Combining a Filter Method with SVMs

Thomas Navin Lal, Olivier Chapelle, and Bernhard Schölkopf

Max-Planck-Institute for Biological Cybernetics, Tübingen, Germany.
{navin|olivier.chapelle|bs}@tuebingen.mpg.de

Summary. Our goal for the competition was to evaluate the usefulness of simple machine learning techniques. We decided to use the correlation criteria (see Chapter **??**) as a feature selection method and Support Vector Machines (see Chapter **??**) for the classification part. Here we explain how we chose the regularization parameter C of the SVM, how we determined the kernel parameter σ and how we estimated the number of features used for each data set. All analyzes were carried out on the training sets of the competition data. We choose the data set ARCENE as an example to explain the approach step by step.
In our view the point of this competition was the construction of a well performing classifier rather than the systematic analysis of a specific approach. This is why our search for the best classifier was only guided by the described methods and that we deviated from the road map at several occasions.
All calculations were done with the software Spider [2004].

1 The Parameters σ and C of the SVM

For numerical reasons every data point was normalized such that the average l_2-norm is 1: let $\{\mathbf{x}_k, k = 1, ..., m\}$ be the set of training examples. Every $\mathbf{x}$ was divided by $(\frac{1}{m}\sum_k \|\mathbf{x}_k\|^2)^{\frac{1}{2}}$.
For the data sets ARCENE, DEXTER, GISETTE and MADELON a hard margin SVM was calculated. For the unbalanced data set DOROTHEA a soft margin SVM was chosen and the regularization parameter C was obtained by cross validation prior to feature selection. An example of the cross validation error estimates for the data set ARCENE can be found in Figure 1. Furthermore, we used a class specific 2-norm penalty by adding a ridge to the kernel matrix (Brown et al. [2000]): Let r_1 be the fraction of positive examples in the training data and let r_{-1} be the fraction of negative examples. For each of the two classes we added a different ridge to the kernel matrix k: for positive examples x_i we set $k_{ii} \rightarrow k_{ii} + r_1$ and for negative examples we set $k_{ii} \rightarrow k_{ii} + r_{-1}$. Adding class specific ridges to the diagonal of the kernel matrix is equivalent to choosing two different values of C for the different classes (e.g. Schmidt

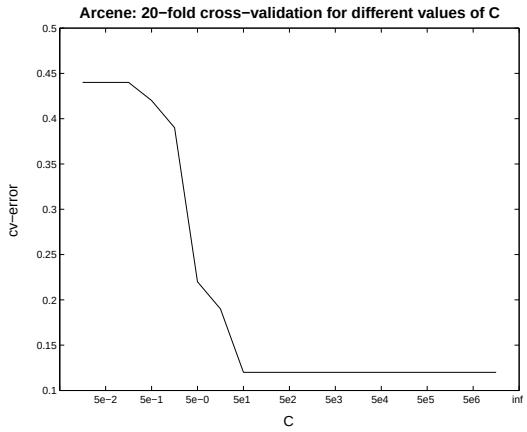

Fig. 1. For each value of C (x-axis) the 20-fold cross-validation error (y-axis) is plotted using a linear SVM on data set ARCENE.

and Gish [1996], Schölkopf and Smola [2002]).

For the data sets DOROTHEA, GISETTE and MADELON we chose a Gaussian kernel. Prior to feature selection, the kernel parameter σ was found by a heuristic: for each k let t_k denote the distance of point x_k to the set formed by the points of the other class. The value of σ was set to the mean of the t_k values[1]. For the remaining two data sets ARCENE and DEXTER we used a linear kernel.

2 Feature Ranking

The features were ranked according to their correlation coefficients (see Chapter **??**): For a set $T = \{\mathbf{t}_1, ..., \mathbf{t}_m\} \subset \mathbb{R}^n$ define the mean $\mu_i(T) = \frac{1}{m}\sum_{k=1}^{m} t_{k,i}$ and the variance $V_i(T) = \frac{1}{m}\sum_{k=1}^{m}(t_{k,i} - \mu_i(T))^2$ $(i = 1, ..., n)$. The score R_i of feature i is then given by:

$$R_i(X) = \frac{(\mu_j(X^+) - \mu_j(X^-))^2}{V_j(X^+) + V_j(X^-)},$$

with $X^+ := \{\mathbf{x}_k \in X \mid y_k = 1\}$ and X^- similarly. From Figure 2 it can be seen that only few of the 10000 features of the data set ARCENE show a

[1] In later steps we use an SVM with this σ in a cross validation scheme for further model selection. The calculation of σ involves all labels. Thus, when we test a trained model on a cross validation test set, we make a systematic error, because the label information of the test set is contained in σ. However, we find that the value for σ is not affected much when the test data is removed before the calculation.

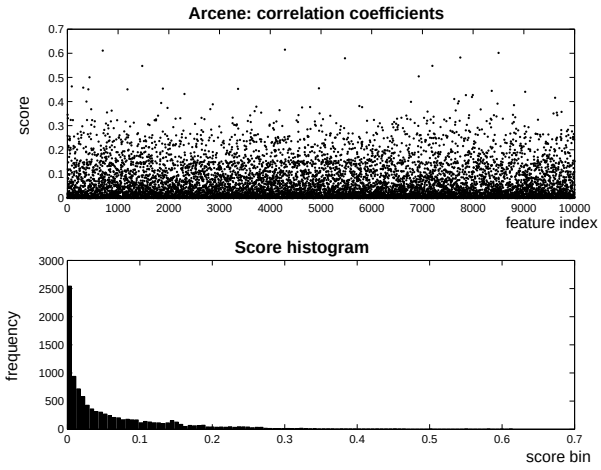

Fig. 2. The upper plot shows the correlation coefficient (y-axis) for each feature (x-axis) of the data set ARCENE. The lower part of the figure is a histogram of the correlation scores for the same data set. Please note that only few features show a high correlation with the labels.

high correlation with the labels. However, it is not obvious how many features should be used for classification.

3 Number of Features

The list of ranked features provides an estimate of how valuable a feature is for a given classification task. We are interested in the expected risk of an SVM for any given number N of best features. Provided with these values, we could choose the best number of features, i.e. the number N that minimizes the expected risk.

The expected risk could be estimated by ranking the features using the complete training set. In a second step a cross-validation error estimation can be applied for every number n of best features. However, this approach bears the risk of overfitting since all data is used during the ranking procedure. To avoid this drawback, we proceed as follows:

For given number N of best features we approximate the expected risk using a ten-fold cross-validation scheme (see Figure 3): ten times, the training data are split into a training set which forms 90% of the data and a test set forming the remaining 10%. The training data are split, such that the union of the test sets forms the training data (partition). A training set - test set - pair is called a fold. For each fold we proceed as follows:

The features are ranked based on the training set. For a given N we restrict the training and the test set examples to the best N ranked features. An SVM

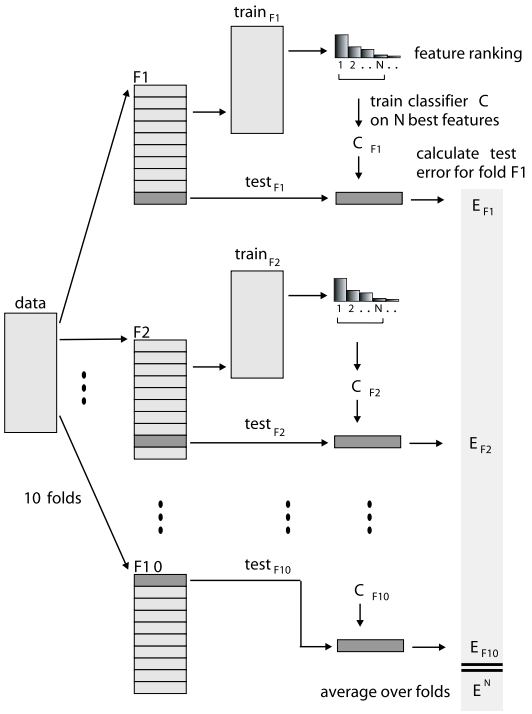

Fig. 3. This plot describes the process of calculating an estimate of the expected risk when only using the best N ranked features for classification. First the data is split into 10 train-test folds. The features are ranked on each training set, a classifier is trained using the best N features only and tested on the corresponding test set. The 10 test errors are averaged. Please note that the set of features used by the classifier might vary. Please see also Figure 4: every point of the plot is one error estimation for a specific N (from Lal et al. [2004]) .

is trained on the restricted training set and then tested on the restricted test set of the fold. For each fold we obtain a test error - these errors are averaged over the folds. As a results we get an estimate of the expected risk for the best N features. We repeat this procedure for different values of N.

Figure 4 shows the best N features versus the 10-fold cross-validation errors for the data set ARCENE. Based on this result we used the best 4700 features for the competition.

Once the number N of best features is estimated all data are used, restricted to these best N features and an SVM is trained. To avoid overfitting no further adjustment of C or σ is done. The trained model is then used to predict the labels of the unseen test set examples.

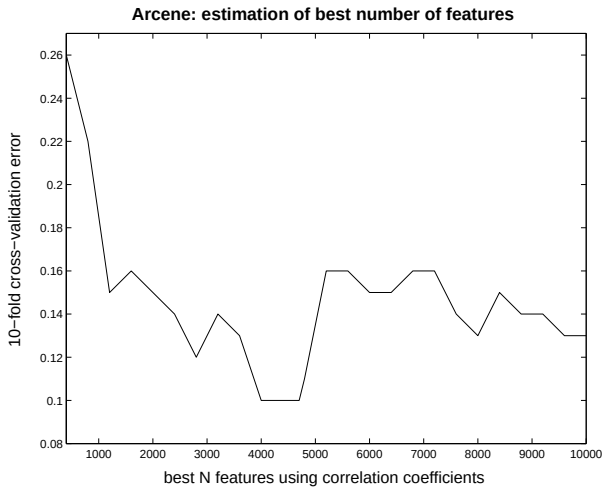

Fig. 4. Data set ARCENE: This plot shows an error estimation of the expected risk (y-axis) using only the best N features (x-axis) which are ranked according to their correlation score. The error estimates were obtained by the cross validation scheme shown in Figure 3.

Methods *FS + SVM* used for the competition

Require: data set d, kernel function k, kernel parameter σ

1: normalize data

$$\forall \mathbf{x} \in d \qquad \mathbf{x} \rightsquigarrow \mathbf{x} / \sqrt{\frac{1}{|d|} \sum_{\mathbf{y} \in d} \|\mathbf{y}\|^2}$$

2: estimate SVM parameter C via cross-validation on d using all features (for soft margin SVM only).
3: estimate the number n_0 of best features as described in Figure 3.
4: use all available data d to rank the features according to their correlation coefficients.
5: restrict d to the best n_0 ranked features.
6: train an SVM based on the restricted data d using kernel k and regularization parameter C.

---

4 Summary

We explained how a simple filter method can be combined with SVMs. More specifically we reported how we estimated the number of features to be used for classification and how the kernel parameter as well as the regularization parameter of the SVM can be determined. The performance on the competition data sets can be found in Tables 2 and 3.

Table 1. Parameter values and number of used features for the competition. A value of ∞ for C corresponds to a hard margin SVM.

	C	σ	number used of features
Arcene	∞	-	4700
Dexter	∞	-	3714
Dorothea	10 + balanced	0.115	1000
Gisette	∞	0.382	1700
Madelon	∞	0.011	20

Table 2. NIPS 2003 challenge results for the December 1^{st} data sets. Comparison of our approach *FS + SVM* to the best challenge entry.

Dec. 1^{st}	Our best challenge entry					The winning challenge entry				
Dataset	Score	BER	AUC	Feat	Probe	Score	BER	AUC	Feat	Probe
Overall	12.73	11.56	88.44	16.91	21.5	88.00	6.84	97.22	80.3	47.8
Arcene	12.73	18.20	81.80	47.0	13.6	98.18	13.30	93.48	100.0	30.0
Dexter	85.45	4.20	95.80	18.6	49.8	96.36	3.90	99.01	1.5	12.9
Dorothea	-41.82	19.68	80.32	1.0	8.9	98.18	8.54	95.92	100.0	50.0
Gisette	49.09	1.69	98.31	14.0	0.0	98.18	1.37	98.63	18.3	0.0
Madelon	-41.82	14.06	85.94	4.0	35.0	100.00	7.17	96.95	1.6	0.0

Table 3. NIPS 2003 challenge results for the December 8^{th} data sets. Comparison of our approach *FS + SVM* to the best challenge entry.

Dec. 8^{th}	Our best challenge entry					The winning challenge entry				
Dataset	Score	BER	AUC	Feat	Probe	Score	BER	AUC	Feat	Probe
Overall	31.43	8.99	91.01	20.9	17.3	71.43	6.48	97.20	80.3	47.8
Arcene	65.71	12.76	87.24	47.0	5.9	94.29	11.86	95.47	10.7	1.0
Dexter	100.00	3.30	96.70	18.6	42.1	100.00	3.30	96.70	18.57	42.1
Dorothea	-42.86	16.34	83.66	1.0	3.2	97.14	8.61	95.92	100.0	50.0
Gisette	82.86	1.31	98.69	34.0	0.2	97.14	1.35	98.71	18.3	0.0
Madelon	-48.57	11.22	88.78	4.0	35.0	94.29	7.11	96.95	1.6	0.0

Acknowledgements

This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. T.N.L. was supported by a grant from the Studienstiftung des deutschen Volkes.

References

M.P.S. Brown, W. N. Grundy, D. Lin, N. Cristianini, C.W. Sugnet, T. S Furey, M. Ares Jr., and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *PNAS*, 97(1):262–267, 2000.

T.N. Lal, M. Schröder, T. Hinterberger, J. Weston, M. Bogdan, N. Birbaumer, and B. Schölkopf. Support Vector Channel Selection in BCI. *IEEE Transactions on Biomedical Engineering. Special Issue on Brain-Computer Interfaces*, 51(6):1003–1010, June 2004.

M. Schmidt and H. Gish. Speaker Identification via Support Vector Classifiers]. In *Proceedings ICASSP'96*, pages 105–108, Atlanta, GA, 1996.

B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, USA, 2002.

Spider. Machine Learning Toolbox http://www.kyb.tuebingen.mpg.de/bs/people/spider/, 2004.