Nonlinear Feature Selection with the Potential Support Vector Machine

Sepp Hochreiter and Klaus Obermayer

Technische Universität Berlin
Fakultät für Elektrotechnik und Informatik
Franklinstraße 28/29, 10587 Berlin, Germany
{hochreit,oby}@cs.tu-berlin.de

Summary. We describe the "Potential Support Vector Machine" (P-SVM) which is a new filter method for feature selection. The idea of the P-SVM feature selection is to exchange the role of features and data points in order to construct "support features". The "support features" are the selected features. The P-SVM uses a novel objective function and novel constraints – one constraint for each feature. As with standard SVMs, the objective function represents a complexity or capacity measure whereas the constraints enforce low empirical error. In this contribution we extend the P-SVM in two directions. First, we introduce a parameter which controls the redundancy among the selected features. Secondly, we propose a nonlinear version of the P-SVM feature selection which is based on neural network techniques. Finally, the linear and nonlinear P-SVM feature selection approach is demonstrated on toy data sets and on data sets from the NIPS 2003 feature selection challenge.

1 Introduction

Our focus is on the selection of relevant features, that is on the identification of features, which have dependencies with the target value. Feature selection is important (1) to reduce the effect of the "curse of dimensionality" (Bellman, 1961) when predicting the target in a subsequent step, (2) to identify features which allow to understand the data as well as control or build models of the data generating process, and (3) to reduce costs for future measurements, data analysis, or prediction. An example for item (2) and (3) are gene expression data sets in the medical context (e.g. gene expression patterns of tumors), where selecting few relevant genes may give hints to develop medications and reduce costs through smaller microarrays. Another example is the World Wide Web domain, where selecting relevant hyperlinks corresponds to the identification of hubs and authorities. Regarding items (2) and (3), we investigate feature selection methods which are not tailored to a certain predictor but are filter methods and lead to compact feature sets.

We propose the "Potential Support Vector Machine" (P-SVM, Hochreiter and Obermayer, 2004a) as filter method for feature selection. The P-SVM describes the classification or regression function through complex features vectors (certain

directions in input space) rather than through the input vectors as standard support vector machines (SVMs, Boser et al., 1992; Cortes and Vapnik, 1995; Schölkopf and Smola, 2002; Vapnik, 1998) do. This description imposes no restriction on the chosen function class because it is irrelevant how a function is represented. A feature value is computed by the dot produce between the corresponding complex feature vector and an input vector analogous to measurements in physics.

In the following we give an outline of the P-SVM characteristics. (1) The P-SVM avoids redundant information in the selected features as will be discussed in Section 3. Redundancy is not only opposed to compact feature sets but may reduce the performance of subsequent model selection methods as shown in (Hochreiter and Obermayer, 2004a) and in Section 5.1. For example statistical feature selection approaches suffer from redundant features. (2) The P-SVM has a sparse representation in terms of complex features as SVM-regression has with the ϵ-insensitive loss. (3) The P-SVM assigns feature relevance values which are Lagrange multipliers for the constraints and, therefore, are easy to interpret. A large absolute value of a Lagrange multiplier is associated with large empirical error if the according complex feature vector is removed from the description. (4) The P-SVM is suited for a large number of features because "sequential minimal optimization" (SMO, Platt, 1999) can be used as solver for the P-SVM optimization problem. Due to the missing equality constraint, for the P-SVM the SMO is faster than for SVMs (Hochreiter and Obermayer, 2004a). (5) The P-SVM is based on a margin-based capacity measure and, therefore, has a theoretical foundation as standard SVMs have.

In this chapter we will first introduce the P-SVM. Then we will extend the basic approach to controlling the redundancy between the selected features. Next, we describe a novel approach which extends the P-SVM to nonlinear feature selection. As discussed later in Section 4, kernelizing is not sufficient to extract the nonlinear relevance of the original features because the nonlinearities which are investigated are restricted by the kernel. Finally, we apply the generic P-SVM method to the data sets of the NIPS 2003 feature selection challenge. The "nonlinear" variant of the P-SVM feature selection method is tested on the nonlinear MADELON data set.

2 The Potential Support Vector Machine

We consider a two class classification task, where we are given the training set of m objects described by input vectors $\boldsymbol{x}_i \in \mathbb{R}^n$ and their binary class labels $y_i \in \{+1, -1\}$. The input vectors and labels are summarized in the matrix $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ and the vector $\boldsymbol{y}$. The learning task is to select a classifier g with minimal risk, $\mathrm{R}(g) = \min$, from the set of classifiers

$$g(\boldsymbol{x}) = \mathrm{sign}\left(\boldsymbol{w} \cdot \boldsymbol{x} + b\right) , \tag{1}$$

which are parameterized by the weight vector $\boldsymbol{w}$ and the offset b. The SVM optimization procedure is given by (Schölkopf and Smola, 2002; Vapnik, 1998)

$$\min_{\boldsymbol{w},b} \frac{1}{2} \|\boldsymbol{w}\|^2 \qquad \text{subject to} \quad y_i \left(\boldsymbol{w} \cdot \boldsymbol{x}_i + b\right) \geq 1 , \tag{2}$$

for linearly separable data. In this SVM formulation the constraints enforce correct classification for a hyperplane in its canonical form whereas the objective function maximizes the margin $\gamma = \|\boldsymbol{w}\|^{-1}$. The margin relates directly to a capacity measure. Let R be the radius of the sphere containing all training data, then the term $\frac{\mathrm{R}}{\gamma}$ is an upper bound for an capacity measure, the VC-dimension (Schölkopf and Smola, 2002; Vapnik, 1998). SVMs are based on the structural risk minimization principle which suggests to select from all classifiers, which correctly classify the training data, the classifier with minimal capacity.

However, the disadvantage of the SVM technique is that it is not scaling invariant, e.g. normalization of the data changes both the support vector solution and the bound $\frac{\mathrm{R}}{\gamma}$ on the capacity. If scaling is justified, we propose to scale the training data such that the margin γ remains constant while R becomes as small as possible. Optimality is achieved when all directions orthogonal the normal vector $\boldsymbol{w}$ of the hyperplane with maximal margin γ are scaled to zero. The new radius is $\tilde{\mathrm{R}} \leq \max_i |\hat{\boldsymbol{w}} \cdot \boldsymbol{x}_i|$, where $\hat{\boldsymbol{w}} := \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$. Here we assumed centered data and a centered sphere otherwise an offset allows to shift the data or the sphere. The new radius is the maximal distance from the origin in an one-dimensional problem. Finally, we suggest to minimize the new objective $\left\|\boldsymbol{X}^\top \boldsymbol{w}\right\|^2$, which is an upper bound on the new capacity measure:

$$\frac{\tilde{\mathrm{R}}^2}{\gamma^2} = \tilde{\mathrm{R}}^2 \|\boldsymbol{w}\|^2 \leq \max_i |\boldsymbol{w} \cdot \boldsymbol{x}_i| \leq \sum_i (\boldsymbol{w} \cdot \boldsymbol{x}_i)^2 = \left\|\boldsymbol{X}^\top \boldsymbol{w}\right\|^2 . \tag{3}$$

The new objective function can also be derived from bounds on the generalization error when using covering numbers because the output range of the training data – which must be covered – is bounded by $2 \max_i |\boldsymbol{w} \cdot \boldsymbol{x}_i|$. The new objective function corresponds to an implicit sphering (whitening) if the data has zero mean (Hochreiter and Obermayer, 2004c). Most importantly, the solution of eqs. (2) with objective function eq. (3) is now invariant under linear transformation of the data. Until now we motivated a new objective function. In the following we derive new constraints which ensure small empirical error.

Definition 1. *A* **complex feature vector** $\boldsymbol{z}_j$ *is a direction in the input space where the feature value* $f_{i,j}$ *of an input vector* $\boldsymbol{x}_i$ *is obtained through* $f_{i,j} = \boldsymbol{x}_i \cdot \boldsymbol{z}_j$.

We aim at expressing the constraints which enforces small empirical error by N complex feature vectors $\boldsymbol{z}_j, 1 \leq j \leq N$. Complex features and feature values are summarized in the matrices $\boldsymbol{Z} := (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N)$ and $\boldsymbol{F} = \boldsymbol{X}^\top \boldsymbol{Z}$. The ith feature vector is defined as $\boldsymbol{f}_i = (f_{i,1}, \ldots, f_{i,N}) = \boldsymbol{Z}^\top \boldsymbol{x}_i$. The complex features include Cartesian unit direction, if we set $\boldsymbol{z}_j = \boldsymbol{e}_j$, that is $\boldsymbol{Z} = \boldsymbol{I}$, $\boldsymbol{F} = \boldsymbol{X}^\top$, $f_{i,j} = x_{i,j}$ and $N = n$. In this case we obtain input variable selection. The introduction of complex feature vectors is advantageous for feature construction where a function of $\boldsymbol{Z}$ (e.g. minimal number of directions or statistical independent directions) is optimized and for handling relational data (Hochreiter and Obermayer, 2004c).

We now propose to minimize eq. (3) under constraints which are necessary for the empirical mean squared error $\mathrm{R}_{\mathrm{emp}}(g_{\boldsymbol{w},b}) = \frac{1}{2\,m} \sum_{i=1}^m (\boldsymbol{w} \cdot \boldsymbol{x}_i + b - y_i)^2$ to be minimal ($\nabla_{\boldsymbol{w}} \mathrm{R}_{\mathrm{emp}}(g_{\boldsymbol{w},b}) = 0$), and we obtain

$$\lim_{t \to 0^+} \frac{\mathrm{R}_{\mathrm{emp}}\left(g_{\boldsymbol{w}+t\boldsymbol{z}_j,b}\right) \; - \; \mathrm{R}_{\mathrm{emp}}\left(g_{\boldsymbol{w},b}\right)}{t} \; = \; \left(\boldsymbol{z}_j\right)^\top \; \nabla_{\boldsymbol{w}} \mathrm{R}_{\mathrm{emp}}\left(g_{\boldsymbol{w},b}\right) \; = \; 0 \qquad (4)$$

$$\text{and} \quad \frac{\partial \mathrm{R}_{\mathrm{emp}}\left(g_{\boldsymbol{w},b}\right)}{\partial b} \; = \; 0 \qquad (5)$$

for the constraints. The empirical error is a convex function in $(\boldsymbol{w}, b)$ and possesses only one minimum, therefore all constraints can be fulfilled simultaneously. The model selection method which combines both the new objective from eq. (3) and the new constraints from eqs. (4) is called "Potential Support Vector Machine" (P-SVM). Each complex feature $\boldsymbol{z}_j$ is associated with a constraint in eqs. (4).

Our approach enforces minimal empirical error and, therefore, is prone to overfitting. To avoid overfitting, we allow for violation of the constraints, controlled by a hyperparameter ϵ. Standardization (mean subtraction and dividing by the standard deviation) is performed for the feature values $(f_{1,j}, \ldots, f_{m,j})$. We now require, that

$$\left| \left(\boldsymbol{z}_j\right)^\top \; \nabla_{\boldsymbol{w}} \mathrm{R}_{\mathrm{emp}}\left(g_{\boldsymbol{w},b}\right) \right| \; = \; \left| \left[\boldsymbol{F}^\top \; \left(\boldsymbol{X}^\top \; \boldsymbol{w} \; - \; \boldsymbol{y} \right) \right]_j \right| \; \leq \; \epsilon \; . \qquad (6)$$

in analogy to the concept of the ϵ-insensitive loss (Schölkopf and Smola, 2002) for standard SVMs. Hence, absolute constraint values, i.e. directional derivatives, smaller than ϵ are considered to be spurious. Note, that standardization leads to $\boldsymbol{F}^\top \mathbf{1} \; = \; \mathbf{0}$ and the term $\boldsymbol{F}^\top b \; \mathbf{1}$ vanishes. The value ϵ correlates with the noise level of the data and is a hyperparameter of model selection. Combining eq. (3) and eqs. (6) results in the primal P-SVM optimization problem for feature selection:

$$\min_{\boldsymbol{w}} \; \frac{1}{2} \left\| \boldsymbol{X}^\top \; \boldsymbol{w} \right\|^2 \quad \text{subject to} \quad \begin{array}{l} \boldsymbol{F}^\top \; \left(\boldsymbol{X}^\top \; \boldsymbol{w} \; - \; \boldsymbol{y} \right) \; + \; \epsilon \; \mathbf{1} \; \geq \; \mathbf{0} \\ \boldsymbol{F}^\top \; \left(\boldsymbol{X}^\top \; \boldsymbol{w} \; - \; \boldsymbol{y} \right) \; - \; \epsilon \; \mathbf{1} \; \leq \; \mathbf{0} \end{array} , \qquad (7)$$

for which the dual formulation is the

P-SVM feature selection

$$\begin{array}{ll} \min\limits_{\boldsymbol{\alpha}^+, \boldsymbol{\alpha}^-} & \frac{1}{2} \left(\boldsymbol{\alpha}^+ \; - \; \boldsymbol{\alpha}^- \right)^\top \; \boldsymbol{F}^\top \; \boldsymbol{F} \; \left(\boldsymbol{\alpha}^+ \; - \; \boldsymbol{\alpha}^- \right) \\ & - \; \boldsymbol{y}^\top \; \boldsymbol{F} \; \left(\boldsymbol{\alpha}^+ \; - \; \boldsymbol{\alpha}^- \right) \; + \; \epsilon \; \mathbf{1}^\top \left(\boldsymbol{\alpha}^+ \; + \; \boldsymbol{\alpha}^- \right) \\ \text{subject to} & \mathbf{0} \; \leq \; \boldsymbol{\alpha}^+ \; , \quad \mathbf{0} \; \leq \; \boldsymbol{\alpha}^- \quad . \end{array} \qquad (8)$$

- ϵ: parameter to determine the number of features, large ϵ means few features
- $\alpha_j \; = \; \alpha_j^+ \; - \; \alpha_j^-$: relevance value for complex feature vector $\boldsymbol{z}_j$, $\alpha_j \; \neq \; 0$ means that vector no. j is selected, positive α_j means class 1 indicative vector $\boldsymbol{z}_j$ and negative α_j means class -1 indicative
- $\boldsymbol{F} \; = \; \boldsymbol{X}^\top \; \boldsymbol{Z}$ with data matrix $\boldsymbol{X}$ and the matrix of complex features vectors $\boldsymbol{Z}$ (variable selection: $\boldsymbol{F} \; = \; \boldsymbol{X}$)
- $\boldsymbol{y}$: vector of labels

Here $\boldsymbol{\alpha}^+$ and $\boldsymbol{\alpha}^-$ are the Lagrange multipliers for the constraints (See Hochreiter and Obermayer, 2004a, for the derivation of these equations). Eqs. (8) can be solved using a new sequential minimal optimization (SMO) technique (Hochreiter and Obermayer, 2004a). This is important to solve problems with many features because $\boldsymbol{F}^\top \boldsymbol{F}$ is a $N \times N$ matrix, therefore the optimization problem is quadratic in the number of complex features.

Using $\boldsymbol{\alpha} = \boldsymbol{\alpha}^+ - \boldsymbol{\alpha}^-$, the weight vector $\boldsymbol{w}$ and the offset b are given by

$$\boldsymbol{w} = \boldsymbol{Z}\,\boldsymbol{\alpha} = \sum_{j=1}^{N} \alpha_j\, \boldsymbol{z}_j \quad \text{and} \quad b = \frac{1}{m} \sum_{i=1}^{m} y_i \; . \tag{9}$$

Note, that for feature selection, i.e. $\boldsymbol{Z} = \boldsymbol{I}$, $\boldsymbol{w} = \boldsymbol{\alpha}$ holds, but still we recommend to solve the dual optimization problem because it has only box constraints while the primal has twice as many constraints as input variables.

The classification (or regression) function is the given by

$$g(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b = \sum_{j=1}^{N} \alpha_j\, \boldsymbol{z}_j \cdot \boldsymbol{x} + b = \sum_{j=1}^{N} \alpha_j\, f_j + b \; . \tag{10}$$

Most importantly, the vector $\boldsymbol{w}$ is expressed through a weighted sum of the complex features. Note, that the knowledge of $f_{i,j}$ and labels y_i for all training input vectors $\boldsymbol{x}_i$ is sufficient to select a classifier (see eqs. (8)). The complex feature vectors $\boldsymbol{z}_j$ must not be known explicitly. Complex feature vectors corresponding to spurious derivatives (absolute values smaller than ϵ) do not enter $\boldsymbol{w}$ because the corresponding Lagrange multipliers are zero. In particular the term $\epsilon\, \boldsymbol{1}^\top \left(\boldsymbol{\alpha}^+ + \boldsymbol{\alpha}^-\right)$ in the dual eqs. (8) leads to sparse representation of $\boldsymbol{w}$ through complex features and, therefore, to feature selection.

Note, that the P-SVM is basically a classification method. On UCI benchmark datasets the P-SVM showed comparable to better results than ν-SVMs and C-SVMs (Hochreiter and Obermayer, 2004a). However for classification the constraints are relaxed differently (by slack variables) to the approach presented here.

3 P-SVM Discussion and Redundancy Control

3.1 Correlation Considerations

In this subsection we focus on feature selection and consider the case of $\boldsymbol{F}^\top \boldsymbol{F} = \boldsymbol{X}\boldsymbol{X}^\top$ for the quadratic term of the optimization problem (8). Now it is the empirical covariance matrix of the features. The linear term $\boldsymbol{y}^\top \boldsymbol{X}^\top$ in eqs. (8) computes the correlation between features and target. Thus, such features are selected which have large target correlation and are not correlated to other features. Large target correlations result in large negative contributions to the objective function and small mutual feature correlations in small positive contributions. Consequently, highly correlated features are not selected together.

In contrast to statistical methods, the P-SVM selects features not only on the basis of their target correlation. For example, given the values of the left hand side in following table, the target t is computed from two features f_1 and f_2 as $t = f_1 + f_2$. All values have mean zero and the correlation coefficient between t and f_1 is zero. In this case the P-SVM also selects f_1 because it has negative correlation with f_2. The top ranked feature may not be correlated to the target, e.g. if it contains target-independent information which can be removed from other features.

f_1	f_2	t	f_1	f_2	f_3	t
-2	3	1	0	-1	0	-1
2	-3	-1	1	1	0	1
-2	1	-1	-1	0	-1	-1
2	-1	1	1	0	1	1

The right hand side of the table depicts another situation, where $t = f_2 + f_3$. f_1, the feature which has highest correlation coefficient with the target (0.9 compared to 0.71 of the other features) is not selected because it is correlated to all other features.

3.2 Redundancy versus Selecting Random Probes

For the NIPS feature selection challenge we applied the P-SVM technique and found that the P-SVM selected a high percentage of random probes as can be see at Table 4. Random probes are selected because they have by chance a small, random correlation with the target and are not correlated to other selected features. Whereas many features with high target correlation are not selected if they are correlated with other selected features. Avoiding redundancy results in selecting random probes.

In this subsection we extent the P-SVM approach in order to control the redundancy among the selected features. We introduce slack variables in the primal formulation eqs. (7) to allow to trade lower correlations in the objective function for errors in the constraints:

$$\begin{aligned} \min_{\boldsymbol{w}} \quad & \frac{1}{2} \left\| \boldsymbol{X}^\top \boldsymbol{w} \right\|^2 + C \, \mathbf{1}^\top \left(\boldsymbol{\xi}^+ + \boldsymbol{\xi}^- \right) \\ \text{subject to} \quad & \boldsymbol{F}^\top \left(\boldsymbol{X}^\top \boldsymbol{w} - \boldsymbol{y} \right) + \epsilon \, \mathbf{1} + \boldsymbol{\xi}^+ \geq \mathbf{0} \\ & \boldsymbol{F}^\top \left(\boldsymbol{X}^\top \boldsymbol{w} - \boldsymbol{y} \right) - \epsilon \, \mathbf{1} - \boldsymbol{\xi}^- \leq \mathbf{0} \, , \quad \mathbf{0} \leq \boldsymbol{\xi}^+, \boldsymbol{\xi}^- \, . \end{aligned} \tag{11}$$

As dual formulation we obtain the following optimization problem.

P-SVM feature selection with redundancy control

$$\begin{aligned} \min_{\boldsymbol{\alpha}^+,\boldsymbol{\alpha}^-} \quad & \frac{1}{2}\left(\boldsymbol{\alpha}^+ - \boldsymbol{\alpha}^-\right)^\top \boldsymbol{F}^\top \boldsymbol{F} \left(\boldsymbol{\alpha}^+ - \boldsymbol{\alpha}^-\right) \\ & - \boldsymbol{y}^\top \boldsymbol{F} \left(\boldsymbol{\alpha}^+ - \boldsymbol{\alpha}^-\right) + \epsilon\, \mathbf{1}^\top \left(\boldsymbol{\alpha}^+ + \boldsymbol{\alpha}^-\right) \\ \text{subject to} \quad & \mathbf{0} \leq \boldsymbol{\alpha}^+ \leq C\,\mathbf{1}\,, \quad \mathbf{0} \leq \boldsymbol{\alpha}^- \leq C\,\mathbf{1}\ . \end{aligned} \tag{12}$$

- variables as in problem eqs. (8)
- C controls redundancy of selected features, small C results in more redundancy

The eqs. (9) for $\boldsymbol{w}$ and b still hold. The effect of introducing the slack variables can be best seen at the dual problem. Because the α_j are bounded by C, high correlations are lower weighted in the objective function. Consequently, correlated features have a lower positive contribution in the objective function and, therefore, selecting redundant features does not cost as much as in the original P-SVM formulation. The effect is demonstrated at the following two toy experiments.

In the first two class classification experiment six dimension out of 100 are indicative for the class. The class membership was chosen with equal probability (0.5) and with equal probability 0.5 either the first three features were class indicators or the features 4 to 6. If the first three features are class indicators, features a chosen according to $x_{i,j} \sim y_i\ \mathrm{N}(j,1)$, $1 \leq j \leq 3$, $x_{i,j} \sim \mathrm{N}(0,1)$, $4 \leq j \leq 6$, $x_{i,j} \sim \mathrm{N}(0,20)$, $7 \leq j \leq 100$. If features 4 to 6 are class indicators, features a chosen according to $x_{i,j} \sim \mathrm{N}(0,1)$, $1 \leq j \leq 3$, $x_{i,j} \sim y_i\ \mathrm{N}(j-3,1)$, $4 \leq j \leq 6$, $x_{i,j} \sim \mathrm{N}(0,20)$, $7 \leq j \leq 100$. Only the first six feature are class indicators but mutual redundant. Finally, the class labels were switched with probability 0.2. In the experiments ϵ is adjusted to obtain 6 features, i.e. to obtain 6 support vectors. The top part of Table 1 shows the result for different values of C. With decreasing C more relevant features are selected because the redundancy weighting is down-scaled.

In the next experiment we extended previous experiment by using 940 probes and 60 features (1000 input components), where either the first 30 or features 31 to 60 are indicative for the class label. Indicative features are chosen according to $x_{i,j} \sim \mathrm{N}(2,1)$. All other value were as in previous experiment. The value of ϵ is adjusted to a value that only about 60 features are selected ($\alpha \neq 0$). The bottom part of Table 1 shows the result for different values of C. The percentage of probes in the selected variables is 93 % for $C = 10$ and reduces to 33 % for $C = 0.003$.

These simple experiments demonstrated that the original P-SVM selects many random probes because it minimized feature redundancy. Here we controlled this effect by introducing slack variables. Note, that for the NIPS challenge submissions no slack variables were used.

3.3 Comparison to Related Methods

1-norm SVMs (Bi et al., 2003). The P-SVM feature selection is related to the 1-norm SVMs because both use a 1-norm sparsity constraint. However the P-SVM contains – in contrast to the 1-norm SVM – a quadratic part. The effect of the

Table 1. Toy example for redundancy control. TOP: Feature ranking where the first 6 features are relevant to predict the class label. With decreasing C more redundant features are selected and, therefore, more relevant features are found (their number is given in column "c"). α values are given in brackets. BOTTOM: 60 relevant features exist. Starting from 4 relevant features (93 % probes) reducing C leads to 40 relevant features (33 % probes).

C	ϵ	c	1.	2.	3.	4.	5.	6.
10	1.5	3	6 (1.78)	2 (1.12)	26(-0.55)	18(-0.44)	3 (0.35)	52(-0.07)
1	2	4	6 (1.00)	2 (0.71)	5 (0.29)	3 (0.13)	18(-0.09)	26(-0.06)
0.5	2	5	2 (0.50)	5 (0.50)	6 (0.50)	3 (0.22)	18(-0.09)	4 (0.04)

C	ϵ	# relevant features	C	ϵ	# relevant features
10	0.65	4	0.1	1.7	23
0.5	1	11	0.05	2	31
0.2	1.45	17	0.003	2	40

quadratic part was demonstrated in Subsection 3.1, were we found that important features are select through correlation with other features. Comparisons can be found in the experiments in Subsection 5.1.

Zero-norm SVMs (Weston et al., 2003). Zero-norm SVMs optimize a different objective than the P-SVM, where the scaling factor of the selected features is no longer important. Scaling factors may, however, be important if different features contain different levels of noise. We compared the P-SVM with zero-norm SVMs in the experiments in Subsection 5.1 (only 2 features selected). The zero-norm SVMs select features by successively repeating 1-norm SVMs. The P-SVMs can be extended in a similar way if after standardization features are weighted by their actual importance factors.

LASSO (Tibshirani, 1996). The LASSO is quite similar to the P-SVM method. In contrast to P-SVM, LASSO does not use the linear term of the dual P-SVM in the objective function but constraints it. P-SVM is derived from an SVM approach, therefore contains a primal and a dual formulation which allows to apply a fast SMO procedure. A major difference between P-SVM and LASSO is that LASSO cannot control the redundancy among the selected features as the P-SVM can with its slack variables as demonstrated in Subsection 3.2. Comparisons to LASSO are implicitly contained the NIPS feature selection challenge, where the methods of Saharon Rosset and Ji Zhu are based on the LASSO.

4 Nonlinear P-SVM Feature Selection

In this section we extend the P-SVM feature selection approach to assigning relevance values to complex features $\boldsymbol{z}_j$, where we now consider also nonlinear combinations of the features. To construct new features by nonlinearly combining the original features (Kramer, 1991; Oja, 1991; Schölkopf et al., 1997; Smola et al., 2001; Tishby et al., 1999) by using kernels is possible but not sufficient to extract arbitrary nonlinear dependencies of features. Only those nonlinearities can be detected which are determined by the kernel. A wrong kernel choice does not allow to extract

proper nonlinearities. Therefore, we attempt to construct proper nonlinearities by training multi-layer perceptrons (MLPs). After training we determine the relevance of input variables. Our approach is related to input pruning methods (Hassibi and Stork, 1993; Moody and Utans, 1992) and automatic relevance determination (ARD, MacKay, 1993; Neal, 1996).

For input $\boldsymbol{x}$ the value $y(\boldsymbol{x})$ is the output function of the MLP and $\text{net}_l = \boldsymbol{w}_l \cdot \boldsymbol{x} + b_l$ the net input of the hidden unit l. After training on the training set $\{(\boldsymbol{x}_i, y_i)\}\}$, we set $\text{net}_l = 0$ for the forward and backward pass. For training example $\boldsymbol{x}_i$ this leads to a new output of $\tilde{y}_l(\boldsymbol{x}_i)$ and an induced error of $e_l(\boldsymbol{x}_i) = \frac{1}{2}(\tilde{y}_l(\boldsymbol{x}_i) - y(\boldsymbol{x}_i))^2$. The error indicates the relevance of net_l but does not supply a desired value for net_l. However, the gradient descent update signal for net_l supplies a new target value $y_{l,i}$ for $\text{net}_l(\boldsymbol{x}_i)$ and we arrive at the regression task:

$$y_{l,i} = \boldsymbol{w}_l \cdot \boldsymbol{x}_i + b_l \;, \quad y_{l,i} := -\frac{\partial e_l(\boldsymbol{x}_i)}{\partial \text{net}_l(\boldsymbol{x}_i)} \;. \tag{13}$$

This regression problem is now solved by the P-SVM which selects the relevant input variables for hidden unit l. Fig. 1 depicts the regression task. The vectors $\boldsymbol{w}_l$ are now expressed through complex features $\boldsymbol{z}_j$ and allow to assign for each l a relevance value $\alpha_{j,l}$ to $\boldsymbol{z}_j$. Finally, the results for all l are combined and the complex features $\boldsymbol{z}_j$ are ranked by their relevance values $\alpha_{j,l}$, e.g. by the maximal absolute weight or squared weight sum. The pseudo code of the algorithm is shown in Algorithm 1.

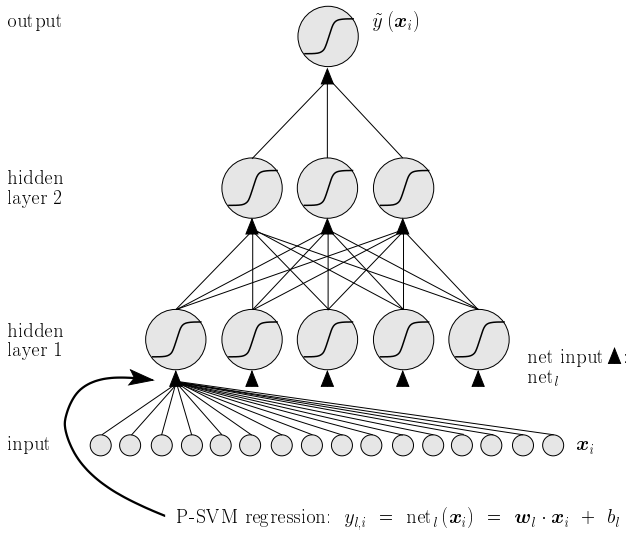

Fig. 1. Outline of the nonlinear P-SVM. After MLP training the P-SVM solves the regression task $y_{l,i} = \boldsymbol{w}_l \cdot \boldsymbol{x}_i + b_l$, where $y_{l,i} := -\frac{\partial e_l(\boldsymbol{x}_i)}{\partial \text{net}_l(\boldsymbol{x}_i)}$ for $\boldsymbol{w}_l = \boldsymbol{0}$ and $b_l = 0$. The P-SVM selects the relevant input variables to hidden unit l.

Algorithm 1 Nonlinear P-SVM Feature Selection

BEGIN INITIALIZATION
training set $\{(\boldsymbol{x}_i, y_i)\}$,
MLP architecture and activation function,
MLP training parameters (learning rate),
MLP learning stop criterion: small error threshold
END INITIALIZATION

BEGIN PROCEDURE
Step 1: perform standardization
Step 2: train an MLP with standard back-propagation until stop criterion
Step 3: {determine feature relevance values for each new feature}
for all hidden units l in chosen hidden layer **do**
for $i = 1$ to m **do**
MLP forward pass with unit l clamped to 0
MLP backward pass to compute $y_{l,i} = -\frac{\partial e_l(\boldsymbol{x}_i)}{\partial \mathrm{net}_l(\boldsymbol{x}_i)}$
end for
hidden layer with the new features (recommended: first hidden layer),
chose P-SVM parameter ϵ
solve regression task $\forall_i : \quad y_{l,i} = \boldsymbol{w}_l \cdot \boldsymbol{x}_i + b_l$ by the P-SVM method and determine relevance values $\alpha_{j,l}$ per new feature
end for
Step 4: {compute relevance values}
combine (squared sum or maximal value) all $\alpha_{j,l}$ to determine relevance of $\boldsymbol{z}_j$
END PROCEDURE

Other targets. The net input $\mathrm{net}_l(\boldsymbol{x}_i)$ as target value instead of $y_{l,i}$ does not take into account that a hidden unit may not be used, may be less used than others, or may have varying influence on the net output over the examples. Setting net_l to other values than 0 (e.g. its mean value) when $y_{l,i}$ is computed works as well as long as saturating regions are avoided. Saturation regions lead to scaling effects through different derivatives at different input regions and reduce the comparability of relevance values of one input variable at different units.

Redundancy. The P-SVM is applied to each unit l, therefore the selected input variables may be redundant after combining the results for each l.

5 Experiments

5.1 Linear P-SVM Feature Selection

Weston Data

We consider a 2 class classification task with 600 data points (300 from each class) which is similar to the data set in (Weston et al., 2000) but more difficult. 100 randomly chosen data points are used for feature and model selection. The remaining

500 data points serve as test set. We constructed 2000 input variables from which only the first 20 input variables have dependencies with the class and the remaining 1980 are random probes. For each data point four out of the first 20 input variables are indicative for the class label. The data points are in one of five modes, where the mode determines which input variable is indicative. The modes, which lead to objects groups, are $l = 0, 4, 8, 12, 16$ with associated input variables: $l = 0 \longrightarrow \boldsymbol{x}_{i,1} - \boldsymbol{x}_{i,4}$; $l = 4 \longrightarrow \boldsymbol{x}_{i,5} - \boldsymbol{x}_{i,8}$; $l = 8 \longrightarrow \boldsymbol{x}_{i,9} - \boldsymbol{x}_{i,12}$; $l = 12 \longrightarrow \boldsymbol{x}_{i,13} - \boldsymbol{x}_{i,16}$; $l = 16 \longrightarrow \boldsymbol{x}_{i,17} - \boldsymbol{x}_{i,20}$.

A label from $\{+1, -1\}$ and a mode from $\{0, 4, 8, 12, 16\}$ was randomly and uniformly chosen. Then the four indicative input variables $x_{i,l+\tau}$, $1 \leq \tau \leq 4$, were chosen according to $x_{i,l+\tau} \sim y_i \cdot \mathrm{N}(2, 0.5\ \tau)$. Input variables $x_{i,j}$ for $j \neq l+\tau$, $j \leq 20$ were chosen according to $x_{i,j} \sim \mathrm{N}(0, 1)$. Finally, for $21 \leq j \leq 2000$ the input variables $x_{i,j}$ were chosen according to $x_{i,j} \sim \mathrm{N}(0, 20)$.

Table 2. Classification performance for the "Weston" data set. Results are an average over 10 runs on different training and test sets. The values are the fractions of misclassification. The table shows the results using the top ranked 5, 10, 15, 20, and 30 features for the methods: Fisher statistics (Kendall and Stuart, 1977), Recursive Feature Elimination (RFE), R2W2, and the P-SVM.

Method	number of features				
	5	10	15	20	30
Fisher	0.31	0.28	0.26	0.25	0.26
RFE	0.33	0.32	0.32	0.31	0.32
R2W2	0.29	0.28	0.28	0.27	0.27
P-SVM	0.28	0.23	0.24	0.24	0.26

We compare the linear P-SVM feature selection technique to Fisher statistics (Kendall and Stuart, 1977), Recursive Feature Elimination (RFE) method of Guyon et al. (2002) and the linear R2W2 method (Weston et al., 2000). The experiment is taken from (Hochreiter and Obermayer, 2004b). First we ranked features on the training set, where for RFE the ranking was based on multiple runs. Then we trained a standard C-SVM[1] with the top ranked 5, 10, 15, 20, and 30 input variables. The hyperparameter C was selected from the set $\{0.01, 0.1, 1, 10, 100\}$ through 5-fold cross-validation. Table 2 shows the results. The P-SVM method performed best.

The performance of the methods depends on how many modes are represented through the input variables. The results in Table 2 must be compared to the classification performance with 20 relevant features (perfect selection), which leads to a fractional error of 0.10, and without feature selection, which leads to a fractional error of 0.38.

This benchmark is a very difficult feature selection task because it contains many features but only few of them are indicative, features are indicative for only 1/5 of the data, and features are noisy. It is difficult to extract the few indicative features for all objects groups with few examples available because target correlation by chance is likely for some features.

[1]For this experiment we used the Spider-Software, where the C-SVM was easier to use as classifier than the ν-SVM.

Two best features experiment

To compare the P-SVM feature selection technique to the 1-norm and 0-norm support vector machine by performing the benchmark in Weston et al. (2003). The two class classification task has six dimension out of 100 which are indicative for the class. The class membership was chosen with equal probability (0.5) and with probability 0.7 the first three features were class indicators and otherwise the features 4 to 6 are class indicators. For the first case input variables are chosen according to $x_{i,j} \sim y_i \, \mathrm{N}(j, 1)$, $1 \leq j \leq 3$, $x_{i,j} \sim \mathrm{N}(0, 1)$, $4 \leq j \leq 6$, and $x_{i,j} \sim \mathrm{N}(0, 20)$, $7 \leq j \leq 100$. For the second case the input variables are $x_{i,j} \sim \mathrm{N}(0, 1)$, $1 \leq j \leq 3$, $x_{i,j} \sim y_i \, \mathrm{N}(j-3, 1)$, $4 \leq j \leq 6$, and $x_{i,j} \sim \mathrm{N}(0, 20)$, $7 \leq j \leq 100$. Only the first six input variables are class indicators but mutual redundant. The two top ranked input variables are used for classification. Training and feature selection is performed on 10, 20, and 30 randomly chosen training points and the selected model is tested on additionally 500 test points. The result is an average over 100 trials.

The feature selection methods, which are compared in Weston et al. (2003), are: no feature selection (no FS), 2-norm SVM (largest weights), 1-norm SVM (largest weights), correlation coefficient (CORR), RFE, R2W2, and three approaches to zero-norm feature selection, namely FSV (Bradley and Mangasarian, 1998; Bradley et al., 1998), ℓ_2-AROM, and ℓ_1-AROM (Weston et al., 2003). The correlation coefficient is computed as $(\mu_+ - \mu_-)^2 \; / \; \left(\sigma_+^2 + \sigma_-^2\right)$, where μ_+ and σ_+ are the mean and the standard deviation of the feature value for the positive class and μ_- and σ_- the according values for the negative class. The authors in (Weston et al., 2003) only mentioned that they used "linear decision rules" while used for the P-SVM a linear ν-SVM with $\nu = 0.3$ as classifier.

Table 3. Comparison of different compact feature set selection methods. The percentage of the test error with its standard deviation in rectangular brackets is given. The number of trials where two relevant non-redundant features are selected is in round brackets. For 10 and 20 point the P-SVM method performs as good as the best methods and for 30 data points the P-SVM performs worse than the zero-norm methods but better than the others.

Method	10 points	20 points	30 points
no FS	33.8 [std: 6.6] (0)	23.2 [std: 5.6] (0)	16.4 [std: 3.9] (0)
2-norm SVM	26.8 [std:13.9] (3)	16.3 [std: 7.7] (16)	13.4 [std: 4.2] (17)
1-norm SVM	25.9 [std:14.5] (17)	11.0 [std:10.9] (67)	12.1 [std:13.5] (66)
CORR	23.6 [std:12.9] (9)	15.8 [std: 5.4] (9)	14.3 [std: 3.2] (5)
RFE	30.1 [std:14.5] (10)	11.6 [std:11.0] (64)	8.2 [std: 6.1] (73)
R2W2	26.3 [std:14.1] (14)	9.8 [std: 8.6] (66)	7.8 [std: 6.1] (67)
FSV	24.6 [std:14.9] (17)	9.1 [std: 8.3] (70)	5.9 [std: 5.4] (85)
ℓ_2-AROM	26.7 [std:14.6] (15)	8.8 [std: 9.0] (74)	5.7 [std: 5.0] (85)
ℓ_1-AROM	25.8 [std:14.9] (20)	8.9 [std: 9.7] (77)	5.9 [std: 5.1] (83)
P-SVM	26.0 [std:13.8] (13)	8.6 [std: 7.4] (67)	6.9 [std: 9.1] (73)

Table 3 shows the results as an average over 100 trials. The table reports the percentage of test error with the according standard deviation[2] and the number of times that the selected features are relevant and non-redundant. The P-SVM method performs as good as the best methods for 10 and 20 data points but for 30 data points worse than the zero-norm methods and better than other methods. Because the zero-norm approaches solve iteratively one- or two-norm SVM problems, it may be possible to do the same for the P-SVM approach by re-weighting the features by their α-values.

NIPS Challenge

In this section we report the results of the P-SVM method for the NIPS 2003 feature selection challenge. The method and the results are given in the Fact Sheet **??** and the results at the top of Table 4. In order to obtain a compact feature set we applied the P-SVM method without slack variables. Therefore, the P-SVM method selects a high percentage of random "probes", i.e. features which are artificially constructed and are not related to the target. Especially prominent is this behavior for the data set ARCENE, where features are highly correlated with each other. This correlation was figured out by a post challenge submission and by the data set description which was made available after the challenge.

We computed the NIPS challenge results for methods with compact feature sets, i.e. methods which based their classification on less than 10 % extracted features. Only methods are reported which have a non-negative score to ensure sufficient classification performance. Bottom of Table 4 reports the results. The P-SVM method yields good results if compact feature sets are desired. In summary, the P-SVM method has shown good performance as a feature selection method especially for compact feature sets.

5.2 Nonlinear P-SVM Feature Selection

Toy Data

In this experiment we constructed two data sets in which the relevant features cannot be found by linear feature selection techniques. We generated 500 data vectors $\boldsymbol{x}_i$ ($1 \leq i \leq 500$) with 100 input variables $x_{i,j}$ ($1 \leq j \leq 100$). Each input variable was chosen according to $x_{i,j} \sim \mathrm{N}(0,1)$. The attributes y_i of the data vectors $\boldsymbol{x}_i$ were computed from the first two variables by A) $y_i = x_{i,1}^2 + x_{i,2}^2$ and B) $y_i = x_{i,1} x_{i,2}$. We thresholded y_i by $y_i > 1 \Rightarrow y_i = 1$ and $y_i < -1 \Rightarrow y_i = -1$. For both tasks the correlation coefficient between the target and the relevant input variables is zero. For task A) this follows from the fact that the first and third moments of the zero-mean Gaussian are zero and for task B) if follows from the zero mean of input variables (XOR problem).

We performed 10 trials for each task with the P-SVM nonlinear relevance extraction method. First, a 3-layered multi-layer perceptron (100 inputs, 10 hidden, one output) with sigmoid units in $[-1, 1]$ was trained until the error was 5 % of its

[2]Note, that in Weston et al. (2003) the standard deviation of the mean is given, which scales the standard deviation by a factor of 10.

Table 4. NIPS 2003 challenge results for P-SVM. "Score": The score used to rank the results by the organizers (times 100). "BER": Balanced error rate (in percent). "AUC": Area under the ROC curve (times 100). "Feat": Percent of features used. "Probe": Percent of probes found in the subset selected. "Test": Result of the comparison with the best entry using the MacNemar test. TOP: General result table. BOTTOM: Results for compact feature sets with non-negative score. The column "Method" gives the method name. The P-SVM has multiple entries were different weighting of the CV folds is used to select features and hyperparameters. The results are listed according to the percentage of features used.

Dec. 1^{st}	Our best challenge entry					The winning challenge entry					
Dataset	Score	BER	AUC	Feat	Probe	Score	BER	AUC	Feat	Probe	Test
Overall	14.18	11.28	93.66	4.6	34.74	88.00	6.84	97.22	80.3	47.8	1
Arcene	16.36	20.55	87.75	7	61	98.18	13.30	93.48	100	30.0	1
Dexter	-60	8.70	96.39	2.5	46.6	96.36	3.90	99.01	1.5	12.9	1
Dorothea	29.09	16.21	88.00	0.2	29.58	98.18	8.54	95.92	100	50.0	1
Gisette	18.18	2.06	99.76	12	36.5	98.18	1.37	98.63	18.3	0.0	1
Madelon	67.27	8.89	96.39	1.4	0	100.0	7.17	96.95	1.6	0.0	1

Dec. 1^{st}	Method	Feat	Score	BER	AUC	Probe	Test
	P-SVM (1)	3.83	0	11.82	93.41	34.6	1
	Modified-RF	3.86	6.91	10.46	94.58	9.82	1
	P-SVM (2)	4.63	14.18	11.28	93.66	34.74	1
	BayesNN-small	4.74	68.73	8.20	96.12	2.91	0.8
	final-1	6.23	40.36	10.38	89.62	6.1	0.6
	P-SVM (3)	7.38	5.09	12.14	93.46	45.65	1
	Collection2	7.71	28	10.03	89.97	10.6	1

initial value. The P-SVM method was applied and features were ranked according to their maximal values of $\alpha_{j,l}$. In all trials the P-SVM ranked the two relevant features $x_{i,1}$ and $x_{i,2}$ on top and produced a clear visible gap between the relevance values of the true relevant features and the remaining features. For comparison we also performed 5 trials with linear P-SVM feature selection on each of both tasks. *The linear version failed to detect the true relevant features.* For comparison we selected input variables with "Optimal Brain Surgeon" (OBS, Hassibi and Stork, 1993) and "Optimal Brain Damage" (OBD, LeCun et al., 1990). We applied OBS and OBD in two ways after the neural network has been trained: first, we computed the saliency values for all weights with OBS and OBD and ranked the features according to their highest values; secondly, we successively deleted weights according to the OBS and OBD procedure and ranked a feature before another feature if at least one input weight is removed later than all the input weights of the other feature. For the latter we retrained the neural network if the error increase more than 10 % since the last training. OBS and OBD lead also to success at this task. This experiment demonstrated that P-SVM nonlinear relevance extraction is able to reliably detect relevant features whereas the linear P-SVM method could not identify relevant features.

NIPS Challenge: Madelon

The data generation procedure of the NIPS feature selection challenge was made public after the challenge. Therefore, we know that the class labels for the data set MADELON were constructed nonlinearly from the input variables. After the challenge, when knowing the data generation process, we computed Pearson's correlation coefficient for each pair of features. That allowed us to extract the 20 relevant features through looking for a set of 20 features which have high intercorrelation.

For nonlinear feature selection (P-SVM, OBS, OBD) we used 3-layered multilayer perceptrons (MLPs) with 20 hidden units and 4-layered MLPs with 10 hidden units in each hidden layer. All non-input units have a sigmoid activation function in range $[-1, 1]$. We trained the MLPs with backpropagation until the error was at 5 % of its initial value. Features were ranked by the P-SVM, OBS, and OBD as in Subsection 5.2.

The linear P-SVM ranked in 10 runs 13 out of 20 true relevant features at the top. Nonlinear P-SVMs with 3-layered and 4-layered nets ranked in 10 runs always 18 to 20 relevant features at the top, however no gap in relevance values between features and probes was visible. Table 5 shows typical results. Increasing the ϵ value produces a gap in the relevance values between the true relevant features and the probes, however fewer true relevant features are ranked at the top (see in Table 5). Both the ranking by OBS and OBD through the saliency and through backward elimination lead to inferior results compared to the P-SVM method. On average 3 true relevant features were extracted (Table 5 presents typical results). For backward elimination we started by removing a sets of weights (4-layered: 4×500, 3×400, 3×300, 3×200, 2×100, 3×50, 20, 2×10 = 5,090; 3-layered: 19×500, 200, 100, 2×50, 5×20 = 10,000). After removing a set we extensively retrained. After removing weight sets, we deleted weights step by step. As seen in previous studies, OBS and OBD tend to keep large weights which result from overfitting (Hochreiter and Schmidhuber, 1997). Only the nonlinear P-SVM was able to rank almost all relevant features at the top. This experiment showed that the nonlinear extension of the P-SVM feature selection method can detect relevant features which are missed with the linear version and also missed by OBS and OBD.

6 Conclusion

In the future we intend to investigate how optimization of the complex feature vectors (e.g. to obtain few feature vectors or independent features) can be integrated into our approach. Further we intend to apply the P-SVM method to genomic data (e.g. microsatellites) to identify genetic causes for various diseases (e.g. schizophrenia).

On the NIPS 2003 feature selection challenge data sets we have experimentally shown that the linear P-SVM method is one of the best methods for selecting a compact feature set. The linear P-SVM approach has been generalized to include redundancy control and nonlinearities. Nonlinear P-SVM feature selection does not only extract features which are missed by its linear version but has the potential to give the features a more appropriate ranking. This property is especially important for data sets, where only few top ranked features control the data generating process.

Table 5. MADELON **nonlinear feature selection examples.** Typical runs for linear (average over four runs with different ϵ) and nonlinear P-SVM, OBS, and OBD. Selected input variables are ordered line-wise and true features are marked boldface. The nonlinear methods are based on a 3-layered and a 4-layered neural network. OBS and OBD ranking uses either the saliency values or successively deleted weights. For the latter a feature is ranked according to when its last weight is deleted. For the P-SVM the ϵ values are given in brackets. The linear P-SVM was not able to find all true relevant features whereas the nonlinear P-SVM finds all of them.

Method	feature ranking									
linear P-SVM	**242**	**476**	**337**	**65**	**339**	**454**	**494**	**443**	**49**	**379**
(ϵ = 3.0,	**473**	**129**	**106**	431	324	120	425	378	44	11
2.6, 2.2, 1.8)	297	56	164	495	121	227	137	283	412	482
nonlinear P-SVM	**452**	**494**	**49**	**319**	**242**	**473**	**443**	**379**	**65**	**456**
(3-layered net)	**106**	**154**	**282**	**29**	**129**	**337**	**339**	**434**	**454**	**476**
(ϵ = 0.01)	122	195	223	343	21	402	315	479	409	330
nonlinear P-SVM	**65**	**494**	**242**	**379**	**443**	**454**	**434**	**476**	**129**	**282**
(4-layered net)	**106**	**154**	**473**	**452**	**319**	**339**	**49**	**456**	308	387
(ϵ = 0.01)	283	311	139	162	236	457	229	190	16	453
nonlinear P-SVM	**65**	**494**	**242**	**379**	**443**	**476**	**434**	**454**	**106**	**129**
(4-layered net)	**282**	308	387	311	283	139	162	16	236	457
(ϵ = 0.2)	229	190	453	35	136	474	359	407	76	336
OBS	62	**49**	169	324	457	424	442	348	302	497
saliency	66	310	61	336	44	299	453	161	212	48
(3-layered net)	78	383	162	5	317	425	197	331	495	153
OBS	425	302	**443**	66	246	**49**	297	497	249	39
saliency	169	164	453	324	166	298	137	11	311	421
(4-layered net)	292	62	433	404	6	310	224	349	**476**	431
OBS	**242**	**49**	**337**	497	324	457	318	50	**154**	62
elimination	162	206	56	299	310	169	348	5	412	128
(3-layered net)	495	27	6	442	415	424	19	47	61	25
OBS	**49**	**443**	283	324	497	297	**319**	164	138	5
elimination	62	249	86	349	246	43	208	6	310	491
(4-layered net)	410	291	298	54	166	302	**476**	457	482	212
OBD	**49**	62	169	457	324	424	442	348	497	310
saliency	61	299	336	**154**	161	78	453	313	293	5
(3-layered net)	495	153	331	292	128	162	121	302	287	411
OBD	425	66	**443**	246	**49**	297	249	169	302	497
saliency	39	453	164	224	324	62	298	349	137	310
(4-layered net)	421	6	43	292	291	58	457	404	166	433

Acknowledgments. We thank Merlyn Albery-Speyer, Christoph Büscher, Raman Sanyal, Sambu Seo and Peter Wiesing for their help. This work was funded by the DFG (SFB 618) and the Anna-Geissler-Stiftung.

References

R. E. Bellman. *Adaptive Control Processes*. Princeton University Press, Princeton, NJ, 1961.

J. Bi, K. Bennett, M. Embrechts, C. Breneman, and M. Song. Dimensionality reduction via sparse support vector machines. *Journal of Machine Learning Research*, 3:1229–1243, 2003. Special Issue on Variable and Feature Selection.

B. E. Boser, I. M. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In D. Haussler, editor, *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pages 144–152, Pittsburgh, PA, 1992. ACM Press.

P. S. Bradley and O. L. Mangasarian. Feature selection via concave minimization and support vector machines. In J. Shavlik, editor, *Machine Learning Proceedings of the Fifteenth International Conference(ICML '98)*, pages 82–90, San Francisco, California, 1998. Morgan Kaufmann. `ftp://ftp.cs.wisc.edu/math-prog/tech-reports/98-03.ps.Z`.

P. S. Bradley, O. L. Mangasarian, and W. N. Street. Feature selection via mathematical programming. *INFORMS Journal on Computing*, 10(2):209–217, 1998.

C. Cortes and V. N. Vapnik. Support vector networks. *Machine Learning*, 20: 273–297, 1995.

I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3):389–422, 2002.

B. Hassibi and D. G. Stork. Second order derivatives for network pruning: Optimal brain surgeon. In J. D. Cowan S. J. Hanson and C. L. Giles, editors, *Advances in Neural Information Processing Systems 5*, pages 164–171. San Mateo, CA: Morgan Kaufmann, 1993.

S. Hochreiter and K. Obermayer. Classification, regression, and feature selection on matrix data. Technical Report 2004/2, Technische Universität Berlin, Fakultät für Elektrotechnik und Informatik, 2004a.

S. Hochreiter and K. Obermayer. Gene selection for microarray data. In B. Schölkopf, K. Tsuda, and J.-P. Vert, editors, *Kernel Methods in Computational Biology*, pages 319–355. MIT Press, 2004b.

S. Hochreiter and K. Obermayer. Sphered support vector machine. Technical report, Technische Universität Berlin, Fakultät für Elektrotechnik und Informatik, 2004c.

S. Hochreiter and J. Schmidhuber. Flat minima. *Neural Computation*, 9(1):1–42, 1997.

M. G. Kendall and A. Stuart. *The advanced theory of statistics*. Charles Griffin & Co LTD, 4 edition, 1977.

M. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal*, 37:233–243, 1991.

Y. LeCun, J. S. Denker, and S. A. Solla. Optimal brain damage. In D. S. Touretzky, editor, *Advances in Neural Information Processing Systems 2*, pages 598–605. San Mateo, CA: Morgan Kaufmann, 1990.

D. J. C. MacKay. Bayesian non-linear modelling for the 1993 energy prediction competition. In G. Heidbreder, editor, *Maximum Entropy and Bayesian Methods, Santa Barbara 1993*. Kluwer, Dordrecht, 1993.

J. E. Moody and J. Utans. Principled architecture selection strategies for neural networks: Application to corporate bond rating prediction. In J. E. Moody, S. J.

Hanson, and R. P. Lippman, editors, *Advances in Neural Information Processing Systems 4*, pages 683–690. San Mateo, CA: Morgan Kaufmann, 1992.

R. Neal. *Bayesian Learning for Neural Networks*. Springer Verlag, New York, 1996.

E. Oja. Data compression, feature extraction, and autoassociation in feedforward neural networks. In T. Kohonen, K. Mäkisara, O. Simula, and J. Kangas, editors, *Artificial Neural Networks*, volume 1, pages 737–745. Elsevier Science publishers B.V., North-Holland, 1991.

J. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods – Support Vector Learning*, pages 185–208, Cambridge, MA, 1999. MIT Press.

B. Schölkopf and A. J. Smola. *Learning with kernels – Support Vector Machines, Reglarization, Optimization, and Beyond*. MIT Press, Cambridge, 2002.

B. Schölkopf, A. J. Smola, and K.-R. Müller. Kernel principal component analysis. In W. Gerstner, A. Germond, M. Hasler, and J.-D. Nicoud, editors, *Artificial Neural Networks – ICANN'97*, pages 583–588, Berlin, 1997. Springer Lecture Notes in Computer Science, Vol. 1327.

A. J. Smola, S. Mika, B. Schölkopf, and R. C. Williamson. Regularized principal manifolds. *Journal of Machine Learning Research*, 1:179–209, 2001. http://www.jmlr.org.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, B*, 58(1):267–288, 1996.

N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method. In *Proceedings of the 37-th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377, 1999.

V. N. Vapnik. *Statistical Learning Theory*. Adaptive and learning systems for signal processing, communications, and control. Wiley, New York, 1998.

J. Weston, A. Elisseeff, B. Schölkopf, and M. Tipping. Use of the zero-norm with linear models and kernel methods. *Journal of Machine Learning Research*, 3: 1439–1461, 2003. Special Issue on Variable and Feature Selection.

J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik. Feature selection for SVMs. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems*, volume 13. MIT Press, Cambridge, MA, 2000.