SVM-based Feature Selection by Direct Objective Minimisation

Julia Neumann, Christoph Schnörr, and Gabriele Steidl

Dept. of Mathematics and Computer Science
University of Mannheim, 68131 Mannheim, Germany
http://www.cvgpr.uni-mannheim.de, http://kiwi.math.uni-mannheim.de
{jneumann,schnoerr,steidl}@uni-mannheim.de

Abstract. We propose various novel embedded approaches for (simultaneous) feature selection and classification within a general optimisation framework. In particular, we include linear and nonlinear SVMs. We apply difference of convex functions programming to solve our problems and present results for artificial and real-world data.

1 Introduction

Overview and related work. Given a pattern recognition problem as a training set of labelled feature vectors, our goal is to find a mapping that classifies the data correctly. In this context, *feature selection* aims at picking out some of the original input dimensions (*features*) (i) for performance issues by facilitating data collection and reducing storage space and classification time, (ii) to perform semantics analysis helping to understand the problem, and (iii) to improve prediction accuracy by avoiding the "curse of dimensionality" (cf. [6]).

Feature selection approaches divide into *filters* that act as a preprocessing step independently of the classifier, *wrappers* that take the classifier into account as a black box, and *embedded approaches* that simultaneously determine features and classifier during the training process (cf. [6]). In this paper, we deal with the latter method and focus on *direct objective minimisation*. Our linear classification framework is based on [4], but takes into account that the *Support Vector Machine* (SVM) provides good generalisation ability by its ℓ_2-regulariser. There exist only few papers on nonlinear classification with embedded feature selection. An approach for the quadratic 1-norm SVM was suggested in [12]. An example for a wrapper method employing a Gaussian kernel SVM error bound is [11].

Contribution. We propose a range of new embedded methods for feature selection regularising linear embedded approaches and construct feature selection methods for nonlinear SVMs. To solve the non-convex problems, we apply the general difference of convex functions (d.c.) optimisation algorithm.

Structure. In the next section, we present various extensions of the linear embedded approach proposed in [4] and consider feature selection methods in conjunction with nonlinear classification. The d.c. optimisation approach and its application to our problems is described in Sect. 3. Numerical results illustrating and evaluating various approaches are given in Sect. 4.

2 Feature Selection by Direct Objective Minimisation

Given a training set $\{(\mathbf{x}_i, y_i) \in \mathcal{X} \times \{-1, 1\} : i = 1, \ldots, n\}$ with $\mathcal{X} \subset \mathbb{R}^d$, our goal is both to find a classifier $F : \mathcal{X} \to \{-1, 1\}$ and to select features.

2.1 Linear Classification

The linear classification approaches construct two parallel bounding planes in $\mathbb{R}^d$ such that the differently labelled sets are to some extent in the two opposite half spaces determined by these planes. More precisely, one solves the minimisation problem

$$\min_{\mathbf{w}\in\mathbb{R}^d, b\in\mathbb{R}} (1-\lambda)\sum_{i=1}^{n}(1 - y_i(\mathbf{w}^T\mathbf{x}_i + b))_+ + \lambda\rho(\mathbf{w}) \tag{1}$$

with $\lambda \in [0, 1)$, regulariser ρ and $x_+ := \max(x, 0)$. Then the classifier is $F(\mathbf{x}) = \operatorname{sgn}(\mathbf{w}^T\mathbf{x} + b)$. For $\rho = 0$, the linear method (1) was proposed as *Robust Linear Programming* (RLP) by Bennett and Mangasarian [2]. Note that these authors weighted the training errors by $1/n_{\pm 1}$, where $n_{\pm 1} = |\{i : y_i = \pm 1\}|$.

In order to maximise the margin between the two parallel planes, the original SVM penalises the ℓ_2-norm $\rho(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|_2^2$. Then (1) can be solved by a convex Quadratic Program (QP).

In order to suppress features, ℓ_p-norms with $p < 2$ are used. In [4], the ℓ_1-norm (lasso penalty) $\rho(\mathbf{w}) = \|\mathbf{w}\|_1$ leads to good feature selection and classification results. Moreover, for the ℓ_1-norm, (1) can be solved by a linear program.

The feature selection can be further improved by using the so-called ℓ_0-"norm" $\|\mathbf{w}\|_0 = |\{i : w_i \neq 0\}|$ [4, 10]. Since the ℓ_0-norm is non-smooth, it was approximated in [4] by the concave functional

$$\rho(\mathbf{w}) = \mathbf{e}^T\left(\mathbf{e} - \left(e^{-\alpha|w_i|}\right)_{i=1}^{d}\right) \approx \|\mathbf{w}\|_0 \tag{2}$$

with approximation parameter $\alpha \in \mathbb{R}_+$ and $\mathbf{e} = (1, \ldots, 1)^T$. Problem (1) with penalty (2) is known as *Feature Selection concaVe* (FSV). Now the solution of (1) becomes more sophisticated and can be obtained, e.g., by the *Successive Linearization Algorithm* (SLA) as proposed in [4].

New feature selection approaches. Since the ℓ_2 penalty term leads to very good classification results while the ℓ_1 and ℓ_0 penalty terms focus on feature selection, we suggest using combinations of these terms. As common, to eliminate the absolute values in the ℓ_1-norm or in the approximate ℓ_0-norm, we introduce additional variables $v_i \geq |w_i|$ $(i = 1, \ldots, d)$ and consider $\nu\rho(\mathbf{v}) + \chi_{[-\mathbf{v},\mathbf{v}]}(\mathbf{w})$ instead of $\lambda\rho(\mathbf{w})$, where χ_C denotes the indicator function $\chi_C(x) = 0$ if $x \in C$ and $\chi_C(x) = \infty$ otherwise (cf. [7, 8]). As a result, for $\mu, \nu \in \mathbb{R}_+$, we minimise

$$f(\mathbf{w}, b, \mathbf{v}) := \frac{\mu}{n}\sum_{i=1}^{n}(1 - y_i(\mathbf{w}^T\mathbf{x}_i + b))_+ + \frac{1}{2}\|\mathbf{w}\|_2^2 + \nu\rho(\mathbf{v}) + \chi_{[-\mathbf{v},\mathbf{v}]}(\mathbf{w}) \ . \tag{3}$$

In case of the ℓ_1-norm, problem (3) can be solved by a convex QP. For the approximate ℓ_0-norm an appropriate method is presented in Sect. 3.

2.2 Nonlinear Classification

For problems which are not linearly separable a so-called *feature map* $\boldsymbol{\phi}$ which usually maps the set $\mathcal{X} \subset \mathbb{R}^d$ onto a higher dimensional space $\boldsymbol{\phi}(\mathcal{X}) \subset \mathbb{R}^{d'}$ ($d' \geq d$) is used. Then the linear approach (1) is applied in the new feature space $\boldsymbol{\phi}(\mathcal{X})$. This results in a nonlinear classification in the original space $\mathbb{R}^d$, i.e., in nonlinear separating surfaces.

Quadratic feature map. We start with the simple quadratic feature map

$$\boldsymbol{\phi} : \mathcal{X} \to \mathbb{R}^{d'} \quad , \quad \mathbf{x} \mapsto (\mathbf{x}^{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathbb{N}_0^d \ , \ 0 < \|\boldsymbol{\alpha}\|_1 \leq 2) \ ,$$

where $d' = \frac{d(d+3)}{2}$, and apply (1) in $\mathbb{R}^{d'}$ with the approximate ℓ_0-penalty (2):

$$\begin{aligned} f(\mathbf{w}, b, \mathbf{v}) :=& (1-\lambda) \sum_{i=1}^{n} (1 - y_i(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_i) + b))_+ + \lambda \mathbf{e}^T (\mathbf{e} - e^{-\alpha \mathbf{v}}) \\ &+ \sum_{i=1}^{d'} \sum_{\boldsymbol{\phi}_i(\mathbf{e}_j) \neq 0} \chi_{[-v_j, v_j]}(w_i) \qquad \longrightarrow \min_{\mathbf{w} \in \mathbb{R}^{d'}, b \in \mathbb{R}, \mathbf{v} \in \mathbb{R}^d} \ , \end{aligned} \tag{4}$$

where $\mathbf{e}_j \in \mathbb{R}^d$ denotes the j-th unit vector. We want to select features in the *original* space $\mathbb{R}^d$ due to (i)-(ii) in Sect. 1. Thus we include the appropriate indicator functions. A similar approach in [12] does not involve this idea and achieves only a feature selection in the *transformed* feature space $\mathbb{R}^{d'}$. We will refer to (4) as *quadratic FSV*. In principle, the approach can be extended to other feature maps $\boldsymbol{\phi}$, especially to other polynomial degrees.

Gaussian kernel feature map. Next we consider SVMs with the feature map related to the Gaussian kernel

$$K(\mathbf{x}, \mathbf{z}) = K_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) = e^{-\|\mathbf{x}-\mathbf{z}\|_{2,\boldsymbol{\theta}}^2 / 2\sigma^2} \tag{5}$$

with weighted ℓ_2-norm $\|\mathbf{x}\|_{2,\boldsymbol{\theta}}^2 = \sum_{k=1}^{d} \theta_k |x_k|^2$ by $K(\mathbf{x}, \mathbf{z}) = \langle \boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{z}) \rangle$ for all $\mathbf{x}, \mathbf{z} \in \mathcal{X}$. We apply the usual SVM classifier. For further information on nonlinear SVMs see, e.g., [9]. Direct feature selection, i.e., the setting of as many θ_k to zero as possible while retaining or improving the classification ability, is a difficult problem. One possible approach is to use a wrapper as in [11]. In [5], the alignment $\hat{A}(\mathbf{K}, \mathbf{y}\mathbf{y}^T) = \mathbf{y}^T \mathbf{K} \mathbf{y} / (n \|\mathbf{K}\|_F)$ was proposed as a measure of conformance of a kernel with a learning task. Therefore, we suggest to maximise in a modified form $\mathbf{y}_n^T \mathbf{K} \mathbf{y}_n$ where $\mathbf{y}_n = (y_i / n_{y_i})_{i=1}^n$. Then, with penalty (2), we define our *kernel-target alignment approach* for feature selection as

$$f(\boldsymbol{\theta}) := -(1-\lambda) \frac{1}{2} \mathbf{y}_n^T \mathbf{K}_{\boldsymbol{\theta}} \mathbf{y}_n + \lambda \frac{1}{d} \mathbf{e}^T (\mathbf{e} - e^{-\alpha \boldsymbol{\theta}}) + \chi_{[\mathbf{0}, \mathbf{e}]}(\boldsymbol{\theta}) \longrightarrow \min_{\boldsymbol{\theta} \in \mathbb{R}^d} \ . \tag{6}$$

The scaling factors $\frac{1}{2}$, $\frac{1}{d}$ ensure that both objective terms take values in $[0, 1]$.

3 D.C. Programming and Optimisation

A robust algorithm for minimising non-convex problems is the *Difference of Convex functions Algorithm* (DCA) proposed in [7]. Its goal is to minimise a function $f : \mathbb{R}^d \to \mathbb{R} \cup \{\infty\}$ which reads

$$f(\mathbf{x}) = g(\mathbf{x}) - h(\mathbf{x}) \longrightarrow \min_{\mathbf{x}\in\mathbb{R}^d} \ , \tag{7}$$

where $g, h : \mathbb{R}^d \to \mathbb{R} \cup \{\infty\}$ are lower semi-continuous, proper convex functions cf. [8]. In the next subsections, we first introduce the DCA and then apply it to our non-convex feature selection problems.

3.1 D.C. Programming

For g as assumed above, we introduce the *domain* of g, its *conjugate function* at $\tilde{\mathbf{x}} \in \mathbb{R}^d$ and its *subdifferential* at $\mathbf{z} \in \mathbb{R}^d$ by $\operatorname{dom} g := \{\mathbf{x} \in \mathbb{R}^d : g(\mathbf{x}) < \infty\}$, $g^*(\tilde{\mathbf{x}}) := \sup_{\mathbf{x}\in\mathbb{R}^d}\{\langle \mathbf{x}, \tilde{\mathbf{x}}\rangle - g(\mathbf{x})\}$ and $\partial g(\mathbf{z}) := \{\tilde{\mathbf{x}} \in \mathbb{R}^d : g(\mathbf{x}) \geq g(\mathbf{z}) + \langle \mathbf{x} - \mathbf{z}, \tilde{\mathbf{x}}\rangle \quad \forall\, \mathbf{x} \in \mathbb{R}^d\}$, respectively. For differentiable functions we have that $\partial g(\mathbf{z}) = \{\nabla g(\mathbf{z})\}$. According to [8, Theorem 23.5], it holds

$$\partial g(\mathbf{x}) = \arg\max_{\tilde{\mathbf{x}}\in\mathbb{R}^d}\{\mathbf{x}^T\tilde{\mathbf{x}} - g^*(\tilde{\mathbf{x}})\} \ , \quad \partial g^*(\tilde{\mathbf{x}}) = \arg\max_{\mathbf{x}\in\mathbb{R}^d}\{\tilde{\mathbf{x}}^T\mathbf{x} - g(\mathbf{x})\} \ . \tag{8}$$

Further assume that $\operatorname{dom} g \subset \operatorname{dom} h$ and $\operatorname{dom} h^* \subset \operatorname{dom} g^*$. It was proved in [7] that then every limit point of the sequence $\left(\mathbf{x}^k\right)_{k\in\mathbb{N}_0}$ produced by the following algorithm is a critical point of f in (7):

Algorithm 3.1: D.C. MINIMISATION ALGORITHM (DCA)(g, h, tol)

choose $\mathbf{x}^0 \in \operatorname{dom} g$ arbitrarily
for $k \in \mathbb{N}_0$
do
- select $\tilde{\mathbf{x}}^k \in \partial h(\mathbf{x}^k)$ arbitrarily
- select $\mathbf{x}^{k+1} \in \partial g^*(\tilde{\mathbf{x}}^k)$ arbitrarily
- **if** $\min\left(\left|x_i^{k+1} - x_i^k\right|, \left|\frac{x_i^{k+1} - x_i^k}{x_i^k}\right|\right) \leq tol \quad \forall\, i = 1, \ldots, d$
  - **then return** $(\mathbf{x}^{k+1})$

We can show – but omit this point due to lack of space – that the DCA applied to a *particular* d.c. decomposition (7) of FSV coincides with the SLA.

3.2 Application to our Feature Selection Problems

The crucial point in applying the DCA is to define a suitable d.c. decomposition (7) of the objective function. The aim of this section is to propose such decompositions for our different approaches.

ℓ_2-ℓ_0-SVM. A viable d.c. decomposition for (3) with (2) reads

$$g(\mathbf{w}, b, \mathbf{v}) = \frac{\mu}{n}\sum_{i=1}^{n}(1 - y_i(\mathbf{w}^T\mathbf{x}_i + b))_+ + \frac{1}{2}\|\mathbf{w}\|_2^2 + \chi_{[-\mathbf{v},\mathbf{v}]}(\mathbf{w}) \ ,$$
$$h(\mathbf{v}) = -\nu\mathbf{e}^T(\mathbf{e} - e^{-\alpha\mathbf{v}})$$

which gives rise to a convex QP in each DCA step.

Quadratic FSV. To solve (4) we use the d.c. decomposition

$$g(\mathbf{w}, b, \mathbf{v}) = (1-\lambda)\sum_{i=1}^{n}(1 - y_i(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_i) + b))_+ + \sum_{i=1}^{d'}\sum_{\boldsymbol{\phi}_i(\mathbf{e}_j)\neq 0}\chi_{[-v_j,v_j]}(w_i) \ ,$$
$$h(\mathbf{v}) = -\lambda\mathbf{e}^T(\mathbf{e} - e^{-\alpha\mathbf{v}}) \ ,$$

which leads to a linear problem in each DCA step.

Kernel-target alignment approach. For the function defined in (6), as the kernel (5) is convex in $\boldsymbol{\theta}$, we split f as

$$g(\boldsymbol{\theta}) = \frac{1-\lambda}{2n_{+1}n_{-1}}\sum_{\substack{i,j=1\\ y_i\neq y_j}}^{n} e^{-\|\mathbf{x}_i-\mathbf{x}_j\|^2_{2,\boldsymbol{\theta}}/2\sigma^2} + \chi_{[\mathbf{0},\mathbf{e}]}(\boldsymbol{\theta}) \ ,$$
$$h(\boldsymbol{\theta}) = \frac{1-\lambda}{2}\sum_{\substack{i,j=1\\ y_i= y_j}}^{n}\frac{1}{n_{y_i}^2}e^{-\|\mathbf{x}_i-\mathbf{x}_j\|^2_{2,\boldsymbol{\theta}}/2\sigma^2} - \frac{\lambda}{d}\mathbf{e}^T(\mathbf{e} - e^{-\alpha\boldsymbol{\theta}}) \ .$$

Now h is differentiable, so applying the DCA we find the solution in the first step of iteration k as $\tilde{\boldsymbol{\theta}}^k = \nabla h(\boldsymbol{\theta}^k)$. In the second step, we are looking for $\boldsymbol{\theta}^{k+1} \in \partial g^*(\tilde{\boldsymbol{\theta}}^k) \overset{(8)}{=} \arg\max_{\boldsymbol{\theta}}\{\boldsymbol{\theta}^T\tilde{\boldsymbol{\theta}}^k - g(\boldsymbol{\theta})\}$ which leads to solving the *convex non-quadratic* problem

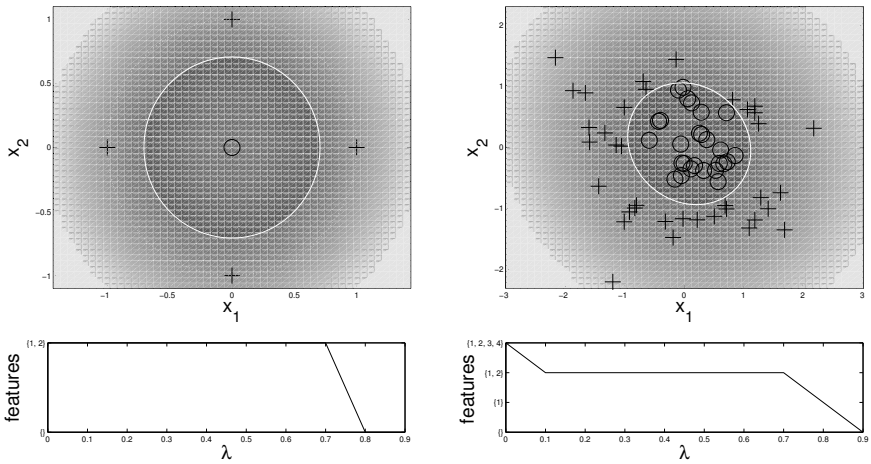
$$\min_{\boldsymbol{\theta}\in\mathbb{R}^d} \ \frac{1-\lambda}{2n_{+1}n_{-1}}\sum_{\substack{i,j=1\\ y_i\neq y_j}}^{n} e^{-\|\mathbf{x}_i-\mathbf{x}_j\|^2_{2,\boldsymbol{\theta}}/2\sigma^2} - \boldsymbol{\theta}^T\tilde{\boldsymbol{\theta}}^k \quad \text{subject to } \ \mathbf{0}\leq\boldsymbol{\theta}\leq\mathbf{e}$$

with a valid initial point $\mathbf{0} \leq \boldsymbol{\theta}^0 \leq \mathbf{e}$. We efficiently solve this problem by a penalty/barrier multiplier method with logarithmic-quadratic penalty function as proposed in [1].

4 Evaluation

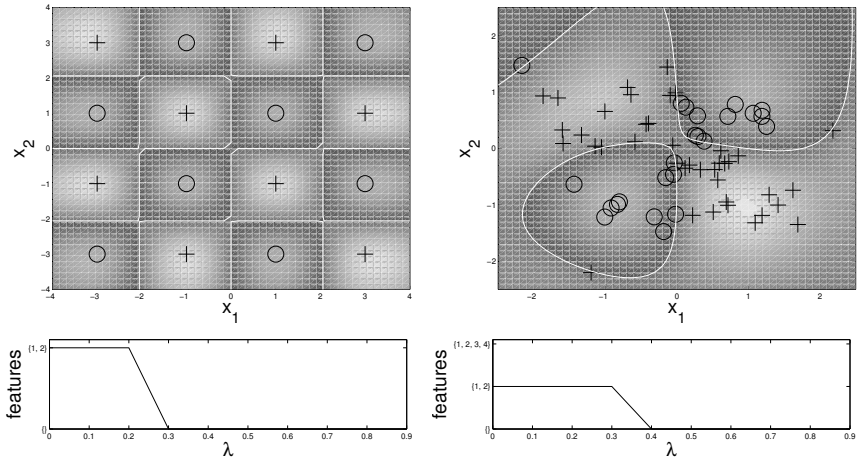
4.1 Ground Truth Experiments

In this section, we consider artificial training sets in $\mathbb{R}^2$ and $\mathbb{R}^4$ where y is a function of the first two features x_1 and x_2. The examples in Fig. 1 show that our quadratic FSV approach indeed performs feature selection and finds classification rules for quadratic, not linearly separable problems. For the non-quadratic chess board classification problems in Fig. 2, our kernel-target alignment approach performs very well, in contrast to all other feature selection approaches presented. Remarkably, the alignment functional incorporates implicit feature selection for $\lambda = 0$. In both cases, only relevant feature sets are selected as can be seen in the bottom plots.



deterministic in $\mathbb{R}^2$ | random in $\mathbb{R}^4$ (four normal random variables $x_1, \ldots, x_4$ with variances 1, 1, 1 and 2)

Fig. 1. Quadratic classification problems with $y = \operatorname{sgn}(x_1^2 + x_2^2 - 1)$. *Top:* Training points and decision boundaries (*white lines*) computed by (4) for $\lambda = 0.1$, *left:* in $\mathbb{R}^2$, *right:* projection onto selected features. *Bottom:* Features determined by (4)



deterministic ($\mathbf{x}_i \in \{-3, -1, 1, 3\}^2$) | 4 random features (same as Fig. 1 right)

Fig. 2. Chess board classification problems with $\frac{y+1}{2} = (\lfloor \frac{x_1}{2} \rfloor \bmod 2) \oplus (\lfloor \frac{x_2}{2} \rfloor \bmod 2)$. *Top:* Training points and Gaussian SVM decision boundaries (*white lines*) for $\sigma = 1$, $\lambda = 0.1$, *left:* in $\mathbb{R}^2$, *right:* projection onto selected features. *Bottom:* Features determined by (6)

4.2 Real-World Data

To test all our methods on real-world data, we use several data sets from the UCI repository [3] resumed in Table 1. We rescaled the features linearly to zero mean and unit variance and compare our approaches with RLP and FSV favoured in [4].

Table 1. Statistics for data sets used

data set	number of features d	number of samples n	class distribution n_{+1}/n_{-1}
wpbc60	32	110	41/ 69
wpbc24	32	155	28/127
liver	6	345	145/200
cleveland	13	297	160/137
ionosphere	34	351	225/126
pima	8	768	500/268
bcw	9	683	444/239

Choice of parameters. We set $\alpha = 5$ in (2) as proposed in [4] and $\sigma = \frac{\sqrt{d}}{2}$ in (5) which maximises the problems' alignment. We start the DCA with $\mathbf{v}^0 = \mathbf{1}$ for the ℓ_2-ℓ_0-SVM, FSV and quadratic FSV and with $\boldsymbol{\theta}^0 = \mathbf{e}/2$ for the kernel-target alignment approach, respectively. We stop on $\mathbf{v}$ with $tol = 10^{-5}$ resp. $tol = 10^{-3}$ for $\boldsymbol{\theta}$. We retain one half of each run's cross-validation training set for parameter selection. The parameters are chosen to minimise the validation error from $\ln \mu \in \{0, \ldots, 10\}, \ln \nu \in \{-5, \ldots, 5\}, \lambda \in \{0.05, 0.1, 0.2, \ldots, 0.9, 0.95\}$ for (quadratic) FSV and $\lambda \in \{0, 0.1, \ldots, 0.9\}$ for the kernel-target alignment approach. In case of equal validation error, we choose the larger values for (ν, μ) resp. λ. In the same manner, the SVM weight parameter λ is chosen according to the smallest $\frac{1-\lambda}{\lambda} \in \{e^{-5}, e^{-4}, \ldots, e^5\}$ independently of the selected features.

The results are summarised in Table 2 where the number of features is determined as $|\{j = 1, \ldots, d : |v_j| > 10^{-8}\}|$ resp. $|\{j = 1, \ldots, d : |\theta_j| > 10^{-2}\}|$. It is clear that all proposed approaches perform feature selection: linear FSV discards most features followed by the kernel-target alignment approach and then the ℓ_2-ℓ_0-SVM, then the ℓ_2-ℓ_1-SVM. In addition, for all approaches the test error is often smaller than for RLP. The quadratic FSV performs well mainly for special problems (e.g., 'liver' and 'ionosphere'), but the classification is good in general for all other approaches.

Table 2. Feature selection and classification tenfold cross-validation performance (average number of features, average test error [%]); bold numbers indicate lowest errors

	linear classification								nonlinear classification			
	RLP		FSV		ℓ_2-ℓ_1-SVM		ℓ_2-ℓ_0-SVM		quad. FSV		k.-t. align.	
data set	dim.	err	dim.	err	dim.	err	dim.	err	dim.	err	dim.	err
wpbc60	32.0	40.9	0.4	36.4	12.4	**35.5**	13.4	37.3	3.2	37.3	3.9	**35.5**
wpbc24	32.0	27.7	0.0	18.1	12.6	**17.4**	2.9	18.1	0.0	18.1	1.9	18.1
liver	6.0	**31.9**	2.1	36.2	6.0	35.1	5.0	34.2	3.2	**32.5**	2.5	35.4
cleveland	13.0	**16.2**	1.8	23.2	9.9	**16.5**	8.2	**16.5**	9.2	30.3	3.2	23.6
ionosphere	33.0	13.4	2.3	21.7	24.8	13.4	14.0	15.7	32.9	10.8	6.6	**7.7**
pima	8.0	**22.5**	0.7	28.9	6.6	25.1	6.1	24.7	4.7	29.9	1.6	25.7
bcw	9.0	3.4	2.4	4.8	8.7	**3.2**	7.9	**3.1**	5.4	9.4	2.8	4.2

5 Summary and Conclusion

We proposed several novel methods that extend existing linear embedded feature selection approaches towards better generalisation ability by improved regularisation and constructed feature selection methods in connection with nonlinear classifiers. In order to apply the DCA, we found appropriate splittings of our non-convex objective functions. In the experiments with real data, effective feature selection was always carried out in conjunction with a small classification error. So direct objective minimisation feature selection is profitable and viable for different types of classifiers. In higher dimensions, the curse of dimensionality affects the classification error even more such that our methods will become more important here. A further evaluation of high-dimensional problems as well as the incorporation of other feature maps is future work.

Acknowledgements. This work was funded by the DFG, Grant Schn 457/5.

References

1. A. Ben-Tal and M. Zibulevsky. Penalty/barrier multiplier methods for convex programming problems. *SIAM Journal on Optimization*, 7(2):347–366, May 1997.
2. K. P. Bennett and O. L. Mangasarian. Robust linear programming discrimination of two linearly inseparable sets. *Optimization Methods and Software*, 1:23–34, 1992.
3. C. L. Blake and C. J. Merz. UCI repository of machine learning databases, 1998.
4. P. S. Bradley and O. L. Mangasarian. Feature selection via concave minimization and support vector machines. In *Proceedings of the 15th International Conference on Machine Learning*, pages 82–90, San Francisco, CA, USA, 1998. Morgan Kaufmann.
5. N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. Kandola. On kernel-target alignment. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 367–373. MIT Press, Cambridge, MA, USA, 2002.
6. I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, Mar. 2003.
7. T. Pham Dinh and L. T. Hoai An. A d.c. optimization algorithm for solving the trust-region subproblem. *SIAM Journal on Optimization*, 8(2):476–505, May 1998.
8. R. T. Rockafellar. *Convex Analysis*. Princeton University press, Princeton, NJ, USA, 1970.
9. B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, USA, 2002.
10. 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, Mar. 2003.
11. 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 13*, pages 668–674. MIT Press, Cambridge, MA, USA, 2001.
12. J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani. 1-norm support vector machines. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, USA, 2004.