Robustness Analysis for Least Squares Kernel Based Regression: an Optimization Approach

Tillmann Falck, Johan A. K. Suykens and Bart De Moor

***Abstract*—In kernel based regression techniques (such as Support Vector Machines or Least Squares Support Vector Machines) it is hard to analyze the influence of perturbed inputs on the estimates. We show that for a nonlinear black box model a convex problem can be derived if it is linearized with respect to the influence of input perturbations. For this model an explicit prediction equation can be found. The cast into a convex problem is possible as we assume that the perturbations are bounded by a design parameter ϱ. The problem requires the solution of linear systems in Nd (the number of training points times the input dimensionality) variables. However, approximate solutions can be obtained with moderate computational effort. We demonstrate on simple examples that possible applications are in robust model selection, experiment design and model analysis.**

I. Introduction

The estimation of a black-box model in order to make precise forecasts starting from a set of observations is a common practice in system identification [1]. In nonlinear system identification [2], [3] kernel based estimation techniques, like Support Vector Machines (SVMs) [4], Least Squares Support Vector Machines (LS-SVMs) [5], [6] or Splines [7] have shown to be capable nonlinear black-box regression methods. The natural model structure for many of these methods is an autoregressive model. Extensions for LS-SVMs to output-error (OE) models or partially linear models are given in [8] and [9] respectively. The latter can still be formulated as a convex problem while the former corresponds to a nonconvex recurrent formulation.

In this work we are going to consider the case that the regressors are subject to an unknown but bounded perturbation and analyze the effect on the resulting predictive model. Except from boundedness we impose no further restrictions on this perturbation. For linear parametric models [10], [11] this corresponds to perturbed measurement matrices and is analyzed in the setting of robust estimation and related to classical regularization schemes for LS and to Total Least Squares (TLS) [12]. Iterative schemes for parametric nonlinear systems first appeared in [13] while [14] and [15] are able to pose them as SOCP problems. For linear in the parameters models the results are exact except for a first order approximation of the nonlinear basis functions which will be used in a similar fashion later in this work.

Nonparametric and nonlinear models can be estimated using Support Vector techniques by solving convex problems. In particular the solution of LS-SVMs is given by a system of linear equations. They solve a parametric linear regression problem in a high dimensional feature space. Alternatively a nonparametric dual problem with dimensionality proportional to the number of data can be solved. This dual relies on a chosen positive definite kernel function implicitly defining the feature space. Robust solutions for SVMs in terms of SOCP problems are given in [16] in a probabilistic setting while [17] looks at the deterministic case. A result for the related TLS problem in a LS-SVM context can be found in [18].

In this paper we derive a deterministic model explicitely incorporating the robust assumption using first order derivative information in the prediction equations for out of sample extensions. Furthermore the presented method is computationally much more attractive than schemes based on SOCP problems as it relies on solving a linear system.

This paper is organized as follows. In Section II the problem setting is defined and a convex formulation is derived. The solution for the robust estimation problem is obtained in Section III by solving a linear sytem of equations in its dual domain. Section IV discusses numerical issues and proposes an approximation scheme. Numerical experiments and simulations are carried out in Section V. Finally the conclusions and an outlook are given in VI.

II. Problem Statement & Assumptions

A. Problem Setting

Based on a set of measurements $\{\boldsymbol{x}_k, y_k\}_{k=1}^N$ where the inputs $\boldsymbol{x}_k = \boldsymbol{x}_k^0 + \boldsymbol{\delta}_k \in \mathbb{R}^d$ are corrupted by unstructured perturbations $\boldsymbol{\delta}_k$ and the outputs $y_k = y_k^0 + e_k \in \mathbb{R}$ are subject to additive noise e_k with bounded variance $\mathrm{Var}[e_k] < \infty$ a nonlinear model should be estimated. We will consider problems for which the perturbations are bounded $\|\boldsymbol{\delta}_k\|_2 \leq \varrho$ for all k and ϱ is a given value. The form of the estimator $\hat{y} : \mathbb{R}^d \to \mathbb{R}$ is given by

$$\hat{y}(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}) + b \tag{1}$$

This work has been supported by Research Council KUL: GOA AM-BioRICS, CoE EF/05/006 Optimization in Engineering(OPTEC), IOF-SCORES4CHEM, several PhD/postdoc & fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects G.0452.04 (new quantum algorithms), G.0499.04 (Statistics), G.0211.05 (Nonlinear), G.0226.06 (cooperative systems and optimization), G.0321.06 (Tensors), G.0302.07 (SVM/Kernel), G.0320.08 (convex MPC), G.0558.08 (Robust MHE), G.0557.08 (Glycemia2), G.0588.09 (Brain-machine) research communities (ICCoS, ANMMM, MLDM); IWT: PhD Grants, McKnow-E, Eureka-Flite+ Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011) ; EU: ERNSI; FP7-HD-MPC (Collaborative Project STREP-grantnr. 223854) Contract Research: AMINAL Other: Helmholtz: viCERP

T. Falck, J.A.K. Suykens and B. De Moor, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Leuven (Heverlee), Belgium {tillmann.falck@esat.kuleuven.be, johan.suykens@esat.kuleuven.be}, J.A.K. Suykens is a professor and B. De Moor is a full professor at K.U. Leuven.

with the model parameters $\boldsymbol{w} \in \mathbb{R}^{n_h}$ and $b \in \mathbb{R}$ and a given mapping $\boldsymbol{\varphi} : \mathbb{R}^d \to \mathbb{R}^{n_h}$. In kernel based techniques the mapping $\boldsymbol{\varphi}$ is called feature map and implicitly defined by a positive (semi) definite kernel function $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$. According to Mercer's theorem [4], [19] for a positive definite kernel $K(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\varphi}(\boldsymbol{x})^T \boldsymbol{\varphi}(\boldsymbol{y})$. The dimension of $\boldsymbol{\varphi}$ is usually very large $n_h \gg d$. For the commonly used Gaussian RBF kernel $K(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\|\boldsymbol{x} - \boldsymbol{y}\|_2^2 / \sigma^2)$ it is infinite dimensional whereas for the polynomial kernel $K(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x}^T \boldsymbol{y} + c_1)^{c_2}$ with $c_1 > 0$ and $c_2 \in \mathbb{N}$ has a finite dimensional feature map with $n_h = \binom{d+c_2}{d}$.

The estimation objective is to fit the model in (1) to the perturbed data as formalized in the following optimization problem

$$\min_{\boldsymbol{w}, b, e_k, \boldsymbol{\delta}_k} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{k=1}^{N} e_k^2 \tag{2a}$$

$$\text{subject to} \quad y_k = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_k - \boldsymbol{\delta}_k) + b + e_k, \quad k = 1, \ldots, N. \tag{2b}$$

Solving this problem is difficult as the constraints in (2b) are nonlinear in $\boldsymbol{\delta}_k$ and possibly nonconvex. If the $\boldsymbol{x}_k$ are unperturbed $\boldsymbol{\delta}_k = \boldsymbol{0}$ this problem corresponds to the Least Squares Support Vector Machine (LS-SVMs) regression [5].

B. 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). As common in machine learning techniques those parameters have to be selected in some optimal way. We consider the perturbation level ϱ as a user defined variable that is given and has not to be selected. For choosing γ and ϱ we will use well established methods like selecting those parameters according to the prediction performance given by cross validation [20], which work well for LS-SVMs [5], [6].

C. Linearization

To make the problem tractable it has to be simplified. In the scope of this paper we are looking for a related convex problem. Therefore we linearize the feature map $\boldsymbol{\varphi}$ with respect to perturbations $\boldsymbol{\delta}$ on its argument $\boldsymbol{x}$

$$\boldsymbol{\varphi}(\boldsymbol{x} - \boldsymbol{\delta}) \simeq \boldsymbol{\varphi}(\boldsymbol{x}) - \sum_{i=1}^{d} (\partial_{x_i} \boldsymbol{\varphi})(\boldsymbol{x}) \delta_i \tag{3}$$

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 $\boldsymbol{\Phi}'_k = [(\partial_{x_1}\boldsymbol{\varphi})(\boldsymbol{x}_k), \ldots, (\partial_{x_d}\boldsymbol{\varphi})(\boldsymbol{x}_k)] \in \mathbb{R}^{n_h \times d}$. Then the modeling constraint (2b) can be compactly approximated as

$$y_k \simeq \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_k) - \boldsymbol{w}^T \boldsymbol{\Phi}'_k \boldsymbol{\delta}_k + b + e_k. \tag{4}$$

The term $\boldsymbol{w}^T \boldsymbol{\Phi}'_k \boldsymbol{\delta}_k$ is bilinear in the unknowns $\boldsymbol{w}$ and $\boldsymbol{\delta}_k$ and thus the corresponding optimization problem is still non convex.

D. Convexification

In literature for robust solutions of linear systems [10] similar problems are solved by robustifying with respect to bounded perturbations $\|\boldsymbol{\delta}_k\|_2 \leq \varrho$. For the sake of this argument we will consider a related l_2 instead of the least squares (LS) problem (2). Then the following lemma based on [10, Theorem 3.1] states a convex Second Order Cone Programming (SOCP) problem.

Lemma 1 (Robust SOCP): For the worst case scenario over all $\|\boldsymbol{\delta}_k\| \leq \varrho$ the l_2 formulation of Problem (2) with the linearized modeling constraint (4) is given by

$$\min_{\boldsymbol{w}, b, e_k} \quad \max_{\|\boldsymbol{\delta}_k\|_2 \leq \varrho} \quad \|\boldsymbol{w}\|_2 + \gamma \|\boldsymbol{e}\|_2 \tag{5a}$$

$$\text{subject to} \quad y_k = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_k) - \boldsymbol{w}^T \boldsymbol{\Phi}'_k \boldsymbol{\delta}_k + b + e_k, \quad k = 1, \ldots, N. \tag{5b}$$

The solution of this worst case approximation is equivalent to adding an additional regularization term to the objective function

$$\min_{\boldsymbol{w}, b, e_k} \quad \|\boldsymbol{w}\|_2 + \gamma \|\boldsymbol{e}\|_2 + \gamma\varrho \left\| \boldsymbol{\Phi}'^T \boldsymbol{w} \right\|_2 \tag{6a}$$

$$\text{subject to} \quad y_k = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_k) + b + e_k, \quad k = 1, \ldots, N. \tag{6b}$$

Proof: Consider

$$\max_{\|\boldsymbol{\delta}_k\|_2 \leq \varrho} \|\boldsymbol{e}\|_2 \text{ s.t. } e_k = y_k - \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_k) + \boldsymbol{w}^T \boldsymbol{\Phi}'_k \boldsymbol{\delta}_k - b \tag{7}$$

and define the standard unit vectors $\{\boldsymbol{\epsilon}_k\}_{k=1}^N$ and the matrices $\boldsymbol{\Phi} = [\boldsymbol{\varphi}(\boldsymbol{x}_1), \ldots, \boldsymbol{\varphi}(\boldsymbol{x}_N)]$ and $\boldsymbol{\Phi}' = [\boldsymbol{\Phi}'_1, \ldots, \boldsymbol{\Phi}'_N]$. Then the maximum in (7) can be explicitly computed. First split off contributions independent of $\boldsymbol{\delta}_k$: $\max_{\|\boldsymbol{\delta}_k\|_2 \leq \varrho} \|\boldsymbol{e}\|_2 = \max_{\|\boldsymbol{\delta}_k\|_2 \leq \varrho} \left\| \boldsymbol{y} - \boldsymbol{\Phi}^T \boldsymbol{w} - b\boldsymbol{1} + \sum_{k=1}^N \left(\boldsymbol{w}^T \boldsymbol{\Phi}'_k \boldsymbol{\delta}_k\right) \boldsymbol{\epsilon}_k \right\|_2 \leq \left\| \boldsymbol{y} - \boldsymbol{\Phi}^T \boldsymbol{w} - b\boldsymbol{1} \right\|_2 + \max_{\|\boldsymbol{\delta}_k\|_2 \leq \varrho} \left\| \sum_{k=1}^N \left(\boldsymbol{w}^T \boldsymbol{\Phi}'_k \boldsymbol{\delta}_k\right) \boldsymbol{\epsilon}_k \right\|_2$. Note that the sum over $\boldsymbol{\epsilon}_k$ is just forming a vector element by element, then the remainder can be simplified as follows $\max_{\|\boldsymbol{\delta}_k\|_2 \leq \varrho} \left\| \sum_{k=1}^N \left(\boldsymbol{w}^T \boldsymbol{\Phi}'_k \boldsymbol{\delta}_k\right) \boldsymbol{\epsilon}_k \right\|_2^2 = \max_{\|\boldsymbol{\delta}_k\|_2 \leq \varrho} \sum_{k=1}^N \left(\boldsymbol{w}^T \boldsymbol{\Phi}'_k \boldsymbol{\delta}_k\right)^2 \leq \max_{\|\boldsymbol{\delta}_k\|_2 \leq \varrho} \sum_{k=1}^N \left\|\boldsymbol{w}^T \boldsymbol{\Phi}'_k\right\|_2^2 \|\boldsymbol{\delta}_k\|_2^2 \leq \varrho^2 \sum_{k=1}^N \left\|\boldsymbol{w}^T \boldsymbol{\Phi}'_k\right\|_2^2 = \varrho^2 \left\| \boldsymbol{\Phi}'^T \boldsymbol{w} \right\|_2^2$. Thus an upper bound for (7) is given by $\|\boldsymbol{e}\|_2 + \varrho \left\| \boldsymbol{\Phi}'^T \boldsymbol{w} \right\|_2$ s.t. $e_k = y_k - \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_k) - b$. Substitution shows that this upper bound is exact for $\boldsymbol{\delta}_k = \varrho \operatorname{sign}(\boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_k) + b - y_k) \left(\boldsymbol{w}^T \boldsymbol{\Phi}'_k\right) / \left\|\boldsymbol{w}^T \boldsymbol{\Phi}'_k\right\|_2$ which concludes the proof. ∎

Remark 1: It is straightforward to generalize this technique to incorporate simple prior knowledge on the perturbations $\boldsymbol{\delta}_k$. If known a priori that only some components of $\boldsymbol{x}_k$ are perturbed i.e. that $\boldsymbol{\delta}_k$ is sparse, one should modify the approximation in (3) to only consider 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 $\boldsymbol{x}_k$. If known a priori that some components are much more precise than others define a non singular matrix $\boldsymbol{D}$ and maximize Problem (5) with

respect to $\|\boldsymbol{D}\boldsymbol{\delta}_k\|_2 \leq \varrho$ instead of the unweighted norm. This is equivalent to solving the original problem with modified derivative information $\overline{\boldsymbol{\Phi}}'_k = \boldsymbol{\Phi}'_k \boldsymbol{D}^{-1}$.

III. Least Squares Kernel Based Model

A. Problem Statement & Solution

The SOCP in (6) gives a robust convex approximation to the original problem in (2). 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 (6) 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 2: For

$$\gamma' = \frac{\|\boldsymbol{w}\|_2}{\|\boldsymbol{e}\|_2}\gamma \quad \text{and} \quad \varrho' = \frac{\|\boldsymbol{e}\|_2}{\left\|\boldsymbol{\Phi}'^T \boldsymbol{w}\right\|_2}\varrho \tag{8}$$

the solution of the SOCP problem (6) coincides with the solution of the following LS-SVM problem

$$\min_{\boldsymbol{w},b,e_k} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma'\varrho'\boldsymbol{w}^T\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T\boldsymbol{w} + \frac{1}{2}\gamma'\sum_{k=1}^{N} e_k^2 \tag{9a}$$

subject to

$$y_k = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_k) + b + e_k, \qquad k = 1,\ldots,N. \tag{9b}$$

Proof: Substituting (8) into (9a) yields (5a). ■

Making use of Mercer's Theorem the derivatives of the feature map can be written as derivatives of the kernel function as follows $(\partial_{x_i} K)(\boldsymbol{x},\boldsymbol{y}) = (\partial_{x_i}\boldsymbol{\varphi})(\boldsymbol{x})^T\boldsymbol{\varphi}(\boldsymbol{y})$ and $(\partial_{x_i}\partial_{y_j} K)(\boldsymbol{x},\boldsymbol{y}) = (\partial_{x_i}\boldsymbol{\varphi})(\boldsymbol{x})^T(\partial_{x_j}\boldsymbol{\varphi})(\boldsymbol{y})$ also define $\boldsymbol{\Omega} = \boldsymbol{\Phi}^T\boldsymbol{\Phi}$, $\boldsymbol{\Omega}_x = \boldsymbol{\Phi}'^T\boldsymbol{\Phi}$ and $\boldsymbol{\Omega}_{xy} = \boldsymbol{\Phi}'^T\boldsymbol{\Phi}'^T$. All of these matrices can then be efficiently computed using the kernel function $(\boldsymbol{\Omega})_{kl} = K(\boldsymbol{x}_k,\boldsymbol{x}_l)$. $\boldsymbol{\Omega}_x \in \mathbb{R}^{(Nd)\times N}$ has block structure with blocks defined by $(\boldsymbol{\Omega}_x)_{kl} = \boldsymbol{\omega}_x^{kl} \in \mathbb{R}^{d\times 1}$ where $(\boldsymbol{\omega}_x^{kl})_i = (\partial x_i K)(\boldsymbol{x}_k,\boldsymbol{x}_l)$. $\boldsymbol{\Omega}_{xy} \in \mathbb{R}^{(Nd)\times(Nd)}$ is block structured also and its blocks are defined by $(\boldsymbol{\Omega}_{xy})_{kl} = \boldsymbol{\Omega}_{xy}^{kl} \in \mathbb{R}^{d\times d}$ where $(\boldsymbol{\Omega}_{xl}^{kl})_{ij} = (\partial x_i \partial y_j K)(\boldsymbol{x}_k,\boldsymbol{x}_l)$. Using these definitions the solution for (9) is stated in the following lemma.

Lemma 3: For the estimation problem described by (9) the solution is given in the dual by

$$\begin{bmatrix} \boldsymbol{\Omega}' + \boldsymbol{I}_N/\gamma' & \boldsymbol{1} \\ \boldsymbol{1}^T & 0 \end{bmatrix}\begin{bmatrix}\boldsymbol{\alpha}\\ b\end{bmatrix} = \begin{bmatrix}\boldsymbol{y}\\ 0\end{bmatrix} \tag{10}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^N$ are the dual variables. $\boldsymbol{\Omega}'$ is positive semi definite and defined as

$$\boldsymbol{\Omega}' = \boldsymbol{\Omega} - \boldsymbol{\Omega}_x^T\left(\boldsymbol{\Omega}_{xy} + (\gamma'\varrho')^{-1}\boldsymbol{I}_{(Nd)}\right)^{-1}\boldsymbol{\Omega}_x. \tag{11}$$

Proof: The Lagrangian for (9) is $\mathscr{L}(\boldsymbol{w},b,e_k,\alpha_k) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma'\varrho'\boldsymbol{w}^T\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T\boldsymbol{w} + \frac{1}{2}\gamma'\sum_{k=1}^N e_k^2 - \sum_{k=1}^N \alpha_k\left(\boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_k) + b + e_k - y_k\right)$ and the corresponding KKT conditions for optimality are $\nabla_{\boldsymbol{w}}\mathscr{L} = \boldsymbol{0} : \boldsymbol{\Phi}\boldsymbol{\alpha} = \left(\boldsymbol{I}_{n_h} + \gamma'\varrho'\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T\right)\boldsymbol{w}$, $\partial_b\mathscr{L} = 0 : \boldsymbol{1}^T\boldsymbol{\alpha} = 0$, $\partial_{e_k}\mathscr{L} = 0 : \gamma' e_k = \alpha_k$ and $\partial_{\alpha_k}\mathscr{L} = 0 : \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_k) + b + e_k = y_k$ for $k = 1,\ldots,N$ The combination of $\nabla_{\boldsymbol{w}}\mathscr{L} = \boldsymbol{0}$, $\partial_{e_k}\mathscr{L} = 0$ and $\partial_{\alpha_k}\mathscr{L} = 0$ yields

$$\boldsymbol{y} = \boldsymbol{\Phi}^T\left(\gamma'\varrho'\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T + \boldsymbol{I}_{n_h}\right)^{-1}\boldsymbol{\Phi}\boldsymbol{\alpha} + \frac{1}{\gamma'}\boldsymbol{\alpha} + b\boldsymbol{1}. \tag{12}$$

The modified kernel matrix (11) 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 semi definite, note that $\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T$ is positive semi definite (psd) as $\boldsymbol{\Omega}'' = \boldsymbol{\Phi}'^T\boldsymbol{\Phi}'$ 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 $\boldsymbol{x} \in \mathbb{R}^{n_h}$ the following holds $\boldsymbol{x}^T\boldsymbol{\Omega}'\boldsymbol{x} = \boldsymbol{x}^T\boldsymbol{\Phi}^T\left(\gamma'\varrho'\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T + \boldsymbol{I}_{n_h}\right)^{-1}\boldsymbol{\Phi}\boldsymbol{x} = \boldsymbol{z}^T\left(\gamma'\varrho'\boldsymbol{\Phi}'\boldsymbol{\Phi}'^T + \boldsymbol{I}_{n_h}\right)^{-1}\boldsymbol{z} \geq 0$ Finally the linear system (10) follows from the combination of (12) and $\partial_b\mathscr{L} = 0$. ■

Remark 2: It is possible to express the relations between the regularization constants in (8) 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.

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 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 $\|\boldsymbol{\delta}_k\|_2 \leq \varrho$.

B. Predictive Model

For out of sample extensions a prediction equation has to be derived. Therefore the value for $\boldsymbol{w}$ obtained from $\nabla_{\boldsymbol{w}}\mathscr{L} = \boldsymbol{0}$ is substituted into the prediction model (1). For a new point $\boldsymbol{z}$ this yields

$$\hat{y}(\boldsymbol{z}) = \sum_{k=1}^{N}\alpha_k K(\boldsymbol{x}_k,\boldsymbol{z}) + b - \boldsymbol{k}_x^T(\boldsymbol{z})\left(\boldsymbol{\Omega}_{xy} + (\gamma'\varrho')^{-1}\boldsymbol{I}_{(Nd)}\right)^{-1}\boldsymbol{\Omega}_x\boldsymbol{\alpha} \tag{13}$$

where $\boldsymbol{k}_x : \mathbb{R}^d \to \mathbb{R}^{Nd}$ is defined as $\boldsymbol{k}_x(\boldsymbol{z}) = [(\partial_{x_1}K)(\boldsymbol{x}_1,\boldsymbol{z}),\ldots,(\partial_{x_d}K)(\boldsymbol{x}_N,\boldsymbol{z})]^T$.

IV. Numerical Implementation

Although the computational burden of solving the SOCP problem (6) has been greatly reduced to solving a linear system (10), the solution still requires the solution of a very large linear system to compute the modified gram matrix (11). To be able to solve a moderately sized problem with for example $N = 1000$ data points and $d = 10$ dimensions a linear systems in $10,000^2$ variables have to be solved. Storage of the system as well as solving it are expensive. Therefore ways to approximate the solution of (11) 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 semi-definite. In the literature several approximation techniques have been proposed for kernel based learning settings like the Nyström approximation [21] or the incomplete Cholesky decomposition [22]. In the following we will use the Nyström approximation.

A. Nyström Approximation

The Nyström approximation is a subsampling technique that selects a subset of the training set $\{\boldsymbol{x}_k\}_{k=1}^N$ and uses that subset to approximate an eigenvalue decomposition of $\boldsymbol{\Omega}_{xy}$. Based on s points $\boldsymbol{\Omega}_{xy}^{(s)} = \boldsymbol{U}^{(s)}\boldsymbol{V}^{(s)}{\boldsymbol{U}^{(s)}}^T$ is the Nyström approximation of $\boldsymbol{\Omega}_{xy}$ where $\boldsymbol{V}^{(s)}$ is diagonal. The more data is used for the approximation the closer the matrix $\boldsymbol{U}^{(s)}$ will get to an orthogonal matrix. Depending on the process generating the set $\{\boldsymbol{x}_k\}_{k=1}^N$ it might be sufficient to just draw a random subsample. In other cases the approximation performance can be improved by optimizing the selected set according to some performance criterion for example the Renyi entropy [23], [5].

B. Optimizations

Given an (approximate) factorization $\boldsymbol{\Omega}_{xy} \simeq \boldsymbol{F}\boldsymbol{D}\boldsymbol{F}^T$ with $\boldsymbol{D}$ diagonal and of size $a \times a$ and $\boldsymbol{F}$ of size $(Nd) \times a$ one can simplify the computation in (11) 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 (11) has to be computed for many different values of $\gamma'\varrho'$. In that case we suggest to transform the factorization into a real eigenvalue decomposition. Therefore orthogonalize the matrix $\boldsymbol{F}$ by means of a QR factorization $\boldsymbol{F} = \boldsymbol{Q}\boldsymbol{R}$ and compute the eigenvalue decomposition of $\boldsymbol{R}\boldsymbol{D}\boldsymbol{R}^T = \tilde{\boldsymbol{U}}\tilde{\boldsymbol{V}}\tilde{\boldsymbol{U}}^T$. Then define a new factorization as $\boldsymbol{\Omega}_{xy} = \boldsymbol{U}\boldsymbol{V}\boldsymbol{U}^T$ with $\boldsymbol{V} = \tilde{\boldsymbol{V}}$ and $\boldsymbol{U} = \boldsymbol{Q}\tilde{\boldsymbol{U}}$. Using this factorization the expression for (11) 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.

V. Numerical Experiments

The simulations are performed on a Intel Core 2 Quad with 2.66 GHz and 8 GB memory running RHEL5 (Linux, 64 bit). All simulations are limited to a single core and are implemented using Python 2.5.2 [24] and Numpy 1.2 [25].

Let $\{x_k\}_{k=1}^{N+d-1}$ be independent draws from a normal distribution $\mathcal{N}(0,1)$. Then define regressors $\boldsymbol{x}_k = [x_k, \ldots, x_{k+d-1}]^T$ 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 V-E and V-F a fixed bandwidth $\sigma = \sqrt{d}$ is used.

In the next three sections the following setting is considered for the experiments. Given a LS-SVM based model (corresponding to $\varrho' = 0$) what is the potential effect on the model if we increase ϱ. The procedure is outlined in Algorithm 1. For the examples 3-fold cross validation has been used. The considered models have NFIR or NARX structure. For the latter the elements of the training set $\mathbb{T}$ are of course extended with the corresponding lagged outputs.

Algorithm 1 Experimental procedure for Section V

1) estimate a LS-SVM model: solve (10) with $\varrho' = 0$
   - select model with best prediction performance on the training set based on cross validation
   - yields a regularization parameter γ' and a bandwidth σ
2) translate γ' to γ using (8) 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 (8), see Remark 2 and Section V-D
   b) solve (10)
   c) compute predictions using (13)

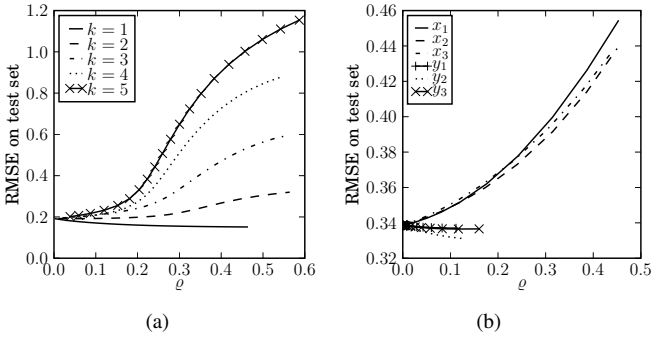

Fig. 1. Sensitivity of the prediction performance of models (a) and (b) in Section V-A with respect to their input variables.

A. 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 give support where to concentrate the effort to acquire good data or for variable selection. This is done by considering perturbations only on a single input at a time using Remark 1. 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 1(a). In Panel 1(b) the dynamical model (b) is analyzed. It can indeed be seen that the model performance depends non uniformly on the used variables, the nonlinear variables have a much higher weight than the linear ones.

B. Sensitivity of kernels

In model selection one often selects the model achieving the best prediction performance. In case one has several mod-

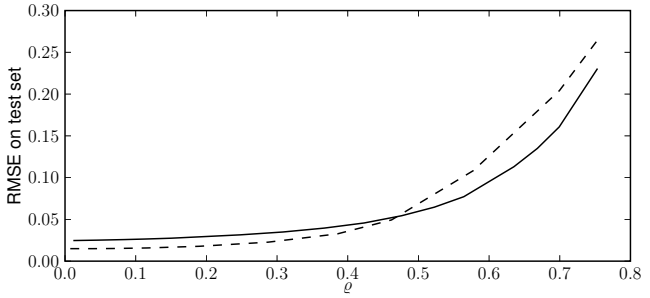

Fig. 2. Sensitivity of the NARX model in Section V-B with respect to chosen kernel function. The dashed and the solid lines mark the polynomial kernel of order 4 ($c_2 = 4$) and the Gaussian RBF respectively.

els achieving a similar performance one might consider trading of performance versus robustness. Therefore scarifying a bit of performance one might gain an improvement in 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 so we compare the performances of the polynomial kernel with a Gaussian RBF kernel. Figure 2 shows that the best model with a polynomial kernel outperforms the RBF based model, yet the model employing on the Gaussian is more robust to perturbation. Depending on the application one might select the one or the other.

C. 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 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 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 we do not know the precise location of $\boldsymbol{x}_k$ the estimate will not be affected.

D. 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 (5) and the LS problem (9) as given by (8) is only explicit once a solution has been computed. Therefore it is only possible to choose a pair ϱ', γ', solve (9) and then map these parameters onto γ and ϱ. An example is shown in Figure 4(a).

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' have to be considered significant therefore it is necessary to search for a pair of

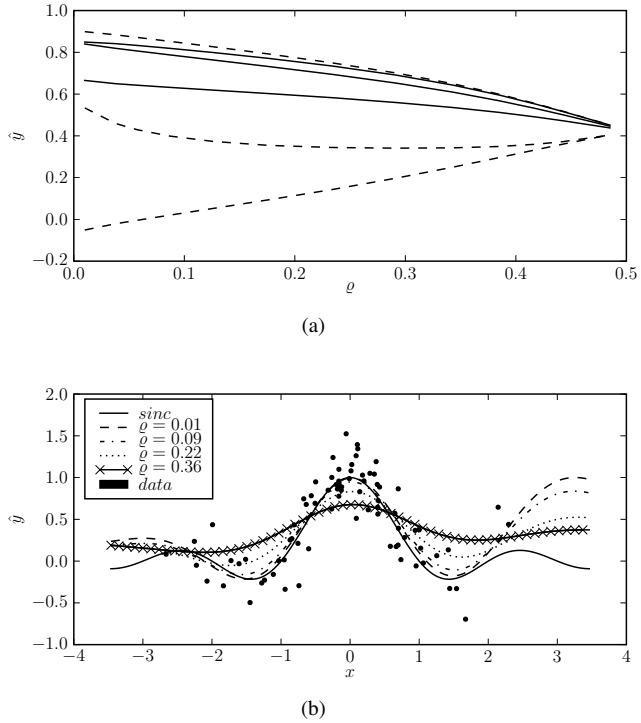

Fig. 3. Examples for sensitivity of pointwise predictions $\hat{y}(x)$ of a simple one-dimensional sinc for $N = 80$ training data. In the left panel shown as a function of ϱ for specific points x where the dashed lines are for out of sample predictions whereas the solid lines represent predictions for training samples. The right panel shows the predicted function for different values of ϱ.

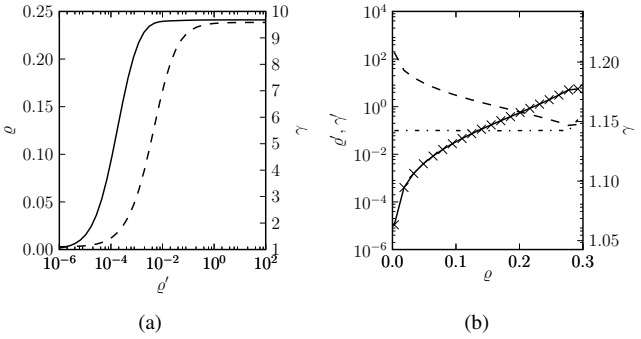

Fig. 4. Relation of regularization parameters γ (dashed line), ϱ (solid line in (a)) and γ' (dashed dotted line) and ϱ' (solid line in (b)). Panel (a) shows the relation as a function of ϱ' with γ' fixed, whereas in Figure (b) ϱ is varied for γ as constant as possible. The analyzed function is $y(\boldsymbol{x}) = \sum_{i=1}^{d} \mathrm{sinc}(x_i)$, $N = 500$.

γ', ϱ' that result in a constant γ and a desired ϱ. This nonlinear optimization problem in two variables is quite sensitive. Figure 4(b) 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 get numerically unstable.

E. Approximation Performance of $\boldsymbol{\Omega}_{xy}$

The approximation performance $\mathrm{PERF}(\boldsymbol{\Omega}_{xy}, s) = 100\ \|\boldsymbol{\Omega}_{xy} - \boldsymbol{\Omega}_{xy}^{(s)}\|_F / \|\boldsymbol{\Omega}_{xy}\|_F$ of $\boldsymbol{\Omega}_{xy}$ as achieved by the

TABLE I
PERFORMANCE OF NYSTRÖM APPROXIMATION $\mathbf{\Omega}_{xy}^{(s)}$ FOR $N = 1000$ AND $d = 10$ AND DIFFERENT SUBSAMPLE SIZES s

s	$\mathrm{PERF}(\mathbf{\Omega}_{xy}, s)$	$\frac{\left\|\boldsymbol{U}^{(s)^T}\boldsymbol{U}^{(s)}\right\|_F}{\sqrt{s \cdot d}}$	runtime
100	7.67 (0.67)	0.90 (0.01)	4.0 s
250	2.92 (0.30)	0.88 (0.01)	36.2 s
500	1.16 (0.12)	0.99 (0.08)	205.6 s
750	0.56 (0.15)	1.06 (0.12)	621.0 s

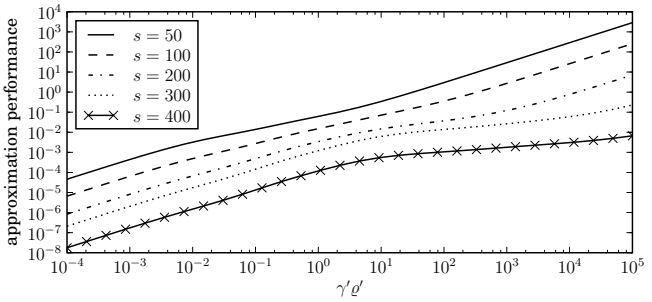

Fig. 5. Composite approximation performance for $N = 500$ and $d = 5$ of $\mathbf{\Omega}_x^T(\mathbf{\Omega}_{xy} + (\gamma'\varrho')^{-1}\boldsymbol{I}_{Nd})^{-1}\mathbf{\Omega}_x$.

Nyström approximation using a randomly chosen subset $\mathbb{S} \subset \mathbb{T}$ with $s = |\mathbb{S}|$ is reported in Table I for different approximation dimensions $a = s \cdot d$. 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. As Nyström approximation is based on an eigenvalue decomposition based on a matrix for the subset $\mathbb{S}$ which is of dimension $(s \cdot d)^2$ it is clear that the approximation dimensions may not be too large as otherwise the computation time gets increasingly long.

F. Composite Approximation Performance

The general setting is identical to before but now we are interested in the approximation performance of $\mathbf{\Omega}_x^T(\mathbf{\Omega}_{xy} + (\gamma'\varrho')^{-1}\boldsymbol{I}_{Nd})^{-1}\mathbf{\Omega}_x$ as a function of $\gamma'\varrho'$. The results are illustrated in Figure 5 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 $\gamma'\varrho'$ is 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.

VI. CONCLUSIONS & OUTLOOK

Based on the assumption of bounded unstructured perturbations on the inputs, we have derived a scheme for robustness analysis of nonlinear black box models. The estimation problem is convex and can be recast from a SOCP problem into a linear system. For this large linear system an approximate solution scheme has been outlined. On simple examples we have shown that this setting is capable of analysing existing models based on the LS-SVM estimator. Potential applications in variable and model selection can be considered.

REFERENCES

[1] L. Ljung, *System identification: Theory for the User*. Prentice Hall PTR Upper Saddle River, NJ, USA, 1999.

[2] J. Sjoberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.-Y. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modeling in system identification: a unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, Dec. 1995.

[3] A. Juditsky, H. Hjalmarsson, A. Benveniste, B. Delyon, L. Ljung, J. Sjoberg, and Q. Zhang, "Nonlinear black-box models in system identification: Mathematical foundations," *Automatica*, vol. 31, no. 12, pp. 1725–1750, Dec. 1995.

[4] V. Vapnik, *Statistical Learning Theory*. John Wiley & Sons, 1998.

[5] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. World Scientific, 2002.

[6] M. Espinoza, J. A. K. Suykens, R. Belmans, and B. De Moor, "Electric load forecasting - using kernel based modeling for nonlinear system identification," *IEEE Control Systems Magazine*, vol. 27, no. 5, pp. 43–57, Oct. 2007.

[7] G. Wahba, *Spline Models for Observational Data*. SIAM, 1990.

[8] J. A. K. Suykens and J. Vandewalle, "Recurrent least squares support vector machines," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 47, no. 7, pp. 1109–1114, 2000.

[9] M. Espinoza, J. A. K. Suykens, and B. De Moor, "Kernel based partially linear models and nonlinear identification," *IEEE Trans. Autom. Control*, vol. 50, no. 10, pp. 1602–1606, Oct. 2005.

[10] L. El Ghaoui and H. Lebret, "Robust solutions to least-squares problems with uncertain data," *SIAM J. Matrix Anal. Appl.*, vol. 18, no. 4, pp. 1035–1064, 1997.

[11] S. Chandrasekaran, G. H. Golub, M. Gu, and A. H. Sayed, "An efficient algorithm for a bounded errors-in-variables model," *SIAM J. Matrix Anal. Appl.*, vol. 20, no. 4, pp. 839–859, 1999.

[12] S. Van Huffel and J. Vandewalle, *The Total Least Squares Problem : Computational Aspects and Analysis, Frontiers in Applied Mathematics Series, Vol. 9*. SIAM, Philadelphia, 1991.

[13] J. B. Rosen, H. Park, and J. Glick, "Structured total least norm for nonlinear problems," *SIAM J. Matrix Anal. Appl.* , vol. 20, no. 1, pp. 14–30, Sept. 1998.

[14] G. A. Watson, "Robust solutions to a general class of approximation problems," *SIAM J. Sci. Comput.*, vol. 25, no. 4, pp. 1448–1460, 2003.

[15] ——, "Robust counterparts of errors-in-variables problems," *Computational Statistics and Data Analysis*, pp. 1080–1089, 2007.

[16] P. K. Shivaswamy, C. Bhattacharyya, and A. J. Smola, "Second order cone programming approaches for handling missing and uncertain data," *JMLR*, vol. 7, pp. 1283–1314, 2006.

[17] T. B. Trafalis and R. C. Gilbert, "Robust classification and regression using support vector machines," *European Journal of Operational Research*, vol. 173, no. 3, pp. 893–909, Sept. 2006.

[18] R. A. Renault, H. Guo, and W. J. Chen, "Regularised total least squares support vector machines," Presentation, May 2005. [Online]. Available: http://math.asu.edu/ rosie/mypresentations/Rosie_talk_svmc.pdf

[19] B. Schölkopf and A. Smola, *Learning with Kernels*. MIT Press Cambridge, Mass, 2002.

[20] D. J. C. MacKay, "Comparison of approximate methods for handling hyperparameters," *Neural Computation*, vol. 11, no. 5, pp. 1035–1068, 1999.

[21] C. K. I. Williams and M. Seeger, "Using the Nyström method to speed up kernel machines," in *Neural Information Processing Systems 13*, T. Leen, T. Dietterich, and V. Tresp, Eds. MIT Press, 2001, pp. 682–688.

[22] S. Fine and K. Scheinberg, "Efficient svm training using low-rank kernel representations," *JMLR*, vol. 2, pp. 243–264, 2002.

[23] M. Girolami, "Orthogonal series density estimation and the kernel eigenvalue problem," *Neural Computation*, vol. 14, no. 3, pp. 669–688, Mar. 2002.

[24] G. van Rossum *et al.*, *Python Language Website*. [Online]. Available: http://www.python.org/

[25] E. Jones, T. Oliphant, P. Peterson, *et al.*, *SciPy: Open source scientific tools for Python*, 2001–. [Online]. Available: http://www.scipy.org/