). "Detection and estimation for abruptly changing systems". In: *Automatica* 18, pp. 607–615.

Van Den Berg, E. and Friedlander, M. P. (2008). "Probing the Pareto frontier for basis pursuit solutions". In: *SIAM Journal on Scientific Computing* 31(2), pp. 890–912.

Van Herpe, T., Mesotten, D., Falck, T., De Moor, B., and Van den Berghe, G. T. (Feb. 2010). "LOGIC-Insulin Algorithm for Blood Glucose Control in the ICU: a pilot test". At: Third International Conference on Advanced Technologies & Treatments for Diabetes (Basel, Switzerland, Feb. 10–13, 2010).

Van Huffel, S. and Vandewalle, J. (1991). *The Total Least Squares Problem : Computational Aspects and Analysis, Frontiers in Applied Mathematics Series, Vol. 9.* SIAM, Philadelphia.

Van Overschee, P. and De Moor, B. (1996). *Subspace Identification for Linear Systems, Theory, Implementation, Applications.* Kluwer Academic Publishers.

Vapnik, V. N. (1998). *Statistical Learning Theory.* John Wiley & Sons.

Vapnik, V. N. and Lerner, A. (1963). "Pattern recognition using generalized portrait method". In: *Automation and Remote Control* 24, pp. 774–780.

Wahba, G. (1984). "Partial spline models for the semiparametric estimation of functions of several variables". In: *Statistical analysis of time series.* Tokyo: Institute of Mathematical Statistics, pp. 319–329.

— (1990). *Spline Models for Observational Data.* SIAM.

— (May 1998). "Support Vector Machines, Reproducing Kernel Hilbert Spaces and the Randomized GACV". In: *Advances in Kernel Methods - Support Vector Learning.* Cambridge, MA: MIT Press, pp. 69–88.

Watson, G. A. (1992). "Characterization of the Subdifferential of Some Matrix Norms". In: *Linear Algebra and its Applications* 45, pp. 33–45.

— (2003). "Robust solutions to a general class of approximation problems". In: *SIAM Journal on Scientific Computing* 25, pp. 1448–1460.

— (2007). "Robust counterparts of errors-in-variables problems". In: *Computational Statistics and Data Analysis*, pp. 1080–1089.

Willems, J. C. (Dec. 2007). "The Behavioral Approach to Open and Interconnected Systems". In: *IEEE Control Systems Magazine* 27(6), pp. 46–99.

Williams, C. K. I. and Seeger, M. (2001). "Using the Nyström Method to Speed Up Kernel Machines". In: *Neural Information Processing Systems 13.* Ed. by T. Leen, T. Dietterich, and V. Tresp. MIT Press, pp. 682–688.

Wright, M. H. (2005). "The interior-point revolution in optimization: history, recent developments, and lasting consequences". In: *Bulletin of the American Mathematical Society (New Series)* 42, pp. 39–56.

Xu, Y.-L. and Chen, D.-R. (2009). "Partially-Linear Least-Squares Regularized Regression for System Identification". In: *IEEE Transactions on Automatic Control* 54(11), pp. 2637–2641.

Yu, S., Falck, T., Daemen, A., Tranchevent, L.-C., Suykens, J. A. K., De Moor, B., and Moreau, Y. (2010). "L2-norm multiple kernel learning and its application to biomedical data fusion". In: *BMC Bioinformatics* 11(309), pp. 1–53.

Zhang, Q. and Benveniste, A. (Nov. 1992). "Wavelet networks". In: *IEEE Transactions on Neural Networks* 3(6), pp. 889–898.

Curriculum vitae

Tillmann Falck was born in Bochum, Germany on June 15th, 1982. He received the “Abitur” from Hellweg Schule, Bochum in 2001, with majors in Mathematics and Physics. In the same year he started his studies in Electrical Engineering at Ruhr University Bochum. He spent the academic year 04/05 at Purdue University, W. Lafayette, IN, USA. In January 2007 he received the degree “Diplom-Ingenieur” in Electrical Engineering after completing his thesis “Improvement of ultrasound contrast agent detection using nonlinear signal processing”. From February 2007 he was a PhD student at the SISTA/SMC research group at KU Leuven, Belgium. His research interests included (nonlinear) system identification, kernel based machine learning and convex optimization. After moving to Stuttgart, Germany he started working for Robert Bosch GmbH in the area of radar based driver assistance systems in May 2012.

List of Publications

Falck, **T.**, Dreesen, P., De Brabanter, K., Pelckmans, K., De Moor, B., and Suykens, J. A. K. (Nov. 2012). “Least-Squares Support Vector Machines for the Identification of Wiener-Hammerstein Systems”. In: *Control Engineering Practice* 20(11), pp. 1165–1174.

Mehrkanoon, S., **Falck**, **T.**, and Suykens, J. A. K. (Sept. 2012a). “Approximate Solutions to Ordinary Differential Equations Using Least Squares Support Vector Machines”. In: *IEEE Transactions on Neural Networks and Learning Systems* 23(9), pp. 1356–1367.

— (July 2012b). “Parameter Estimation for Time Varying Dynamical Systems using Least Squares Support Vector Machines”. In: *Proceedings of the 16th IFAC Symposium on System Identification.* (Brussels, Belgium, July 11–13, 2012), pp. 1300–1305.

Falck, **T.**, Ohlsson, H., Ljung, L., Suykens, J. A. K., and De Moor, B. (Aug. 2011). “Segmentation of time series from nonlinear dynamical systems”. In: *Proceedings of the 18th IFAC World Congress.* (Milan, Italy, Aug. 28–11, 2011), pp. 13209–13214.

Falck, **T.**, Signoretto, M., Suykens, J. A. K., and De Moor, B. (2010). *A two stage algorithm for kernel based partially linear modeling with orthogonality constraints.* Tech. rep. 10-03. ESAT-SISTA, K.U. Leuven.

Falck, **T.**, Suykens, J. A. K., and De Moor, B. (Dec. 2010). “Linear Parametric Noise Models for Least Squares Support Vector Machines”. In: *Proceedings of the 49th IEEE Conference on Decision and Control.* (Atlanta, GA, USA, Dec. 15–17, 2010), pp. 6389–6394.

Falck, **T.**, Suykens, J. A. K., Schoukens, J., and De Moor, B. (Dec. 2010). “Nuclear Norm Regularization for Overparametrized Hammerstein Systems”. In: *Proceedings of the 49th IEEE Conference on Decision and Control.* (Atlanta, GA, USA, Dec. 15–17, 2010), pp. 7202–7207.

Goethals, I., Pelckmans, K., **Falck**, **T.**, Suykens, J. A. K., and De Moor, B. (2010). “NARX Identification of Hammerstein Systems using Least-Squares Support Vector Machines”. In: *Block-oriented Nonlinear System Identification.* Ed. by F. Giri and E.-W. Bai. Vol. 404. Lecture notes in control and information sciences. Springer. Chap. 15, pp. 241–256.

Ojeda, F., **Falck**, **T.**, De Moor, B., and Suykens, J. A. K. (July 2010). "Polynomial componentwise LS-SVM: fast variable selection using low rank updates". In: *Proceedings of the International Joint Conference on Neural Networks 2010.* (Barcelona, Spain, July 18–23, 2010), pp. 3291–3297.

Yu, S., **Falck**, **T.**, Daemen, A., Tranchevent, L.-C., Suykens, J. A. K., De Moor, B., and Moreau, Y. (2010). "L2-norm multiple kernel learning and its application to biomedical data fusion". In: *BMC Bioinformatics* 11(309), pp. 1–53.

Falck, **T.**, Pelckmans, K., Suykens, J. A. K., and De Moor, B. (July 2009). "Identification of Wiener-Hammerstein Systems using LS-SVMs". In: *Proceedings of the 15th IFAC Symposium on System Identification.* (Saint-Malo, France, July 6–8, 2009), pp. 820–825.

Falck, **T.**, Suykens, J. A. K., and De Moor, B. (Dec. 2009). "Robustness analysis for Least Squares Kernel Based Regression: an Optimization Approach". In: *Proceedings of the 48th IEEE Conference on Decision and Control.* (Shanghai, China, Dec. 16–18, 2009), pp. 6774–6779.

Espinoza, M., **Falck**, **T.**, Suykens, J. A. K., and De Moor, B. (Sept. 2008). "Time Series Prediction using LS-SVMs". In: *Proceedings of the European Symposium on Time Series Prediction.* (Porvoo, Finland, Sept. 17–19, 2008), pp. 159–168.

Bandyopadhyay, S., Coyle, E. J., and **Falck**, **T.** (Nov. 2007). "Stochastic Properties of Mobility Models in Mobile Ad Hoc Networks". In: *IEEE Transactions on Mobile Computing* 6(11), pp. 1218–1229.

— (Mar. 2006). "Stochastic Properties of Mobility Models in Mobile Ad Hoc Networks". In: *Proceedings of the 40th Annual Conference on Information Sciences and Systems.* (Princeton, NJ, USA, Mar. 22–24, 2006), pp. 1205–1211.

List of Presentations

Falck, **T.**, Suykens, J. A. K., De Moor, B., Ohlsson, H., and Ljung, L. (Mar. 2011). "Kernel based models and sum-of-norms regularization for nonlinear time series segmentation". At: 30th Benelux Meeting on Systems and Control (Lommel, Belgium, Mar. 15–17, 2011). oral presentation.

Falck, **T.**, Signoretto, M., Suykens, J. A. K., and De Moor, B. (2010). "Estimation of Hammerstein Systems using Nuclear Norm Regularization and Overparametrization". At: 19th ERNSI Workshop in System Identification (Cambridge, UK, Sept. 27, 2010–Sept. 29, 2011). oral presentation.

Falck, **T.**, Suykens, J. A. K., De Moor, B., and Schoukens, J. (Mar. 2010). "Using Linear System Estimates within Least Squares Support Vector Machines". At: 29th Benelux Meeting on Systems and Control (Heeze, The Netherlands, Mar. 30–Apr. 1, 2010). oral presentation.

Van Herpe, T., Mesotten, D., **Falck**, **T.**, De Moor, B., and Van den Berghe, G. T. (Feb. 2010). "LOGIC-Insulin Algorithm for Blood Glucose Control in the ICU: a pilot test". At: Third International Conference on Advanced Technologies & Treatments for Diabetes (Basel, Switzerland, Feb. 10–13, 2010).

Falck, **T.**, Signoretto, M., Suykens, J. A. K., and De Moor, B. (2009). "Imposing orthogonality on LS-SVM based models". At: 18th ERNSI Workshop in System Identification (Stift Vorau, Austria, Sept. 30, 2010–Oct. 2, 2011). poster presentation.

Falck, **T.**, Espinoza, M., Suykens, J. A. K., and De Moor, B. (Sept. 2008a). "Least Squares Support Vector Machines für nichtlineare Systemidentifikation". At: VDE GMA-Fachausschuss 1.30 "Modellbildung, Identifikation und Simulation in der Automatisierungstechnik" (Salzburg, Austria, Sept. 24–26, 2008). oral presentation.

— (Oct. 2008b). "Robust kernel based regression in SOCP and LS formulations for perturbation analysis". At: 17th ERNSI Workshop in System Identification (Sigtuna, Sweden, Oct. 1–3, 2008). oral presentation.

— (Oct. 2007). "Modeling input errors in least squares support vector regression". At: 16th ERNSI Workshop in System Identification (San Servolo, Italy, Oct. 1, 2007–Oct. 3, 2011). poster presentation.