Componentwise Least Squares Support Vector Machines

K. Pelckmans[1], I. Goethals[1], J. De Brabanter[1,2], J.A.K. Suykens[1], and B. De Moor[1]

[1] KULeuven - ESAT - SCD/SISTA,
Kasteelpark Arenberg 10,
3001 Leuven (Heverlee), Belgium,
{Kristiaan.Pelckmans,Johan.Suykens}@esat.kuleuven.ac.be
[2] Hogeschool KaHo Sint-Lieven (Associatie KULeuven), Departement Industrieel Ingenieur

Summary. This chapter describes componentwise Least Squares Support Vector Machines (LS-SVMs) for the estimation of additive models consisting of a sum of nonlinear components. The primal-dual derivations characterizing LS-SVMs for the estimation of the additive model result in a single set of linear equations with size growing in the number of data-points. The derivation is elaborated for the classification as well as the regression case. Furthermore, different techniques are proposed to discover structure in the data by looking for sparse components in the model based on dedicated regularization schemes on the one hand and fusion of the componentwise LS-SVMs training with a validation criterion on the other hand.

keywords: LS-SVMs, additive models, regularization, structure detection

1 Introduction

Non-linear classification and function approximation is an important topic of interest with continuously growing research areas. Estimation techniques based on regularization and kernel methods play an important role. We mention in this context smoothing splines (Wahba, 1990), regularization networks (Poggio and Girosi, 1990), Gaussian processes (MacKay, 1992), Support Vector Machines (SVMs) (Vapnik, 1998; Cristianini and Shawe-Taylor, 2000; Schoelkopf and Smola, 2002) and many more, see e.g. (Hastie *et al.*, 2001). SVMs and related methods have been introduced within the context of statistical learning theory and structural risk minimization. In the methods one solves convex optimization problems, typically quadratic programs. Least Squares Support Vector Machines (LS-SVMs)[3] (Suykens and Vandewalle, 1999; Suykens *et al.*, 2002) are reformulations to standard SVMs which lead to solving linear KKT systems for classification tasks as well as regression. In (Suykens *et al.*, 2002) LS-SVMs have been proposed as a class of kernel machines with primal-dual formulations in relation to kernel Fisher Discriminant Analysis (FDA), Ridge

[3] http://www.esat.kuleuven.ac.be/sista/lssvmlab

Regression (RR), Partial Least Squares (PLS), Principal Component Analysis (PCA), Canonical Correlation Analysis (CCA), recurrent networks and control. The dual problems for the static regression without bias term are closely related to Gaussian processes (MacKay, 1992), regularization networks (Poggio and Girosi, 1990) and Kriging (Cressie, 1993), while LS-SVMs rather take an optimization approach with primal-dual formulations which have been exploited towards large scale problems and in developing robust versions.

Direct estimation of high dimensional nonlinear functions using a non-parametric technique without imposing restrictions faces the problem of the curse of dimensionality. Several attempts were made to overcome this obstacle, including projection pursuit regression (Friedmann and Stuetzle, 1981) and kernel methods for dimensionality reduction (KDR) (Fukumizu *et al.*, 2004). Additive models are very useful for approximating high dimensional nonlinear functions (Stone, 1985; Hastie and Tibshirani, 1990). These methods and their extensions have become one of the widely used nonparametric techniques as they offer a compromise between the somewhat conflicting requirements of flexibility, dimensionality and interpretability. Traditionally, splines are a common modeling technique (Wahba, 1990) for additive models as e.g. in MARS (see e.g. (Hastie *et al.*, 2001)) or in combination with ANOVA (Neter *et al.*, 1974). Additive models were brought further to the attention of the machine learning community by e.g. (Vapnik, 1998; Gunn and Kandola, 2002). Estimation of the nonlinear components of an additive model is usually performed by the iterative *backfitting algorithm* (Hastie and Tibshirani, 1990) or a two-stage *marginal integration based* estimator (Linton and Nielsen, 1995). Although consistency of both is shown under certain conditions, important practical problems (number of iteration steps in the former) and more theoretical problems (the pilot estimator needed for the latter procedure is a too generally posed problem) are still left.

In this chapter we show how the primal-dual derivations characterizing LS-SVMs can be employed to formulate a straightforward solution to the estimation problem of additive models using convex optimization techniques for classification as well as regression problems. Apart from this one-shot optimal training algorithm, the chapter approaches the problem of structure detection in additive models (Hastie *et al.*, 2001; Gunn and Kandola, 2002) by considering an appropriate regularization scheme leading to sparse components. The additive regularization (AReg) framework (Pelckmans *et al.*, 2003) is adopted to emulate effectively these schemes based on 2-norms, 1-norms and specialized penalization terms (Antoniadis and Fan, 2001). Furthermore, a validation criterion is considered to select relevant components. Classically, exhaustive search methods (or stepwise procedures) are used which can be written as a combinatorial optimization problem. This chapter proposes a convex relaxation to the component selection problem.

This chapter is organized as follows. Section 2 presents componentwise LS-SVM regressors and classifiers for efficient estimation of additive models and relates the result with ANOVA kernels and classical estimation procedures. Section 3 introduces the additive regularization in this context and shows how to emulate dedicated regularization schemes in order to obtain sparse components. Section 4 considers the

problem of component selection based on a validation criterion. Section 5 presents a number of examples.

2 Componentwise LS-SVMs and Primal-Dual Formulations

2.1 The Additive Model Class

Giving a training set defined as $\mathcal{D}_N = \{x_k, y_k\}_{k=1}^N \subset \mathbb{R}^D \times \mathbb{R}$ of size N drawn i.i.d. from an unknown distribution F_{XY} according to $y_k = f(x_k) + e_k$ where $f : \mathbb{R}^D \to \mathbb{R}$ is an unknown real-valued smooth function, $E[y_k|X = x_k] = f(x_k)$ and $e_1, \ldots, e_N$ are uncorrelated random errors with $E[e_k|X = x_k] = 0$, $E\left[(e_k)^2|X = x_k\right] = \sigma_e^2 < \infty$. The n data points of the validation set are denoted as $\mathcal{D}_n^{(v)} = \{x_j^{(v)}, y_j^{(v)}\}_{j=1}^n$. The following vector notations are used throughout the text: $X = (x_1, \ldots, x_N) \in \mathbb{R}^{D\times N}$, $Y = (y_1, \ldots, y_N)^T \in \mathbb{R}^N$, $X^{(v)} = \left(x_1^{(v)}, \ldots, x_n^{(v)}\right) \in \mathbb{R}^{D\times n}$ and $Y^{(v)} = \left(y_1^{(v)}, \ldots, y_n^{(v)}\right)^T \in \mathbb{R}^n$. The estimator of a regression function is difficult if the dimension D is large. One way to quantify this is the optimal minimax rate of convergence $N^{-2l/(2l+D)}$ for the estimation of an l times differentiable regression function which converges to zero slowly if D is large compared to l (Stone, 1982). A possibility to overcome the curse of dimensionality is to impose additional structure on the regression function. Although not needed in the derivation of the optimal solution, the input variables are assumed to be uncorrelated (see also *concurvity* (Hastie and Tibshirani, 1990)) in the applications.

Let superscript $x^d \in \mathbb{R}$ denote the d-th component of an input vector $x \in \mathbb{R}^D$ for all $d = 1, \ldots, D$. Let for instance each component correspond with a different dimension of the input observations. Assume that the function f can be approximated arbitrarily well by a model having the following structure

$$f(x) = \sum_{d=1}^{D} f^d(x^d) + b, \tag{1}$$

where $f^d : \mathbb{R} \to \mathbb{R}$ for all $d = 1, \ldots, D$ are unknown real-valued smooth functions and b is an intercept term. The following vector notation is used: $X^d = \left(x_1^d, \ldots, x_N^d\right) \in \mathbb{R}^{1\times N}$ and $X^{(v)d} = \left(x_1^{(v)d}, \ldots, x_n^{(v)d}\right) \in \mathbb{R}^{1\times n}$. The optimal rate of convergence for estimators based on this model is $N^{-2l/(2l+1)}$ which is independent of D (Stone, 1985). Most state-of-the-art estimation techniques for additive models can be divided into two approaches (Hastie *et al.*, 2001):

- *Iterative approaches* use an iteration where in each step part of the unknown components are fixed while optimizing the remaining components. This is motivated as:
$$\hat{f}^{d_1}(x_k^{d_1}) = y_k - e_k - \sum_{d_2\neq d_1} \hat{f}^{d_2}(x_k^{d_2}), \tag{2}$$

for all $k = 1, \ldots, N$ and $d_1 = 1, \ldots, D$. Once the $N-1$ components of the second term are known, it becomes easy to estimate the lefthandside. For a large class of linear smoothers, such so-called backfitting algorithms are equivalent to a Gauss-Seidel algorithm for solving a big ($ND \times ND$) set of linear equations (Hastie *et al.*, 2001). The backfitting algorithm (Hastie and Tibshirani, 1990) is theoretically and practically well motivated.

- *Two-stages marginalization approaches* construct in the first stage a general black-box pilot estimator (as e.g. a Nadaraya-Watson kernel estimator) and finally estimate the additive components by marginalizing (integrating out) for each component the variation of the remaining components.

2.2 Componentwise Least Squares Support Vector Machine Regressors

At first, a primal-dual formulation is derived for componentwise LS-SVM regressors. The global model takes the form as in (1) for any $x_* \in \mathbb{R}^D$

$$f(x_*; w_d, b) = \sum_{d=1}^{D} f^d(x_*^d; w_d) + b = \sum_{d=1}^{D} {w_d}^T \varphi_d(x_*^d) + b. \tag{3}$$

The individual components of an additive model based on LS-SVMs are written as $f^d(x^d; w_d) = w_d^T \varphi_d(x^d)$ in the primal space where $\varphi_d : \mathbb{R} \to \mathbb{R}^{n_{\varphi_d}}$ denotes a potentially infinite ($n_{\varphi_f} = \infty$) dimensional feature map. The regularized least squares cost function is given as (Suykens *et al.*, 2002)

$$\begin{aligned} \min_{w_d, b, e_k} \mathcal{J}_\gamma(w_d, e) = \frac{1}{2} \sum_{d=1}^{D} {w_d}^T w_d + \frac{\gamma}{2} \sum_{k=1}^{N} e_k^2 \\ \text{s.t.} \quad \sum_{d=1}^{D} {w_d}^T \varphi_d(x_k^d) + b + e_k = y_k, \quad k = 1, \ldots, N. \end{aligned} \tag{4}$$

Note that the regularization constant γ appears here as in classical Tikhonov regularization (Tikhonov and Arsenin, 1977). The Lagrangian of the constraint optimization problem becomes

$$\mathcal{L}_\gamma(w_d, b, e_k; \alpha_k) = \frac{1}{2} \sum_{d=1}^{D} {w_d}^T w_d + \frac{\gamma}{2} \sum_{k=1}^{N} e_k^2 - \sum_{k=1}^{N} \alpha_k \Big(\sum_{d=1}^{D} {w_d}^T \varphi_d(x_k^d) + b + e_k - y_k \Big). \tag{5}$$

By taking the conditions for optimality $\partial \mathcal{L}_\gamma / \partial \alpha_k = 0$, $\partial \mathcal{L}_\gamma / \partial b = 0$, $\partial \mathcal{L}_\gamma / \partial e_k = 0$ and $\partial \mathcal{L}_\gamma / \partial w_d = 0$ and application of the kernel trick $K^d(x_k^d, x_j^d) = \varphi_d(x_k^d)^T \varphi_d(x_j^d)$ with a positive definite (Mercer) kernel $K^d : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, one gets the following conditions for optimality

$$\left\{ \begin{array}{rl} y_k = \sum_{d=1}^{D} {w_d}^T \varphi_d(x_k^d) + b + e_k, & k = 1, \ldots, N \quad (a) \\ e_k \gamma = \alpha_k & k = 1, \ldots, N \quad (b) \\ w_d = \sum_{k=1}^{N} \alpha_k \varphi_d(x_k^d) & d = 1, \ldots, D \quad (c) \\ 0 = \sum_{k=1}^{N} \alpha_k. & (d) \end{array} \right. \tag{6}$$

Note that condition (6.b) states that the elements of the solution vector α should be proportional to the errors. The dual problem is summarized in matrix notation as

$$\left[\begin{array}{c|c} 0 & 1_N^T \\ \hline 1_N & \Omega + I_N/\gamma \end{array}\right] \left[\begin{array}{c} b \\ \hline \alpha \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline Y \end{array}\right], \tag{7}$$

where $\Omega \in \mathbb{R}^{N\times N}$ with $\Omega = \sum_{d=1}^{D} \Omega^d$ and $\Omega_{kl}^d = K^d(x_k^d, x_l^d)$ for all $k, l = 1, \ldots, N$, which is expressed in the dual variables $\hat{\alpha}$ instead of $\hat{w}$. A new point $x_* \in \mathbb{R}^D$ can be evaluated as

$$\hat{y}_* = \hat{f}^d(x_*; \hat{\alpha}, \hat{b}) = \sum_{k=1}^{N} \hat{\alpha}_k \sum_{d=1}^{D} K^d(x_k^d, x_*^d) + \hat{b}, \tag{8}$$

where $\hat{\alpha}$ and $\hat{b}$ is the solution to (7). Simulating a validation datapoint x_j for all $j = 1, \ldots, n$ by the d-th individual component

$$\hat{y}_j^d = \hat{f}^d(x_j^d; \hat{\alpha}) = \sum_{k=1}^{N} \hat{\alpha}_k K^d(x_k^d, x_j^d), \tag{9}$$

which can be summarized as follows: $\hat{Y} = (\hat{y}_1, \ldots, \hat{y}_N)^T \in \mathbb{R}^N$, $\hat{Y}^d = \left(\hat{y}_1^d, \ldots, \hat{y}_N^d\right)^T \in \mathbb{R}^N$, $\hat{Y}^{(v)} = \left(\hat{y}_1^{(v)}, \ldots, \hat{y}_n^{(v)}\right)^T \in \mathbb{R}^n$ and $\hat{Y}_d^{(v)} = \left(\hat{y}_1^{(v)d}, \ldots, \hat{y}_n^{(v)d}\right)^T \in \mathbb{R}^n$.

Remarks:

- Note that the componentwise LS-SVM regressor can be written as a linear smoothing matrix (Suykens *et al.*, 2002):

  $$\hat{Y} = S_\gamma Y. \tag{10}$$

  For notational convenience, the bias term is omitted from this description. The smoother matrix $S_\gamma \in \mathbb{R}^{N\times N}$ becomes

  $$S_\gamma = \Omega \left(\Omega + I_N \frac{1}{\gamma}\right)^{-1}. \tag{11}$$

- The set of linear equations (7) corresponds with a classical LS-SVM regressor where a modified kernel is used (see also Figure 1)

  $$K(x_k, x_j) = \sum_{d=1}^{D} K^d(x_k^d, x_j^d). \tag{12}$$

  Figure 1 shows the modified kernel in case a one dimensional Radial Basis Function (RBF) kernel is used for all D (in the example, $D = 2$) components. This observation implies that componentwise LS-SVMs inherit results obtained for classical LS-SVMs and kernel methods in general. From a practical point of view, the previous kernels (and a fortiori componentwise kernel models) result

in the same algorithms as considered in the ANOVA kernel decompositions as in (Vapnik, 1998; Gunn and Kandola, 2002).

$$K(x_k, x_j) = \sum_{d=1}^{D} K^d(x_k^d, x_j^d) + \sum_{d_1 \neq d_2} K^{d_1 d_2}\left((x_k^{d_1}, x_k^{d_2})^T, (x_j^{d_1}, x_j^{d_2})^T\right) + \ldots, \tag{13}$$

where the componentwise LS-SVMs only consider the first term in this expansion. The described derivation bridges the gap between the estimation of additive models and the use of ANOVA kernels.

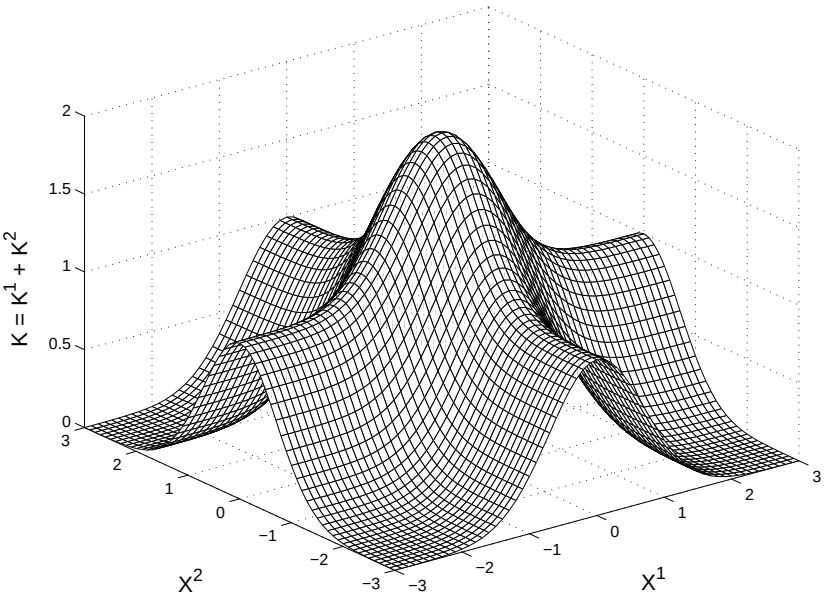


Fig. 1. *The two dimensional componentwise Radial Basis Function (RBF) kernel for componentwise LS-SVMs takes the form* $K(x_k, x_l) = K^1(x_k^1, x_l^1) + K^2(x_k^2, x_l^2)$ *as displayed. The standard RBF kernel takes the form* $K(x_k, x_l) = \exp(-\|x_k - x_l\|_2^2/\sigma^2)$ *with* $\sigma \in \mathbb{R}_0^+$ *an appropriately chosen bandwidth.*

2.3 Componentwise Least Squares Support Vector Machine Classifiers

In the case of classification, let $y_k, y_j^{(v)} \in \{-1, 1\}$ for all $k = 1, \ldots, N$ and $j = 1, \ldots, n$. The analogous derivation of the componentwise LS-SVM classifier is briefly reviewed. The following model is considered for modeling the data

$$f(x) = \operatorname{sign}\left(\sum_{d=1}^{D} f^d(x^d) + b\right), \tag{14}$$

where again the individual components of the additive model based on LS-SVMs are given as $f^d(x^d) = {w_d}^T \varphi_d(x^d)$ in the primal space where $\varphi_d : \mathbb{R} \to \mathbb{R}^{n_{\varphi_d}}$ denotes a potentially infinite ($n_{\varphi_d} = \infty$) dimensional feature map. The regularized least squares cost function is given as (Suykens and Vandewalle, 1999; Suykens *et al.*, 2002)

$$\begin{aligned} \min_{w_d,b,e_k} \mathcal{J}_\gamma(w_d, e) &= \frac{1}{2}\sum_{d=1}^{D} {w_d}^T w_d + \frac{\gamma}{2}\sum_{k=1}^{N} e_k^2 \\ \text{s.t. } & y_k\left(\sum_{d=1}^{D} {w_d}^T \varphi_d(x_k^d) + b\right) = 1 - e_k, \quad k = 1, \ldots, N, \end{aligned} \tag{15}$$

where e_k are so-called slack-variables for all $k = 1, \ldots, N$. After construction of the Lagrangian and taking the conditions for optimality, one obtains the following set of linear equations (see e.g. (Suykens *et al.*, 2002)):

$$\left[\begin{array}{c|c} 0 & Y^T \\ \hline Y & \Omega_y + I_N/\gamma \end{array}\right]\left[\begin{array}{c} b \\ \hline \alpha \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline 1_N \end{array}\right], \tag{16}$$

where $\Omega_y \in \mathbb{R}^{N\times N}$ with $\Omega_y = \sum_{d=1}^{D} \Omega_y^d \in \mathbb{R}^{N\times N}$ and $\Omega_{y,kl}^d = y_k y_l K^d(x_k^d, x_l^d)$. New data points $x_* \in \mathbb{R}^D$ can be evaluated as

$$\hat{y}_* = \operatorname{sign}\left(\sum_{k=1}^{N} \hat{\alpha}_k y_k \sum_{d=1}^{D} K^d(x_k^d, x_*^d) + \hat{b}\right). \tag{17}$$

In the remainder of this text, only the regression case is considered. The classification case can be derived straightforwardly along the lines.

3 Regularizing for Sparse Components via Additive Regularization

A regularization method fixes a priori the answer to the ill-conditioned (or ill-defined) nature of the inverse problem. The classical Tikhonov regularization scheme (Tikhonov and Arsenin, 1977) states the answer in terms of the norm of the solution. The formulation of the additive regularization (AReg) framework (Pelckmans *et al.*, 2003) made it possible to impose alternative answers to the ill-conditioning of the problem at hand. We refer to this AReg level as *substrate LS-SVMs*. An appropriate regularization scheme for additive models is to favor solutions using the smallest number of components to explain the data as much as possible. In this paper, we use the somewhat relaxed condition of *sparse* components to select appropriate components instead of the more general problem of input (or component) selection.

3.1 Level 1: Componentwise LS-SVM Substrate

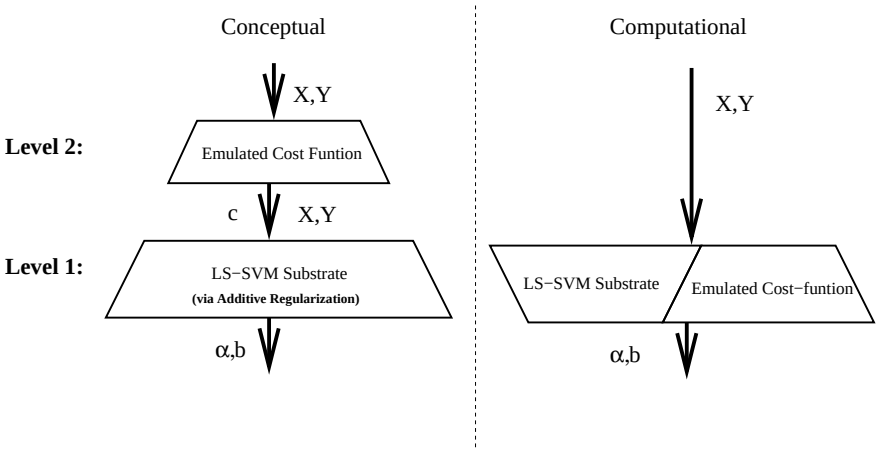


Fig. 2. *Graphical representation of the additive regularization framework used for emulating other loss functions and regularization schemes. Conceptually, one differentiates between the newly specified cost function and the LS-SVM substrate, while computationally both are computed simultanously.*

Using the Additive Regularization (AReg) scheme for componentwise LS-SVM regressors results into the following modified cost function:

$$\min_{w_d,b,e_k} \mathcal{J}_c(w_d,e) = \frac{1}{2}\sum_{d=1}^{D} {w_d}^T w_d + \frac{1}{2}\sum_{k=1}^{N}(e_k - c_k)^2$$
$$\text{s.t. } \sum_{d=1}^{D} {w_d}^T \varphi_d(x_k^d) + b + e_k = y_k, \quad k = 1,\ldots,N, \quad (18)$$

where $c_k \in \mathbb{R}$ for all $k = 1,\ldots,N$. Let $c = (c_1,\ldots,c_N)^T \in \mathbb{R}^N$. After constructing the Lagrangian and taking the conditions for optimality, one obtains the following set of linear equations, see (Pelckmans *et al.*, 2003):

$$\left[\begin{array}{c|c} 0 & 1_N^T \\ \hline 1_N & \Omega + I_N \end{array}\right] \left[\begin{array}{c} b \\ \hline \alpha \end{array}\right] + \left[\begin{array}{c} 0 \\ \hline c \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline Y \end{array}\right] \quad (19)$$

and $e = \alpha + c \in \mathbb{R}^N$. Given a regularization constant vector c, the unique solution follows immediately from this set of linear equations.

However, as this scheme is too general for practical implementation, c should be limited in an appropriate way by imposing for example constraints corresponding with certain model assumptions or a specified cost function. Consider for a moment

the conditions for optimality of the componentwise LS-SVM regressor using a regularization term as in ridge regression, one can see that equation (7) corresponds with (19) if $\gamma^{-1}\alpha = \alpha + c$ for given γ. Once an appropriate c is found which satisfies the constraints, it can be plugged in into the LS-SVM substrate (19). It turns out that one can omit this conceptual second stage in the computations by elimination of the variable c in the constrained optimization problem (see Figure 2).

Alternatively, a measure corresponding with a (penalized) cost function can be used which fulfills the role of model selection in a broad sense. A variety of such explicit or implicit limitations can be *emulated* based on different criteria (see Figure 3).

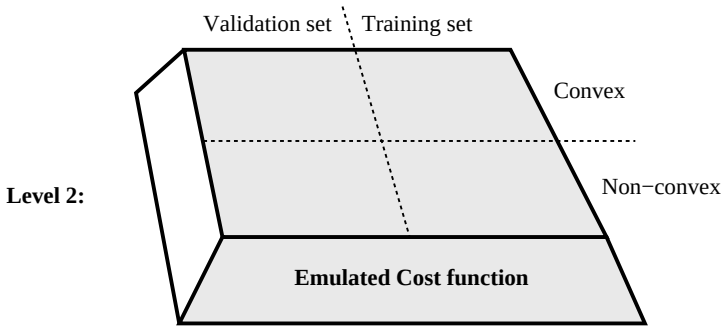


Fig. 3. *The level 2 cost functions of Figure 2 on the conceptual level can take different forms based on validation performance or trainings error. While some will result in convex tuning procedures, other may loose this property depending on the chosen cost function on the second level.*

3.2 Level 2: Emulating an L_1 based Component Regularization Scheme *(Convex)*

We now study how to obtain sparse components by considering a dedicated regularization scheme. The LS-SVM substrate technique is used to emulate the proposed scheme as primal-dual derivations (see e.g. Subsection 2.2) are not straightforward anymore.

Let $\hat{Y}^d \in \mathbb{R}^N$ denote the estimated training outputs of the d-th submodel f^d as in (9). The component based regularization scheme can be translated as the following constrained optimization problem where the conditions for optimality (18) as summarized in (19) are to be satisfied exactly (after elimination of w)

$$\min_{c,\hat{Y}^d,e_k;\alpha,b} \mathcal{J}_\xi(\hat{Y}^d, e_k) = \frac{1}{2}\sum_{d=1}^{D} \|\hat{Y}^d\|_1 + \frac{\xi}{2}\sum_{k=1}^{N} e_k^2$$
$$\text{s.t. } \begin{cases} 1_N^T\alpha = 0, \\ \Omega\,\alpha + 1_N^T b + \alpha + c = Y, \\ \Omega^d\,\alpha = \hat{Y}^d, \quad \forall d = 1,\ldots,D \\ \alpha + c = e, \end{cases} \tag{20}$$

where the use of the robust L_1 norm can be justified as in general no assumptions are imposed on the distribution of the elements of $\hat{Y}^d$. By elimination of c using the equality $e = \alpha + c$, this problem can be written as follows

$$\min_{\hat{Y}^d,e_k;\alpha,b} \mathcal{J}_\xi(\hat{Y}^d, e_k) = \frac{1}{2}\sum_{d=1}^{D} \|\hat{Y}^d\|_1 + \frac{\xi}{2}\sum_{k=1}^{N} e_k^2$$
$$\text{s.t. } \left[\begin{array}{c|c} 0 & 1_N^T \\ \hline 1_N & \Omega \\ \hline 0_N & \Omega^1 \\ \vdots & \vdots \\ 0_N & \Omega^d \end{array}\right] \left[\frac{b}{\alpha}\right] + \left[\begin{array}{c} 0 \\ \hline e \\ \hline \hat{Y}^1 \\ \vdots \\ \hat{Y}^D \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline Y \\ \hline 0_N \\ \vdots \\ 0_N \end{array}\right]. \tag{21}$$

This convex constrained optimization problem can be solved as a quadratic programming problem. As a consequence of the use of the L_1 norm, often sparse components ($\|\hat{Y}^d\|_1 = 0$) are obtained, in a similar way as sparse variables of LASSO or sparse datapoints in SVM (Hastie *et al.*, 2001; Vapnik, 1998). An important difference is that the estimated outputs are used for regularization purposes instead of the solution vector. It is good practice to omit sparse components on the training dataset from simulation:

$$\hat{f}(x_*; \hat{\alpha}, \hat{b}) = \sum_{i=1}^{N} \hat{\alpha}_i \sum_{d\in\mathcal{S}_D} K^d(x_i^d, x_*^d) + \hat{b}, \tag{22}$$

where $\mathcal{S}_D = \{d|\hat{\alpha}^T\Omega^d\hat{\alpha} \neq 0\}$.

Using the L_2 norm $\sum_{d=1}^{D}\|\hat{Y}^d\|_2^2$ instead leads to a much simpler optimization problem, but additional assumptions (Gaussianity) are needed on the distribution of the elements of $\hat{Y}^d$. Moreover, the component selection has to resort on a significance test instead of the sparsity resulting from (21). A practical algorithm is proposed in Subsection 5.1 that uses an iteration of L_2 norm based optimizations in order to calculate the optimum of the proposed regularized cost function.

3.3 Level 2 bis: Emulating a Smoothly Thresholding Penalty Function *(Non-convex)*

This subsection considers extensions to classical formulations towards the use of dedicated regularization schemes for sparsifying components. Consider the componentwise regularized least squares cost function defined as

$$\mathcal{J}_\lambda(w_d, e) = \frac{\lambda}{2}\sum_{d=1}^{D}\ell(w_d) + \frac{1}{2}\sum_{k=1}^{N} e_k^2, \tag{23}$$

where $\ell(w_d)$ is a penalty function and $\lambda \in \mathbb{R}_0+$ acts as a regularization parameter. We denote $\lambda\ell(\cdot)$ by $\ell_\lambda(\cdot)$, so it may depend on λ. Examples of penalty functions include:

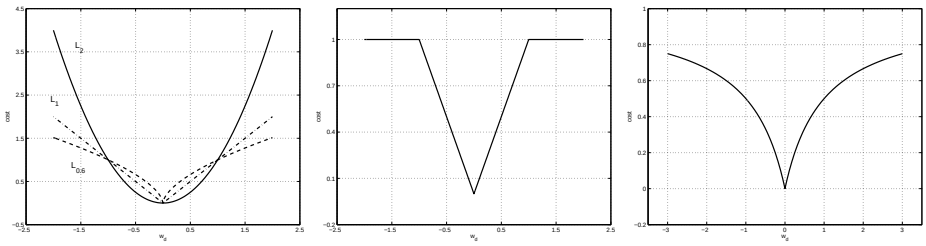
- The L_p penalty function $\ell_\lambda^p(w_d) = \lambda\|w_d\|_p^p$ leads to a bridge regression (Frank and Friedman, 1993; Fu, 1998). It is known that the L_2 penalty function $p = 2$ results in the ridge regression. For the L_1 penalty function the solution is the soft thresholding rule (Donoho and Johnstone, 1994). LASSO, as proposed by (Tibshirani, 1996; Tibshirani, 1997), is the penalized least squares estimate using the L_1 penalty function (see Figure 4.a).
- Let the indicator function $I_{\{x\in\mathcal{A}\}} = 1$ if $x \in \mathcal{A}$ for a specified set $\mathcal{A}$ and 0 otherwise. When the penalty function is given by $\ell_\lambda(w_d) = \lambda^2 - (\|w_d\|_1 - \lambda)^2 I_{\{\|w_d\|_1<\lambda\}}$ (see Figure 4.b), the solution is a hard-thresholding rule (Antoniadis, 1997).

The L_p and the hard thresholding penalty functions do not simultaneously satisfy the mathematical conditions for unbiasedness, sparsity and continuity (Fan and Li, 2001). The hard thresholding has a discontinuous cost surface. The only continuous cost surface (defined as the cost function associated with the solution space) with a thresholding rule in the L_p-family is the L_1 penalty function, but the resulting estimator is shifted by a constant λ. To avoid these drawbacks, (Nikolova, 1999) suggests the penalty function defined as

$$\ell_\lambda^a(w_d) = \frac{\lambda a\|w_d\|_1}{1 + a\|w_d\|_1}, \tag{24}$$

with $a \in \mathbb{R}_0^+$. This penalty function behaves quite similarly as the Smoothly Clipped Absolute Deviation (SCAD) penalty function as suggested by (Fan, 1997). The Smoothly Thresholding Penalty (TTP) function (24) improves the properties of the L_1 penalty function and the hard thresholding penalty function (see Figure 4.c), see (Antoniadis and Fan, 2001). The unknowns a and λ act as regularization parameters. A plausible value for a was derived in (Nikolova, 1999; Antoniadis and Fan, 2001) as $a = 3.7$. The transformed L_1 penalty function satisfies the oracle inequalities (Donoho and Johnstone, 1994). One can plugin the described semi-norm $\ell_\lambda^a(\cdot)$ to improve the component based regularization scheme (20). Again, the additive regularization scheme is used for the emulation of this scheme

$$\min_{c,\hat{Y}^d,e_k;\alpha,b} \mathcal{J}_\lambda(\hat{Y}^d, e_k) = \frac{1}{2}\sum_{d=1}^{D}\ell_\lambda^a(\hat{Y}^d) + \frac{1}{2}\sum_{k=1}^{N} e_k^2$$
$$\text{s.t.} \begin{cases} 1_N^T\alpha = 0, \\ \Omega\,\alpha + 1_N^T b + \alpha + c = Y, \\ \Omega^d\,\alpha = \hat{Y}^d, \quad \forall d = 1,\ldots,D \\ \alpha + c = e, \end{cases} \tag{25}$$



(a) (b) (c)

Fig. 4. *Typical penalty functions:* **(a)** *the L_p penalty family for $p = 2, 1$ and 0.6,* **(b)** *hard thresholding penalty function and* **(c)** *the transformed L_1 penalty function.*

which becomes non-convex but can be solved using an iterative scheme as explained later in Subsection 5.1.

4 Fusion of Componentwise LS-SVMs and Validation

This section investigates how one can tune the componentwise LS-SVMs with respect to a validation criterion in order to improve the generalization performance of the final model. As proposed in (Pelckmans *et al.*, 2003), fusion of training and validation levels can be investigated from an optimization point of view, while conceptually, they can be considered at different levels.

4.1 Fusion of Componentwise LS-SVMs and Validation for Regularization Constant Tuning

For this purpose, the fusion argument as introduced in (Pelckmans *et al.*, 2003) is briefly revised in relation to regularization parameter tuning. The estimator of the LS-SVM regressor on the training data for a fixed value γ is given as (4)

$$\textbf{Level 1}: \ (\hat{w}, \hat{b}) = \arg\min_{w,b,e} \mathcal{J}_\gamma(w, e) \quad \text{s.t. (4) holds,} \tag{26}$$

which results into solving a linear set of equations (7) after substitution of w by Lagrange multipliers α. Tuning the regularization parameter by using a validation criterion gives the following estimator

$$\textbf{Level 2}: \ \hat{\gamma} = \arg\min_{\gamma} \sum_{j=1}^{n} \Big(f(x_j; \hat{\alpha}, \hat{b}) - y_j \Big)^2 \quad \text{with} \quad (\hat{\alpha}, \hat{b}) = \arg\min_{\alpha,b} \mathcal{J}_\gamma \tag{27}$$

satisfying again (4). Using the conditions for optimality (7) and eliminating w and e

$$\textbf{Fusion}: \ (\hat{\gamma}, \hat{\alpha}, \hat{b}) = \underset{\gamma,\alpha,b}{\arg\min} \sum_{j=1}^{n} \left(f(x_j; \alpha, b) - y_j\right)^2 \quad \text{s.t.} \quad (7) \text{ holds}, \tag{28}$$

which is referred to as fusion. The resulting optimization problem was noted to be non-convex. The optimal solutions w or dual α's corresponding with a $\gamma > 0$ describe a non-convex set. To overcome this problem, a re-parameterization of the trade-off was proposed leading to the additive regularization scheme. At the cost of overparameterizing the trade-off, convexity is obtained. To circumvent this drawback, different ways to restrict explicitly or implicitly the (effective) degrees of freedom of the regularization scheme $c \in \mathcal{A} \subset \mathbb{R}^N$ were proposed while retaining convexity ((Pelckmans *et al.*, 2003)). The convex problem resulting from additive regularization is

$$\textbf{Fusion}: \ (\hat{c}, \hat{\alpha}, \hat{b}) = \underset{c\in\mathcal{A},\alpha,b}{\arg\min} \sum_{j=1}^{n} \left(f(x_j; \alpha, b) - y_j\right)^2 \quad \text{s.t.} \quad (19) \text{ holds}, \tag{29}$$

and can be solved efficiently as a convex constrained optimization problem if $\mathcal{A}$ is a convex set, resulting immediately in the optimal regularization trade-off and model parameters (Boyd and Vandenberghe, 2004).

4.2 Fusion for Component Selection using the Additive Regularization Scheme

One possible relaxed version of the component selection problem goes as follows: *Investigate whether it is plausible to drive the components on the validation set to zero without too large modifications on the global training solution*. This is translated as the following cost function much in the spirit of (20). Let $\Omega^{(v)}$ denote $\sum_{d=1}^{D} \Omega^{(v)d} \in \mathbb{R}^{n\times N}$ and $\Omega_{jk}^{(v)d} = K^d(x_j^{(v)d}, x_k^d)$ for all $j = 1, \ldots, n$ and $k = 1, \ldots, N$.

$$\begin{aligned}(\hat{c}, \hat{Y}^{(v)d}, \hat{w}_d, \hat{e}, \hat{\alpha}, \hat{b}) = \underset{c,\hat{Y}^d,\hat{Y}^{(v)d},e,\alpha,b}{\arg\min} & \frac{1}{2}\sum_{d=1}^{D} \|\hat{Y}^{(v)d}\|_1 + \frac{1}{2}\sum_{d=1}^{D} \|\hat{Y}^d\|_1 + \frac{\xi}{2}\sum_{k=1}^{N} e_k^2 \\ \text{s.t.} \quad & \begin{cases} 1_N^T \alpha = 0 \\ \alpha + c = e \\ \Omega\,\alpha + 1_N b + \alpha + c = Y \\ \Omega^d\,\alpha = \hat{Y}^d, \quad \forall d = 1, \ldots, D, \\ \Omega^{(v)d}\,\alpha = \hat{Y}^{(v)d}, \quad \forall d = 1, \ldots, D, \end{cases}\end{aligned} \tag{30}$$

where the equality constraints consist of the conditions for optimality of (19) and the evaluation of the validation set on the individual components. Again, this convex problem can be solved as a quadratic programming problem.

4.3 Fusion for Component Selection using Componentwise Regularized LS-SVMs

We proceed by considering the following primal cost function for a fixed but strictly positive $\eta = (\eta_1, \ldots, \eta_D)^T \in (\mathbb{R}_0^+)^D$

$$\textbf{Level 1}: \min_{w_d, b, e_k} \mathcal{J}_\eta(w_d, e) = \frac{1}{2}\sum_{d=1}^{D} \frac{{w_d}^T w_d}{\eta_d} + \frac{1}{2}\sum_{k=1}^{N} e_k^2$$
$$\text{s.t. } \sum_{d=1}^{D} {w_d}^T \varphi_d(x_k^d) + b + e_k = y_k, \quad k = 1, \ldots, N. \quad (31)$$

Note that the regularization vector appears here similar as in the Tikhonov regularization scheme (Tikhonov and Arsenin, 1977) where each component is regularized individually. The Lagrangian of the constrained optimization problem with multipliers $\alpha^\eta \in \mathbb{R}^N$ becomes

$$\mathcal{L}_\eta(w_d, b, e_k; \alpha_k) = \frac{1}{2}\sum_{d=1}^{D} \frac{{w_d}^T w_d}{\eta_d} + \frac{1}{2}\sum_{k=1}^{N} e_k^2 - \sum_{k=1}^{N} \alpha_k^\eta (\sum_{d=1}^{D} {w_d}^T \varphi_d(x_k^d) + b + e_k - y_k). \quad (32)$$

By taking the conditions for optimality $\partial\mathcal{L}_\eta/\partial\alpha_k = 0$, $\partial\mathcal{L}_\eta/\partial b = 0$, $\partial\mathcal{L}_\eta/\partial e_k = 0$ and $\partial\mathcal{L}_\eta/\partial w_d = 0$, one gets the following conditions for optimality

$$\begin{cases} y_k = \sum_{d=1}^{D} {w_d}^T \varphi_d(x_k^d) + b + e_k, & k = 1, \ldots, N \ (a) \\ e_k = \alpha_k^\eta, & k = 1, \ldots, N \ (b) \\ w_d = \eta_d \sum_{k=1}^{N} \alpha_k^\eta \varphi_d(x_k^d), & d = 1, \ldots, D \ (c) \\ 0 = \sum_{k=1}^{N} \alpha_k^\eta. & (d) \end{cases} \quad (33)$$

The dual problem is summarized in matrix notation by application of the kernel trick,

$$\left[\begin{array}{c|c} 0 & 1_N^T \\ \hline 1_N & \Omega^\eta + I_N \end{array}\right] \left[\begin{array}{c} b \\ \hline \alpha^\eta \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline Y \end{array}\right], \quad (34)$$

where $\Omega^\eta \in \mathbb{R}^{N\times N}$ with $\Omega^\eta = \sum_{d=1}^{D} \eta_d \Omega^d$ and $\Omega_{kl}^d = K^d(x_k^d, x_l^d)$. A new point $x_* \in \mathbb{R}^D$ can be evaluated as

$$\hat{y}_* = \hat{f}(x_*; \hat{\alpha}^\eta, \hat{b}) = \sum_{k=1}^{N} \hat{\alpha}_k^\eta \sum_{d=1}^{D} \eta_d K^d(x_k^d, x_*^d) + \hat{b}, \quad (35)$$

where $\hat{\alpha}$ and $\hat{b}$ are the solution to (34). Simulating a training datapoint x_k for all $k = 1, \ldots, N$ by the d-th individual component

$$\hat{y}_k^{\eta,d} = \hat{f}^d(x_k^d; \hat{\alpha}^\eta) = \eta_d \sum_{l=1}^{N} \hat{\alpha}_l^\eta K^d(x_k^d, x_l^d), \tag{36}$$

which can be summarized in a vector $\hat{Y}^{\eta,d} = (\hat{y}_1^d, \ldots, \hat{y}_N^d) \in \mathbb{R}^N$. As in the previous section, the validation performance is used for tuning the regularization parameters

$$\textbf{Level 2}: \ \hat{\eta} = \arg\min_{\eta} \sum_{j=1}^{n} \left(f(x_j; \hat{\alpha}^\eta, \hat{b}) - y_j \right)^2 \quad \text{with} \quad (\hat{\alpha}^\eta, \hat{b}) = \arg\min_{\alpha^\eta, b} \mathcal{J}_\eta, \tag{37}$$

or using the conditions for optimality (34) and eliminating w and e

$$\textbf{Fusion}: \ (\hat{\eta}, \hat{\alpha}^\eta, \hat{b}) = \arg\min_{\eta, \alpha^\eta, b} \sum_{j=1}^{n} \left(f(x_j; \alpha^\eta, b) - y_j \right)^2 \quad \text{s.t.} \quad (34) \text{ holds}, \tag{38}$$

which is a non-convex constrained optimization problem.

Embedding this problem in the additive regularization framework will lead us to a more suitable representation allowing for the use of dedicated algorithms. By relating the conditions (19) to (34), one can view the latter within the additive regularization framework by imposing extra constraints on c. The bias term b is omitted from the remainder of this subsection for notational convenience. The first two constraints reflect training conditions for both schemes. As the solutions α^η and α do not have the same meaning (at least for model evaluation purposes, see (8) and (35)), the appropriate c is determined here by enforcing the same estimation on the training data. In summary:

$$\left\{ \begin{array}{l} (\Omega + I_N)\,\alpha + c = Y \\ \left(\left(\sum_{d=1}^{D} \eta_d \Omega^d \right) + I_N \right) \alpha^\eta = Y \\ \left(\sum_{d=1}^{D} \eta_d \Omega^d \right) \alpha^\eta = \Omega\alpha \end{array} \right. \Rightarrow \left\{ \begin{array}{l} (\Omega + I_N)\,\alpha + c = Y \\ \\ \Omega\alpha = \eta^T \otimes I_N \left[\begin{array}{c} \Omega^1 \\ \ldots \\ \Omega^D \end{array} \right] (\alpha + c), \end{array} \right. \tag{39}$$

where the second set of equations is obtained by eliminating α^η. The last equation of the righthand side represents the set of constraints of the values c for all possible values of η. The product $\otimes$ denotes $\eta^T \otimes I_N = [\eta_1 I_N, \ldots, \eta_D I_N] \in \mathbb{R}^{N \times ND}$. As for the Tikhonov case, it is readily seen that the solution space of c with respect to η is non-convex, however, the constraint on c is recognized as a bilinear form. The fusion problem (38) can be written as

$$\textbf{Fusion}: \ (\hat{\eta}, \hat{\alpha}, \hat{c}) = \arg\min_{\eta, \alpha, c} \left\| \Omega^{(v)}\alpha - Y^{(v)} \right\|_2^2 \quad \text{s.t.} \quad (39) \text{ holds}, \tag{40}$$

where algorithms as alternating least squares can be used.

5 Applications

For practical applications, the following iterative approach is used for solving non-convex cost-functions as (25). It can also be used for the efficient solution of convex optimization problems which become computational heavy in the case of a large number of datapoints as e.g. (21). A number of classification as well as regression problems are employed to illustrate the capabilities of the described approach. In the experiments, hyper-parameters as the kernel parameter (taken to be constants over the components) and the regularization trade-off parameter γ or ξ were tuned using 10-fold cross-validation.

5.1 Weighted Graduated Non-Convexity Algorithm

An iterative scheme was developed based on the graduated non-convexity algorithm as proposed in (Blake, 1989; Nikolova, 1999; Antoniadis and Fan, 2001) for the optimization of non-convex cost functions. Instead of using a local gradient (or Newton) step which can be quite involved, an adaptive weighting scheme is proposed: in every step, the relaxed cost function is optimized by using a weighted 2-norm where the weighting terms are chosen based on an initial guess for the global solution. For every symmetric loss function $\ell(|e|) : \mathbb{R}^+ \to \mathbb{R}^+$ which is monotonically increasing, there exists a bijective transformation $t : \mathbb{R} \to \mathbb{R}$ such that for every $e = y - f(x;\theta) \in \mathbb{R}$

$$\ell(e) = (t(e))^2 . \tag{41}$$

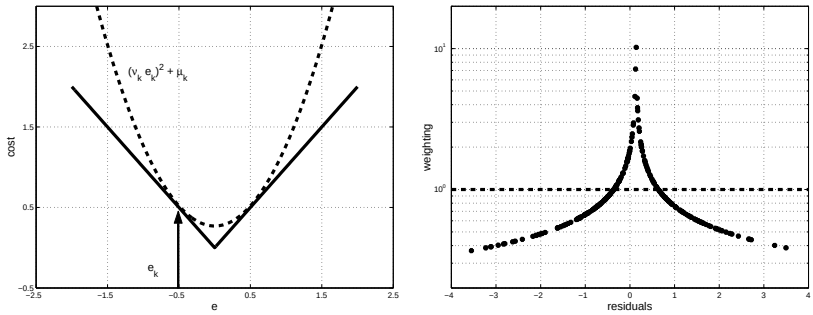
The proposed algorithm for computing the solution for semi-norms employs iteratively convex relaxations of the prescribed non-convex norm. It is somewhat inspired by the simulated annealing optimization technique for optimizing global optimization problems. The weighted version is based on the following derivation

$$\ell(e_k) = (\nu_k e_k)^2 \Leftrightarrow \nu_k = \sqrt{\frac{\ell(e_k)}{e_k^2}}, \tag{42}$$

where the e_k for all $k = 1, \ldots, N$ are the residuals corresponding with the solutions to $\theta = \arg\min_\theta \ell\left(y_k - f(x_k;\theta)\right)$ This is equal to the solution of the convex optimization problem $e_k = \arg\min_\theta \left(\nu_k(y_k - f(x_k;\theta))\right)^2$ for a set of ν_k satisfying (42). For more stable results, the gradient of the penalty function ℓ and the quadratic approximation can be takne equal as follows by using an intercept parameter $\mu_k \in \mathbb{R}$ for all $k = 1, \ldots, N$:

$$\begin{cases} \ell(e_k) = (\nu_k e_k)^2 + \mu_k \\ \ell'(e_k) = 2\nu_k^2 e_k \end{cases} \Leftrightarrow \left[\begin{array}{c|c} e_k^2 & 1 \\ \hline 2e_k & 0 \end{array}\right] \left[\begin{array}{c} \nu_k^2 \\ \hline \mu_k \end{array}\right] = \left[\begin{array}{c} \ell(e_k) \\ \hline \ell'(e_k) \end{array}\right], \tag{43}$$

where $\ell'(e_k)$ denotes the derivative of ℓ evaluated in e_k such that a minimum of $\mathcal{J}_\ell$ also minimizes the weighted equivalent (the derivatives are equal). Note that the constant intercepts μ_k are not relevant in the weighted optimization problem. Under the assumption that the two consecutive relaxations $\ell^{(t)}$ and $\ell^{(t+1)}$ do not have too different global solutions, the following algorithm is a plausible practical tool:



(a) (b)

Fig. 5. (a) *Weighted L_2-norm (dashed) approximation $(\nu_k e_k)^2 + \mu_k$ of the L_1-norm (solid) $\ell(e) = |e|_1$ which follows from the linear set of equations (43) once the optimal e_k are known;* **(b)** *the weighting terms ν_k for a sequence of e_k and $k = 1, \ldots, N$ such that $(\nu_k e_k)^2 + \mu_k = |e_k|_1$ and $2\nu_k^2 e_k = l'(e_k) = \mathrm{sign}(e_k)$ for an appropriate μ_k.*

Algorithm 1 (Weighted Graduated Non-Convexity Algorithm) *For the optimization of semi-norms ($\ell(\cdot)$), a practical approach is based on deforming gradually a 2-norm into the specific loss function of interest. Let ζ be a strictly decreasing series $1, \zeta^{(1)}, \zeta^{(2)}, \ldots, 0$. A plausible choice for the initial convex costfunction is the least squares cost function $\mathcal{J}_{\mathrm{LS}}(e) = \|e\|_2^2$.*

1. *Compute the solution $\theta^{(0)}$ for L_2 norm $J_{\mathrm{LS}}(e) = \|e\|_2^2$ with residuals $e_k^{(0)}$;*
2. *$t = 0$ and $\nu^{(0)} = 1_N$;*
3. *Consider the following relaxed cost function $\mathcal{J}^{(t)}(e) = (1-\zeta_t)\ell(e) + \zeta_t \mathcal{J}_{\mathrm{LS}}(e)$;*
4. *Estimate the solution $\theta^{(t+1)}$ and corresponding residuals $e_k^{(t+1)}$ of the costfunction $\mathcal{J}^{(t)}$ using the weighted approximation $\mathcal{J}_{\mathrm{approx}} = (\nu_k^{(t)} e_k)^2$ of $J^{(t)}(e_k)$*
5. *Reweight the residuals using weighted approximative squares norms as derived in (43):*
6. *$t := t + 1$ and iterate step (3,4,5,6) until convergence.*

When iterating this scheme, most ν_k will be smaller than 1 as the least squares cost function penalizes higher residuals (typically outliers). However, a number of residuals will have increasing weight as the least squares loss function is much lower for small residuals.

5.2 Regression examples

To illustrate the additive model estimation method, a classical example was constructed as in (Hastie and Tibshirani, 1990; Vapnik, 1998). The data were generated according to $y_k = 10\,\mathrm{sinc}(x_k^1) + 20\,(x_k^2 - 0.5)^2 + 10\,x_k^3 + 5\,x_k^4 + e_k$ were

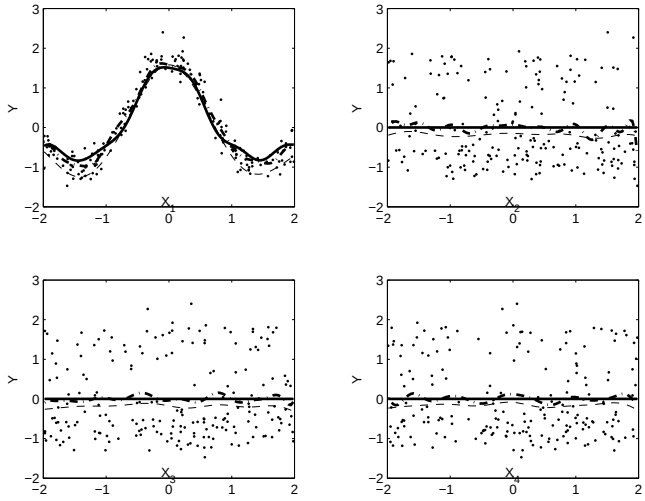


Fig. 6. *Example of a toy dataset consisting of four input components X^1, X^2, X^3 and X^4 where only the first one is relevant to predict the output $f(x) = \mathrm{sinc}(x^1)$. A componentwise LS-SVM regressor (dashed line) has good prediction performance, while the L_1 penalized costfunction of Subsection (3.2) also recovers the structure in the data as the estimated components correspnding with X^2, X^3 and X^4 are sparse.*

$e_k \sim \mathcal{N}(0,1)$, $N = 100$ and the input data X are randomly chosen from the interval $[0,1]^{10}$. Because of the Gaussian nature of the noise model, only results from Least Squares Methods are reported. The described techniques were applied on this dataset and tested on a test set. Furthermore, table 1 reports whether the algorithm recovered the structure in the data (if so, the measure is 100%). The experiment using the smoothly tresholding penalized (STP) cost function was designed as follows: for every 10 components, a version was provided for the algorithm for the use of a linear kernel and another for the use of a RBF kernel (20 components). The regularization scheme was able to select the components with the appropriate kernel (a nonlinear RBF kernel for X^1 and X^2 and linear ones for X^3 and X^4), except for one spurious component (A RBF kernel was selected for the fifth component).

5.3 Classification example

An additive model was estimated by an LS-SVM classifier based on the `spam` data as provided on the UCI benchmark repository, see e.g. (Hastie *et al.*, 2001). The data consists of words frequencies from 4601 email messages, in a study to screen email for spam. A test set of size 1536 randomly drawn from the data leaving 3065 to training purposes. The inputs were preprocessed using following transformation $p(x) = \log(1 + x)$ and standardized to unit variance. Figure 7 gives the indicator functions as found using a regularization based technique to detect structure as described in Subsection 3.3.

Method	Test set Performance			Sparse components
	L_2	L_1	L_∞	% recovered
LS-SVMs	0.1110	0.2582	0.8743	0%
componentwise LS-SVMs (7)	0.0603	0.1923	0.6249	0%
L_1 regularization (21)	0.0624	0.1987	0.6601	100%
STP with RBF (25)	0.0608	0.1966	0.6854	100%
STP with RBF and lin (25)	0.0521	0.1817	0.5729	95%
Fusion with AReg (30)	0.0614	0.1994	0.6634	100%
Fusion with comp. reg. (40)	0.0601	0.1953	0.6791	100%

Table 1. *Results on test data of numerical experiments on the Vapnik regression dataset. The sparseness is expressed in the rate of components which is selected only if the input is relevant (100% means the original structure was perfectly recovered).*

6 Conclusions

This chapter describes nonlinear additive models based on LS-SVMs which are capable of handling higher dimensional data for regression as well as classification tasks. The estimation stage results from solving a set of linear equations with a size approximatively equal to the number of training datapoints. Furthermore, the additive regularization framework is employed for formulating dedicated regularization schemes leading to structure detection. Finally, a fusion argument for component selection and structure detection based on training componentwise LS-SVMs and validation performance is introduced to improve the generalization abilities of the method. Advantages of using componentwise LS-SVMs include the efficient estimation of additive models with respect to classical practice, interpretability of the estimated model, opportunities towards structure detection and the connection with existing statistical techniques.

Acknowledgments. This research work was carried out at the ESAT laboratory of the Katholieke Universiteit Leuven. It is supported by grants from several funding agencies and sources: Research Council KU Leuven: Concerted Research Action GOA-Mefisto 666 (Mathematical Engineering), IDO (IOTA Oncology, Genetic networks), several PhD/postdoc & fellow grants; Flemish Government: Fund for Scientific Research Flanders (several PhD/postdoc grants, projects G.0407.02 (support vector machines), G.0256.97 (subspace), G.0115.01 (bio-i and microarrays), G.0240.99 (multilinear algebra), G.0197.02 (power islands), research communities ICCoS, ANMMM), AWI (Bil. Int. Collaboration Hungary/ Poland), IWT (Soft4s (softsensors), STWW-Genprom (gene promotor prediction), GBOU-McKnow (Knowledge management algorithms), Eureka-Impact (MPC-control), Eureka-FLiTE (flutter modeling), several PhD grants); Belgian Federal Government: DWTC (IUAP IV-02 (1996-2001) and IUAP V-10-29 (2002-2006) (2002-2006): Dynamical Systems and Control: Computation,

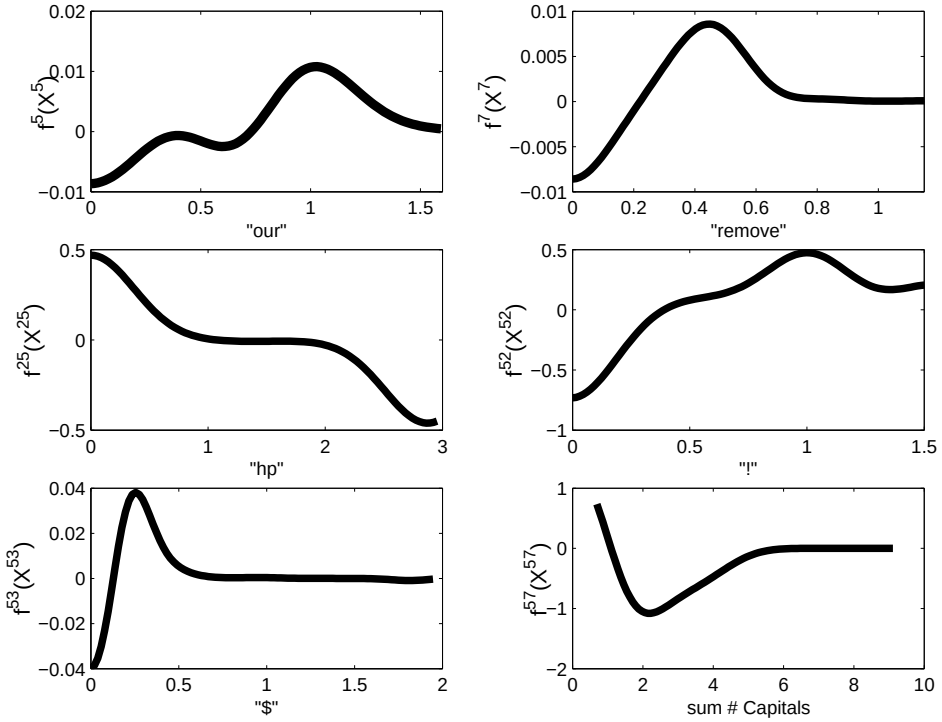


Fig. 7. *Results of the spam dataset. The non-sparse components as found by application of Subsection 3.3 are shown suggesting a number of usefull indicator variables for classifing a mail message as spam or non-spam. The final classifier takes the form* $f(X) = f^5(X^5) + f^7(X^7) + f^{25}(X^{25}) + f^{52}(X^{52}) + f^{53}(X^{53}) + f^{56}(X^{56})$ *where 6 relevant components were selected out of the 56 provided indicators.*

Identification & Modelling), Program Sustainable Development PODO-II (CP/40: Sustainibility effects of Traffic Management Systems); Direct contract research: Verhaert, Electrabel, Elia, Data4s, IPCOS. JS is an associate professor and BDM is a full professor at K.U.Leuven Belgium, respectively.

References

Antoniadis, A. (1997). Wavelets in statistics: A review. *Journal of the Italian Statistical Association* (6), 97–144.

Antoniadis, A. and J. Fan (2001). Regularized wavelet approximations (with discussion). *Journal of the American Statistical Association* **96**, 939–967.

Blake, A. (1989). Comparison of the efficiency of deterministic and stochastic algorithms for visual reconstruction. *IEEE Transactions on Image Processing* **11**, 2–12.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.

Cressie, N. A. C. (1993). *Statistics for spatial data*. Wiley.

Cristianini, N. and J. Shawe-Taylor (2000). *An Introduction to Support Vector Machines*. Cambridge University Press.

Donoho, D.L. and I.M. Johnstone (1994). Ideal spatial adaption by wavelet shrinkage. *Biometrika* **81**, 425–455.

Fan, J. (1997). Comments on wavelets in statistics: A review. *Journal of the Italian Statistical Association* (6), 131–138.

Fan, J. and R. Li (2001). Variable selection via nonconvex penalized likelihood and its oracle properties. *Journal of the American Statistical Association* **96**(456), 1348–1360.

Frank, L.E. and J.H. Friedman (1993). A statistical view of some chemometric regression tools. *Technometrics* (35), 109–148.

Friedmann, J. H. and W. Stuetzle (1981). Projection pursuit regression. *Journal of the American Statistical Association* **76**, 817–823.

Fu, W.J. (1998). Penalized regression: the bridge versus the lasso. *Journal of Computational and Graphical Statistics* (7), 397–416.

Fukumizu, K., F. R. Bach and M. I. Jordan (2004). Dimensionality reduction for supervised learning with reproducing kernel Hilbert spaces. *Journal of Machine Learning Reasearch* (5), 73–99.

Gunn, S. R. and J. S. Kandola (2002). Structural modelling with sparse kernels. *Machine Learning* **48**(1), 137–163.

Hastie, T. and R. Tibshirani (1990). *Generalized addidive models*. London: Chapman and Hall.

Hastie, T., R. Tibshirani and J. Friedman (2001). *The Elements of Statistical Learning*. Springer-Verlag. Heidelberg.

Linton, O. B. and J. P. Nielsen (1995). A kernel method for estimating structured nonparameteric regression based on marginal integration. *Biometrika* **82**, 93–100.

MacKay, D. J. C. (1992). The evidence framework applied to classification networks. *Neural Computation* **4**, 698–714.

Neter, J., W. Wasserman and M.H. Kutner (1974). *Applied Linear Statistical Models*. Irwin.

Nikolova, M. (1999). Local strong homogeneity of a regularized estimator. *SIAM Journal on Applied Mathematics* **61**, 633–658.

Pelckmans, K., J.A.K. Suykens and B. De Moor (2003). Additive regularization: Fusion of training and validation levels in kernel methods. *(Submitted for Publication) Internal Report 03-184, ESAT-SISTA, K.U.Leuven (Leuven, Belgium)*.

Poggio, T. and F. Girosi (1990). Networks for approximation and learning. In: *Proceedings of the IEEE*. Vol. 78. Proceedings of the IEEE. pp. 1481–1497.

Schoelkopf, B. and A. Smola (2002). *Learning with Kernels*. MIT Press.

Stone, C.J. (1982). Optimal global rates of convergence for nonparametric regression. *Annals of Statistics* **13**, 1040–1053.

Stone, C.J. (1985). Additive regression and other nonparameteric models. *Annals of Statistics* **13**, 685–705.

Suykens, J.A.K. and J. Vandewalle (1999). Least squares support vector machine classifiers. *Neural Processing Letters* **9**(3), 293–300.

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

Tibshirani, R.J. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society* (58), 267–288.

Tibshirani, R.J. (1997). The lasso method for variable selection in the cox model. *Statistics in Medicine* (16), 385–395.

Tikhonov, A.N. and V.Y. Arsenin (1977). *Solution of Ill-Posed Problems*. Winston. Washington DC.

Vapnik, V.N. (1998). *Statistical Learning Theory*. John Wiley and Sons.
Wahba, G. (1990). *Spline models for observational data*. SIAM.