Tutz, Binder:

Localized Regression

Sonderforschungsbereich 386, Paper 378 (2004)

Online unter: http://epub.ub.uni-muenchen.de/

Localized Classification

Gerhard Tutz
Institut für Statistik, Ludwig-Maximilians-Universität München,
Akademiestr. 1, D–80799 München, Germany

Harald Binder
Klinik für Psychiatrie und Psychotherapie, Universität Regensburg,
Universitätsstr. 84, D–93042 Regensburg, Germany

Abstract:

The main problem with localized discriminant techniques is the curse of dimensionality, which seems to restrict their use to the case of few variables. This restriction does not hold if localization is combined with a reduction of dimension. In particular it is shown that localization yields powerful classifiers even in higher dimensions if localization is combined with locally adaptive selection of predictors. A robust localized logistic regression (LLR) method is developed for which all tuning parameters are chosen data-adaptively. In an extended simulation study we evaluate the potential of the proposed procedure for various types of data and compare it to other classification procedures. In addition we demonstrate that automatic choice of localization, predictor selection and penalty parameters based on cross validation is working well. Finally the method is applied to real data sets and its real world performance is compared to alternative procedures.

Key words: Local logistic regression, discrimination, data adaptive tuning parameters, selection of predictors, localized discrimination

1 Introduction

There are various ways of structuring the world of classification and discrimination by distinguishing between parametric and nonparametric approaches, Bayesian or non-Bayesian approaches or linear and nonlinear methods. A different type of structuring is based on distinguishing between global classification rules and observation specific rules.

In *global classification rules* a set of parameters is estimated from the total sample of observations. Classification of individual observations is obtained by transformations of the predictor values which are based on these estimated parameters. In this sense Fisher's discriminant analysis, logistic discrimination, quadratic discriminant analysis, classification trees and neural networks are global classifiers.

In *observation specific* approaches the classifier is adapted to each observation. The k-nearest-neighbourhood classifier (Fix and Hodges, 1951) finds the k observations from the training set which are closest to the present predictor value and uses a majority vote among the k neighbours. Although following a common principle, for each observation to be classified a new classification rule is computed. The same principle, computation of a specific rule for a specific observation, is used in localized estimation where a model is fit locally at a given predictor value. Observation specific rules usually are memory-based, instead of parameters the total sample is kept in the memory.

In the present paper the focus is on observation specific approaches by using localization. The technique may be applied to any global classifier. By using a localized version it turns into an observation specific approach. In particular we will consider local versions of logistic discrimination. Local fitting of binary regression models has been investigated carefully by Fan and Gijbels (1996) and Loader (1999). Loader also investigated the use in discrimination. The main problem in using localizing techniques is the curse of dimensionality (Bellman, 1961), see also Hastie et al. (2001). Since in high dimensions local estimates are hardly local localization seems to work properly only for few dimensions. When considering local logistic regression Loader (1999, Chapter 8) applied the method to examples with two covariates, in one case, Fisher's iris data set, he used four variables. The basic idea how to use localization successfully is to combine it with dimension reduction. A strong tool for dimension reduction is the selection of relevant predictors. Even for global classifiers performance often improves if only a small subset of informative predictors is used instead of the whole set of predictors. It is to be assumed that this also works locally. Moreover, for different values different predictors may carry the relevant information. Thus in the following dimension reduction is obtained by locally adaptive selection of predictors. For alternative approaches of local dimension reduction see Shaal et al. (1998), Hastie and Tibshirani (1996). In contrast to these approaches variable selection has the advantage that one obtains information about the relevance of variables even if the selection is performed locally. In statistical applications the user is often interested which variables are relevant and have to be collected in the future.

Localization of a global classifier in certain aspects is similar to boosting where a global procedure is used repeatedly with different weights on observations. In the same way as boosting (Breiman, 1999; Friedman et al. 2000; Friedman, 2001) (often) improves simple global learners, localization in combination with appropriate dimension reduction should be able to improve global learners. In the following this is demonstrated for localized versions of logistic discrimination.

2 Localized classification

Let (x_i, y_i), $i = 1, \ldots, n_L$, denote the training set with $x_i' = (x_{i1}, \ldots, x_{ip})$ denoting measurements on p variables and $y_i \in \{1, \ldots, k\}$ representing class membership. The objective is to predict the class membership of an observation with measurement x by using x and the information from the training set. We restrict consideration to the two class situation ($k = 2$) where for simplicity y_i can take values 0 and 1. In the following we focus on logistic discrimination, although alternative parametric approaches like linear or quadratic discrimination could be applied in a similar way.

Successful use of localization and selection of predictors depends on tuning parameters which have to be chosen. These parameters which for example determine the amount of localization and thresholds for predictor selection are introduced in the following. They are considered as flexible parameters that are chosen data-adaptively.

2.1 Localized logistic regression

The parametric model that is localized is the well known logistic regression model

$$\log\left\{\frac{P(y_i = 1|x_i)}{1 - P(y_i = 1|x_i)}\right\} = z_i'\beta$$

where β is a parameter vector of length $m + 1$ and z_i is a design vector built from x_i. For linear logistic discrimination $z_i' = (1, x_i')$ and for quadratic logistic discrimination $z_i' = (1, x_i', x_{i1}^2, \ldots, x_{ip}^2)$ is used. Interaction terms of the form $x_{ij}x_{kl}$ are deliberately left out to keep the number of parameters in β small.

Local versions of the model are obtained by introducing weights into the (log-)likelihood. For target value x the weighted log-likelihood is given by

$$l_x(\beta) = \sum_i \left(y_i \log \pi(x_i) + (1 - y_i)\log(1 - \pi(x_i))\right) w_k(z, z_i) \qquad (1)$$

where $\pi(x_i) = P(y_i = 1|x_i)$ and z, z_i are the predictor values connected to x, x_i, i.e. $z' = (1, x')$, $z_i = (1, x_i')$ in the linear case and $z' = (1, x', x_1^2, \ldots, x_p^2)$, $z_i' = (1, x_i', x_{i1}^2, \ldots, x_{ip}^2)$ in the quadratic case. The locally adaptive weights $w_k(z, z_i)$ are chosen to depend on the (Euclidian) distance between the (transformed) target value z and the (transformed) observation z_i and a kernel window

$$w_k(z, z_i) = K\left(\frac{||z - z_i||}{d_k(z)}\right)$$

where the kernel width parameter $d_k(z)$ is locally adaptive to the density at z. It is chosen as the distance to the kth nearest neighbour $z_{(k)}$ of z, i.e.

$$d_k(z) = ||z - z_{(k)}||.$$

The order k of the nearest neighbourhood is considered as a flexible parameter of the algorithm.

In contrast to the use of localizing in smoothing, here the weighting scheme is based on the distance in the predictor space instead of the distances in the space of the original variables. While the spaces are identical in the linear case the distances differ for the quadratic case. The performance seems not to depend strongly on the type of weighting. This might be due to the strong relation between the distances in predictor and variable space. By partitioning the vectors into $z' = (1, x', \tilde{x}')$, $z_i' = (1, x_i', \tilde{x}_i')$ one obtains the simple relation $||z - z_i||^2 = ||x - x_i||^2 + ||\tilde{x} - \tilde{x}_i||^2$ which implies that the distance between z and z_i increases with the distance between x and x_i. The use of distances in the predictor space is preferred because it is easier to handle.

Various kernel functions K can be used. For our investigations we mainly used the Gaussian kernel

$$K_G(x) = \exp(-x^2)$$

and the tricube kernel

$$K_T(x) = \begin{cases} (1 - |x|^3)^3 & \text{for}|x| < 1 \\ 0 & \text{otherwise} \end{cases} .$$

We found parameter estimation to be more stable with the Gaussian kernel because even points far away from x receive non-zero weight. On the other hand for localization methods the tricube kernel has the advantage that in estimation only points in the neighbourhood are included since all other points receive weight zero. The tricube kernel in effect uses only the k nearest neighbours of z. Because of the computational advantages, in applications the tricube kernel is used.

Parameter estimation is performed by solving the local score equation $s_{x,k}(\beta) = 0$ by iterative Fisher scoring of the form

$$\hat{\beta}_x^{(s+1)} = \hat{\beta}_x^{(s)} + F_{x,k}(\hat{\beta}_x^{(s)})^{-1} s_{x,k}(\hat{\beta}_x^{(s)}) \tag{2}$$

where $s_{x,k}(\beta) = \partial l_x / \partial \beta$ is the local score function which for the logistic model has the simple form

$$s_{x,k}(\beta) = \sum_i w_k(z, z_i) z_i (y_i - \pi_i(\beta))$$

with $\pi_i(\beta)$ denoting the response probability evaluated at β and $F_{x,k} = E(-\partial^2 l_x / \partial \beta \partial \beta')$ denoting the weighted Fisher matrix

$$F_{x,k} = \sum_i w_k(z, z_i) z_i z_i' \frac{\partial h(\eta_i)}{\partial \eta}$$

whith $\eta_i = z_i'\beta$ and $h(x) = \frac{\exp(x)}{1+\exp(x)}$ being the response function of the logistic regression model. The dependence of the parameter estimates on the target value shows in the notation "$\hat{\beta}_x$". For the asymptotic behaviour of local estimates see Fan and Gijbels (1996).

2.2 Local reduction of dimensions by selection of predictors

A second step in localizing the logistic regression model is to do local selection of predictors. This is based on the assumption that not all predictors are equally informative on class membership throughout the space spanned by all predictors.

Kohavi and John (1998) distinguish between the filter and the wrapper approach for selection of predictors. Algorithms from the first class judge the usefulness of predictors for classification based on the algorithm used for classification whereas in the wrapper approach predictors are judged irrespectively of the classification algorithm. In our case the filter approach corresponds to evaluating the relevance of the coefficients of the fitted local model. We considered one-step selection by including only predictors with standardized coefficients above a certain threshold as well as a stepwise exclusion of predictors with small standardized cofficients and re-estimation in each step. Since the stepwise procedure has not been superior to the one-step selection we decided in favor of the former, computationally more attractive, alternative. From the class of wrapper algorithms, that judge predictors irrespectively of the algorithm used for classification, we employed a weighted variant of the "Relief" algorithm (Kira and Rendell, 1992) because of its low computational complexity. A sample of points is taken and for each predictor separately the distance to the nearest point of the same class and to the nearest point of the different class is calculated. Predictors that have a larger mean difference between these two distances are judged to be more relevant. As we could not find an overall improvement when using this procedure for selection of predictors instead of the one-step coefficient-based method we are going to employ the latter approach.

In methods of dimension reduction one often distinguishes between unsupervised and supervised techniques (e.g. Bishop, 1995) where the former ignores the response values in the sample whereas the latter explicitly uses them. Using the parameter estimates of an initial local classification model for selection of predictors has the advantage that the response is taken into account for dimension reduction. This is in contrast to approaches for local dimension reduction like locally weighted factor analysis or locally weighted principal component analysis that ignore the response (see e.g. Schaal et al. 1998).

In the proposed one-step selection procedure the relevance of predictors is determined by a simple variant of Wald tests. For the local estimates $\hat{\beta}_x$ at target value x the variance may be approximated by

$$\hat{\text{cov}}(\hat{\beta}_x) = F_{x,k}(\hat{\beta}_x)^{-1}.$$

(see Kauermann and Tutz, 2000). This approximation is used to select predictors based on the studentized value

$$c_{x,k}(\hat{\beta}_{x,j}) = \frac{|\hat{\beta}_{x,j}|}{\sqrt{\hat{\text{var}}(\hat{\beta}_{x,j})}} \quad , \; j = 1, \ldots, m,$$

where $\beta'_x = (\beta_{x,0}, \beta_{x,1}, \ldots, \beta_{x,m})$. In a single step those predictors are selected for which $c_{x,k}(\hat{\beta}_j)$ exceeds a value c_β. $c_{x,k}(\hat{\beta})$ is a localized version of the Wald statistic for testing the null hypothesis $\beta_j = 0$ locally. With c_β one obtains the second flexible parameter of the algorithm. In the case of linear logistic discrimination where z_i is given by $(1, x_{i1}, \ldots, x_{ip})$ predictor selection refers to the original variables $x_1, \ldots, x_p$ whereas in quadratic logistic discrimination where z_i is given by $(1, x_{i1}, \ldots, x_{ip}, x_{i1}^2, \ldots, x_{ip}^2)$ predictor selection refers to the extended set of variables $x_1, \ldots, x_p, x_1^2, \ldots, x_p^2$.

When the number of predictors has been reduced the weights $w(z, z_i)$ are recalculated for the subspace spanned by the selected predictors and the estimation is performed for the reduced β-vector of the final local model. Prediction for target value x then is based on the reduced and re-estimated model.

As all predictors are judged separately by the statistic $c_{x,k}(\hat{\beta})$ predictor selection resembles a multiple test situation. Instead of a threshold c_β which is used for single predictors an overall level of significance α could be used. To guarantee that the overall level holds the significance level for the judgment of a single predictor has to be adjusted. We investigated the use of a sequentially rejective Bonferroni test [19], but could not find any improvement in performance over the simpler procedure. For that reason the latter is retained. One might also argue that selection of predictors should not be guided by a flexible parameter c_β but by a fixed level of significance (e.g. $\alpha = 0.05$), because an additional flexible parameter of the algorithm could increase the danger of overfitting. We compared the performance of a fixed level procedure to that with a flexible parameter c_β which is chosen data-adaptively. We found that having a flexible parameter which is optimized automatically by cross validation techniques leads to superior performance compared to a fixed level of significance procedure.

2.3 Computational optimization

The basic algorithm needs some refinements. A problem that occurs is that parameters tend towards infinity as a result of local complete or quasi-complete separability (Albert and Anderson). In such cases iterations are stopped at the point where $\pi_i(\hat{\beta})$ gets too close to 0 or 1. Then the estimates from that stage are used, but we refrain from using the variance-estimates and so we do no covariate selection.

Additional numerical problems are due to (local) collinearities of the covariates. These often cause instability and large estimates of parameters. To avoid these problems a penalization of the parameter estimates is introduced. The likelihood (1) is modified by a penalty term. The resulting penalized weighted log-likelihood is

$$l(\beta) = \sum_i \left(y_i \log \pi(x_i) + (1 - y_i) \log(1 - \pi(x_i))\right) w_k(z, z_i) - \lambda \beta' P \beta. \quad (3)$$

where P is the penalization matrix and λ determines the strength of the penalization. Setting λ to 0 would result in the un-penalized likelihood (1). For the penalty matrix P a simple identity-matrix is used. This leads to a penalization of β_j^2 and is similar to logistic ridge regression (Le Cessie and van Houwelingen, 1992). So λ is the third flexible parameter of the model.

The modifed expression for the penalized weighted local score function and Fisher matrix are

$$s_{x,k}(\beta) = \sum_i w_k(z, z_i) z_i (y_i - \pi_i(\beta)) - 2\lambda P \beta$$

and

$$F_{x,k} = \sum_i w_k(z, z_i) z_i z_i' + 2\lambda P.$$

Special attention has to be given to the intercept parameter β_0, because in logistic discrimination it includes the prior probabilities of class membership. This effect is lost for local estimates. For localized estimates β_0 reflects the predicted local class membership probability. When using penalization this special parameter cannot be penalized, because penalization would force it towards zero, resulting in a 0.5 local class membership probability that is not always optimal. On the other hand without penalization convergence problems arise. Le Cessie and van Houwlingen (1992) suggest to center the predictors (in addition to the standardization that is necessary because of the penalization) and to set the intercept to a fixed value. As for localized procedures the weights have to be taken into account the predictors here are centered and standardized in a weighted way and the intercept is kept fixed at the value of the transformed local class membership proportion.

For given measurement x and fixed parameters of the algorithm k, c_β and λ the estimation and prediction procedure may be summarized into:

1. Determine $d_k(z)$; calculate weights $w_k(z, z_i) = K(\frac{||z - z_i||}{d_k(z)})$.
2. Use iterative Fisher scoring with penalized weighted score function and Fisher matrix (with penalty λ) to determine $\hat{\beta}$.
3. If Fisher scoring converges: Use a subset of predictors where $c_{x,k}(\hat{\beta}_j) > c_\beta$, re-calculate $w_k(z, z_i)$ for that subspace and repeat the Fisher scoring with that subset of predictors.
4. Use the (reduced) model to predict class for x.

In order to obtain an applicable algorithm that does not suffer from ad hoc choices a fully automatic choice of the parameters of the algorithm based on cross validation is suggested. The parameters of localized logistic regression are k (index of the neighbour determining the window size), c_β (cutoff for predictor selection) and λ (penalty on parameters). Cross validation is performed by minimizing the error rate in dependence on the parameters k, c_β and λ.

2.4 Relevance of variables

Users of classifiers are not only interested in the performance of classifiers in terms of misclassification rates. Often they want to know which variables are relevant and and have to be collected in the future. An advantage of simple parametric classification like Fisher's linear discriminant analysis or (global) logistic discrimination is that the relevance of variables may be evaluated by considering the parameter estimates. For nonparametric approaches or advanced procedures like boosting the impact of variables is much harder to evaluate. For approaches in boosting see Friedman (2001).

In the case of linear localizing the approach presented here is explicitly based on variable selection but in a localized way. The underlying assumption is that different variables are relevant at different points in predictor space. Nevertheless based on the distribution of predictor values (or their empirical equivalent, the data $x_1, \ldots, x_{n_L}$) one might construct global measures for the relevance of covariates. By considering

$$I_j(x) = \begin{cases} 1 & \text{if variable } x_j \text{ is selected for prediction of } x \\ 0 & \text{otherwise} \end{cases}$$

one obtains the simple relevance score

$$r_j = \frac{1}{n_T} \sum_{i=1}^{n_T} I_j(x_i)$$

where n_T is the number of points $x \in \{x_1, \ldots, n_T\}$ for which a prediction is wanted. This measure reflects how often variable x_j is considered to be relevant in the observed predictor space. Instead of single variables one might also consider the relevance of combinations or subsets of variables by defining $I_S(x)$ as the indicator function for selection of the subset $S \subset \{x_1, \ldots, x_p\}$.

3 Simulation study

In the following we will compare the localized logistic regression (LLR) algorithm introduced above to several other procedures for classification. We use a linear version of LLR (denoted by lLLR) and a quadratic version (denoted by qLLR). The following procedures are used for comparison:

- LDA: Linear discriminant analysis.
- NNet: Single-hidden-layer neural networks with five units in the hidden layer. 20 networks are trained with different starting values and classification is done by committee voting (as suggested e.g. by Venables and Ripley, 1999).
- 1-NN: 1-nearest-neighbourhood classification (see e.g. Fix and Hodges, 1951).
- 10-NN: 10-nearest-neighbourhood classification.
- Tree: Classification trees (Breiman et al., 1984; Ripley, 1996). Tree size was determined by 10-fold cross-validation.
- Bag: Bagging with classification trees (Breiman, 1996).
- RF: Random forests (Breiman, 2001, 2002).

These procedures have been chosen because they represent a mix of linear, partition-based and model-free methods. So it will be instructive to see in which situations LLR performs similar to which methods. The implementations used are those from the statistical environment R (Ihaka and Gentleman, 1996) and we used the standard settings for all procedures.

A threshold of parameters k, c_β and λ that leads to optimal cross validation scores is obtained by a search on a 3-dimensional grid. Optimization of these parameters is done here in a global way, i.e. cross validation is used to find one set of parameters that is going to be used for all predictions based on that set of training data. This global optimization implies that the same amount of localization, predictor selection and penalization is needed everywhere in predictor space. To evaluate this assumption we employed local optimization based on a weighted version of Akaike's information criterion (see e.g. Loader, 1999). This local selection of parameters led to better results in cases where LLR with global optimization did not work very well (e.g. example HT4), but worsened performance in several examples with simple structure. Given the additional computational burden that results from local optimization at the time of prediction global optimization is retained.

In situations where parameter search on a 3-dimensional grid is computationally not feasible procedures that make assumptions on the cross validation score as a function of the three parameters and optimize them based on these assumptions can be employed. We investigated a procedure that uses quadratic approximations (Powell, 2002)and found that it worked well, i.e. in most examples it produced only slightly inferior results compared to the grid evaluation.

3.1 Example data and results

We used a variety of types of data following Friedman (1994) and Hastie and Tibshirani (1996). The examples vary with respect to importance of variables, number of noise variables, distribution of variables per class and the shape of the class regions. Presentation distinguishes between normally distributed classes, examples with non-overlapping class regions and examples with fractioned class regions. For each example 50 replications have been done.

The size of the learning data was set to $n_L = 200$ and the number of observations in the test data was set to $n_T = 1000$ for most of the examples. Both are equally divided between the two classes used in each example.

3.1.1 Classes with covariates from multivariate normal distributions

We start with a simple example (HT1) which is equivalent to example 1 of Hastie and Tibshirani (1996) where two covariates are drawn from a normal distribution with variance given by $\text{var}(x_1) = 1$, $\text{var}(x_2) = 2$ and correlation 0.75. The mean of the two classes is separated by two units

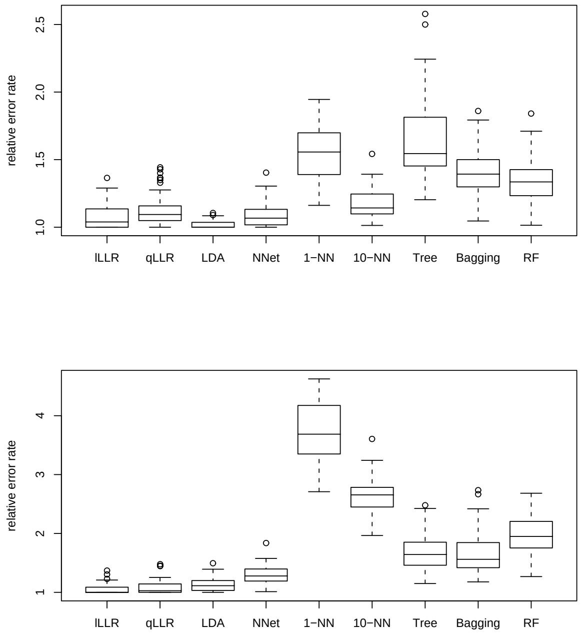

Figure 1: Relative error rates for two classes with covariates with a 2-dimensional normal distribution with non-zero covariance. Top panel: Two covariates (HT1). Bottom panel: Two covariates with additional 14 noise covariates drawn from a standard normal distribution (HT2).

on the first dimension. The top panel of Figure 1 shows the mean relative error rates for 50 replications with $n_L = 200$ and $n_T = 1000$. Relative error rates here means that for each replication the error rate of each procedure is divided by the smallest error rate which for this replication is achieved by any of the classification methods under comparison. So for example a procedure that is the best in each replication would always have the relative error rate one. This type of illustration is used e.g. by Friedman (2001) and makes it easier to judge relative performance.

Linear discriminant analysis (LDA) is seen to have the best performance.

This does not come as a surprise because the decision boundary is a straight line and can be matched very well by LDA. Localized logistic regression (LLR) is similar to LDA in performance. In most of the replications the optimization procedure chooses parameters that result in no localization and no variable selection and so LLR becomes a global logistic discrimination procedure which is similar to LDA. The local fitting of a quadratic model is rather stable. The performance is slightly worse than for the linear localization but still outperforms most of the alternative procedures.

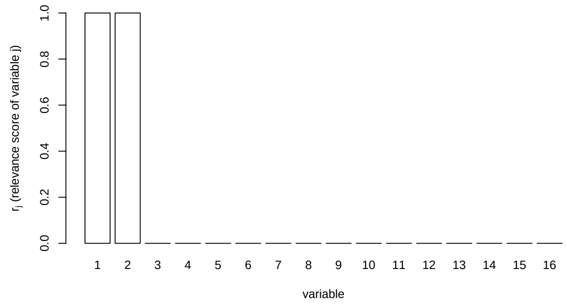

Figure 2: LLR (linear) relevance scores of variables for an example where the first two variables carry information and the other 14 are noise variables (LLR parameters: $k = n_L$, $c_\beta = 1.6$ and $\lambda = 0.42$).

To test the variable selection component of LLR we augmented the two covariates providing information on class membership by 14 noise variables taken from a standard normal distribution (HT2) as in example 2 of Hastie and Tibshirani (1996). As seen in the bottom panel of Figure 1 LLR again performs very well. When noise variables are included it also outperforms LDA. Figure 2 shows the relevance scores of the variables, i.e. which variables have been selected by lLLR. It can be seen that the variables that carry information on class membership have been distinctly identified. This taken together with the good performance indicates that selection of predictors succeeds here. Another indicator for the usefulness of local variable selection is the worsening performance of the nearest-neighbourhood algorithms due to the addition of noise variables. They share with LLR the notion of localization, but lack the possibility of predictor selection. Moreover, it can be seen from Figure 2 that the true relevance of variables is uncovered very well by local techniques.

The performance of quadratic LLR is similar to the linear version. This indicates the even in the presence of a considerable amount of noise the superfluous complexity of the quadratic version does not result in overfitting but is reduced to a level that is appropriate for the underlying structure.

In the next two examples (F1 and F2), equivalent to example 1 and 2 of Friedman (1994), the amount of information provided by the variables is systematically varied. For both classes the covariates are drawn from a 10-dimensional normal distribution. For the class with $y_i = 0$ the data are generated from standard normal distributions. For the other class the mean and the variance depend on the variable index. Covariance is set to zero in both examples.

In the first example (F1) the covariates with the higher index j are intended to be more relevant. So for $y_i = 1$ data are generated from a normal distribution $x_i \sim N(m, C)$ where

$$\left\{m_j = \sqrt{j}/2\right\}_1^p, \ C = \mathrm{diag}\left\{1/\sqrt{j}\right\}_1^p.$$

The top panel of Figure 3 shows the relative error rates for 50 replications with $n_L = 200$ and $n_T = 1000$. The good performance of procedures using linear combinations of predictors (LLR, LDA and neural networks) indicates that the Bayes decisions boundary can be approximated very well by hyperplanes. Similar to the example with a two-dimensional normal distribution without noise LLR does not make much use of localization and predictor selection and so resembles a global procedure.

In the second example (F2) the mean structure of the second class is changed to

$$\left\{m_j = \sqrt{p-j+1}/2\right\}_1^p$$

and so the variables with lower index contain more relevant information on class membership due to the mean and the variables with higher index due to the variance. The bottom panel of Figure 3 shows the relative error rates for this example. Again a hyperplane approximation of the Bayes decision boundary seems to be very efficient and so the procedures

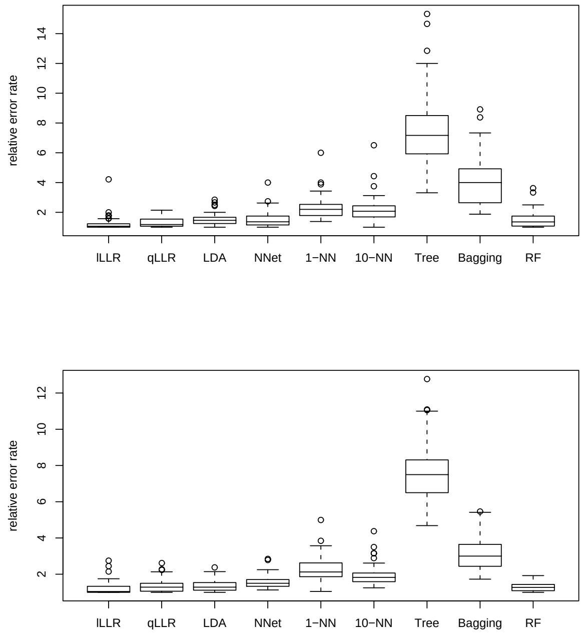

Figure 3: Relative error rates for data with covariates from 10-dimensional normal distributions. Top panel: Variables with higher index carry more information on class membership (F1). Bottom panel: Variables with low index have more information due to the mean, variables with higher index have more information due to the variance (F2).

that employ linear combinations of predictors have the best performance. Local quadratic approximation in combination with predictor selection again performs very stable.

3.1.2 Classes with non-overlapping connected class regions

In the examples presented in this section the variables that carry information on class membership define non-overlapping and connected class regions. In some examples they are augmented by noise variables.

In the first example (F5), which is equivalent to example 5 of Friedman (1994), the class boundary is defined by a linear combination of the covariates. It is constructed in a way so that all input variables have equal local relevance everywhere in the space spanned by the covariates. However, there is a single direction in that space that contains all the discriminating information. There are $p = 10$ covariates. The class membership rule is

$$\sum_{j=1}^{10} x_{ij} \leq 9.8 \Rightarrow y_i = 0, \text{ otherwise} \Rightarrow y_i = 1.$$

The top panel of Figure 4 shows the relative error rates for 50 replications with $n_L = 200$ and $n_T = 1000$. The procedures that utilize a linear combination of predictors clearly have the best performance (with LLR among them). Quadratic approximation is outperformed only by neural networks (and lLLR).

The situation changes for example F4 (as in example 4 of Friedman, 1994) where a quadratic combination of covariates is used to define the class boundary: The bottom panel of Figure 4 shows the results for the class membership rule

$$\sum_{j=1}^{10} x_{ij}^2 \leq 9.8 \Rightarrow y_i = 0, \text{ otherwise} \Rightarrow y_i = 1$$

and training sample size of $n_L = 500$ and $n_T = 1000$. The performance of the linear procedures LDA and neural networks degrades. Localized procedures show the best performance, only bagging and random forest are comparable to local linear procedures. Of course quadratic LLR is the distinct winner in this example, because by utilizing quadratic components the quadratic class boundary can be approximated very well. But it should be noted that *linear* localization works very well even in this case. If consideration of localized procedures is limited to linear procedures localization is still the best procedure for this example.

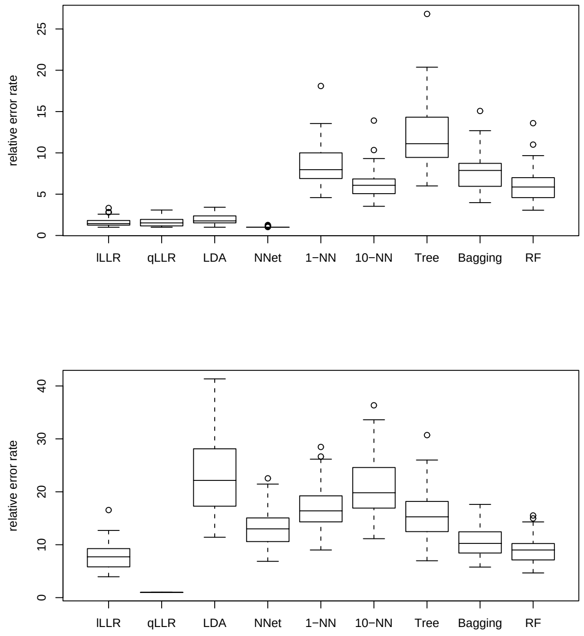

Figure 4: Relative error rates for classes with non-overlapping connected class regions defined by a linear (example F5; top panel) or quadratic (example F4; bottom panel) combination of covariates.

In particularly for the last two examples classification trees show very bad performance compared to other tree-based procedures. Based on the good performance of LLR in these examples we conjectured that local models beyond the capabilities for localization found in classification trees are required here. We therefore investigated the use of a locally weighted version of classification trees. An experimental implementation of localized trees was found to have improved classification performance, especially for the example with a linear class boundary. As it did not come close to the performance of bagging or random forests we did not investigate this class of models further. Nevertheless this result emphasizes the necessity of localized models.

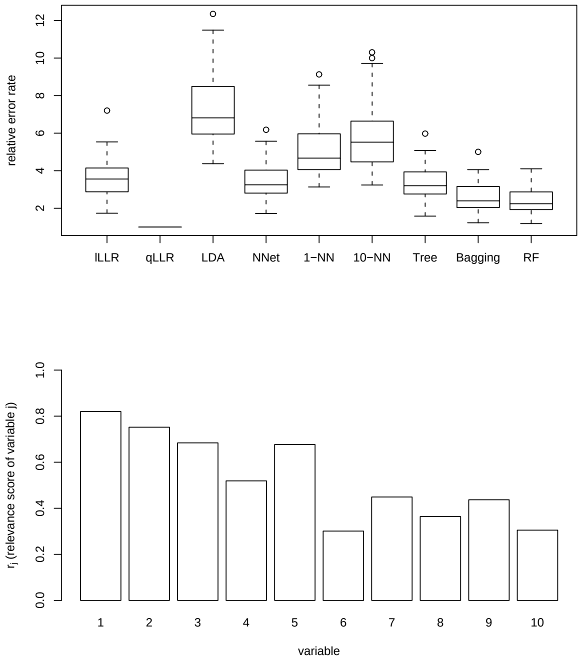

Figure 5: Top panel: Relative error rates for an example with non-overlapping connected class regions with a boundary defined by a weighted quadratic combination of covariates (F3). Bottom panel: Relevance scores of variables from linear LLR predictor selection (LLR parameters: $k = 0.6 \cdot n_L$, $c_\beta = 0.4$ and $\lambda = 0.9$)

A variant of the considered data structure is characterized by varying relevance of predictors. When a weighted contribution of the variables

$$\sum_{j=1}^{10} x_{ij}^2/j \leq 2.5 \Rightarrow y_i = 0, \text{ otherwise} \Rightarrow y_i = 1.$$

with training sample size of $n_L = 200$ and $n_T = 1000$ is used (F3), as in example 3 of Friedman (1994), the performance of linear LLR degrades to the level of neural networks (top panel of Figure 5). One of the reasons for this might be unwaranted exclusion of variables: As is shown in the bottom panel of Figure 5 covariates with high index that receive less weight in the definition of the class boundary are excluded in more than half of the cases from local model building and even the covariates with low index that carry much information are not always used. This indicates that the class boundary is too complicated to be approximated very well by local linear models. In contrast quadratic LLR again performs very well here due to the quadratic approximation of the class boundary.

In the next three examples (HT5), based on example 5 of Hastie and Tibshirani (1996), there are four covariates drawn from standard normal distributions. Class membership is assigned by the following rule:

$$\sqrt{\sum_{j=1}^{4} x_{ij}^2} \leq 3 \Rightarrow y_i = 0, \text{ otherwise} \Rightarrow y_i = 1.$$

These variables are augmented with a varying number of noise variables drawn from a standard normal distribution. The top panel of Figure 6 shows the relative error rates for an example with no noise variables. LLR performs very well here. When adding six standard normal noise variables the performance of linear LLR decreases relative to procedures like Bagging and Random Forest (bottom panel of Figure 6). With 16 noise variables the performance decreases even more (not shown). These results indicate that predictor selection for linear LLR when dealing with quadratic structure only works well up to a certain amount of noise and then fails gradually when too many variables (with irrelevant information) are present. The performance of quadratic LLR relative to other procedures does not seem to be impaired by the inclusion of noise variables. This indicates that selection of predictors works very well if the local models have the appropriate structure.

3.1.3 Fractioned class structure

The examples presented until now used classes where either observations were drawn from one distribution per class or one connected class region was used. In the following we will look at examples with extremely fractioned class regions denoted by HT3 and HT4 similar to examples 3 and 4 of Hastie and Tibshirani (1996).

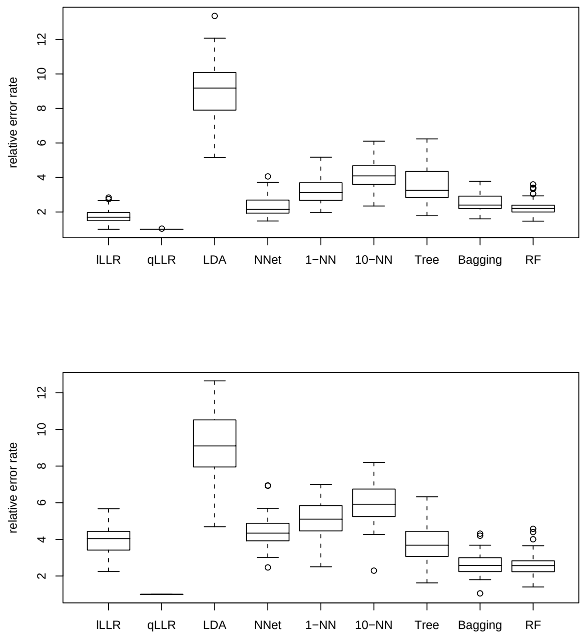

Figure 6: Relative error rates for examples where the class region of one class is a four-dimensional sphere (HT5) without any noise variables (top panel) or with six noise variables in addition (bottom panel).

In the first example (HT3) the distribution of each of the two classes is defined as a mixture of six spherical bivariate normal subclasses. The standard deviation of each subclass is 0.25. The means of the 12 subclasses are chosen for each replication at random (without replacement) from the integers $[1, 2, \ldots, 5] \times [1, 2, \ldots, 5]$. There are 20 observations drawn from the distribution of each subclass and so there are 140 observations per class with a total of $n_L = 240$ observations ($n_T = 960$). As seen in the top panel of Figure 7 all procedures (except LDA) perform very similar on the data. The best performance is found for LLR and 10-nearest neighbourhood. The optimal selection of the localization parameter for LLR leads to very local models here and this indicates that LLR can

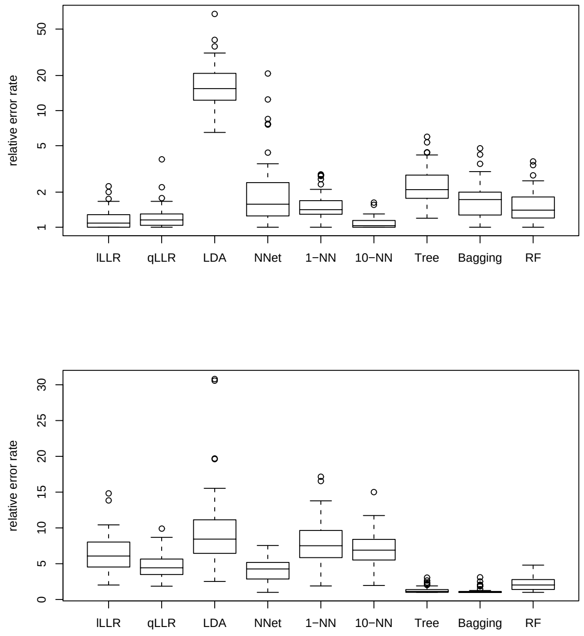

Figure 7: Relative error rates for an extremely fractioned class structure. Top panel: without noise variables (HT3). Bottom panel: with eight noise variables (in addition to two variables carrying information on class membership) (HT4).

become a nearest neighbourhood method in the limiting case.

In the next example (HT4) we augmented the two variables carrying information on class membership with eight noise covariates having standard normal distribution. As seen in the bottom panel of Figure 7 the performance of LLR degrades compared to the partition-based classification tree, bagging and random forest procedures. The latter procedures seem to perform very well in seperating informative variables from noise variables. When looking at which variables got selected by (linear) LLR (Figure 8) it can be seen that selection of predictors did not work very

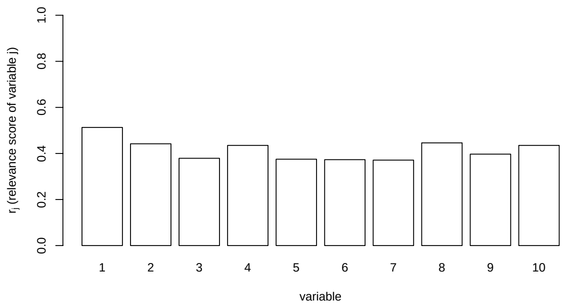

Figure 8: LLR (linear) relevance scores of variables for data with extremely fractioned class regions defined by two variables augmented by eight noise variables. Only the first two variables carry information on class membership. (LLR parameters: $k = 0.2 \cdot n_L$, $c_\beta = 0.4$ and $\lambda = 0.58$)

well for the local models. This may be due to the relatively low number of data points for each subclass compared to the number of noise variables. This is an example where LLR local model building does not work because there are too many noise variables.

3.2 Summary of simulation results

Table 1 gives the mean error rates for all procedures and all examples. The error rates of the two best procedures in each example are printed in boldface. Table 2 shows for each procedure used in the simulation study the ratio between the mean error rate of the procedure and the mean error rate of quadratic localized logistic regression (qLLR).

It is seen that for different situations different classification methods turn out to be the best choice, but some procedures react more flexible to varying data structures. Given that it cannot be expected that one method is superior in all data situations LLR performs rather well in a variety of different data structures.

Table 1: Mean error rates for different simulated data examples and classification procedures (lLLR: linear localized logistic regression; qLLR: quadratic LLR; LDA: linear discriminant analysis; NNet: neural networks with committee voting; 1-NN/10-NN: 1- and 10-nearest-neighbourhood classification; Tree: cross validated classification trees; Bag: bagging with trees; RF: random forests).

	lLLR	qLLR	LDA	NNet	1-NN	10-NN	Tree	Bag	RF
multivariate normal									
HT1	**0.069**	0.072	**0.065**	0.070	0.098	0.074	0.104	0.090	0.086
HT2	**0.072**	**0.074**	0.077	0.089	0.252	0.178	0.115	0.114	0.135
F1	**0.016**	**0.017**	0.019	0.019	0.028	0.026	0.089	0.050	0.019
F2	**0.021**	0.023	**0.022**	0.026	0.037	0.031	0.124	0.052	0.021
no overlap, connected									
F5	**0.045**	0.046	0.053	**0.030**	0.236	0.171	0.329	0.214	0.166
F4	**0.156**	**0.021**	0.453	0.263	0.339	0.413	0.309	0.213	0.179
F3	0.250	**0.074**	0.507	0.240	0.354	0.403	0.229	0.180	**0.164**
HT5									
no noise	**0.097**	**0.057**	0.496	0.128	0.176	0.223	0.196	0.139	0.126
some noise	0.219	**0.056**	0.499	0.241	0.280	0.325	0.210	0.146	**0.145**
more noise	0.312	**0.058**	0.505	0.397	0.346	0.381	0.218	**0.150**	0.180
fractioned class structure									
HT3	**0.024**	0.026	0.330	0.045	0.032	**0.023**	0.048	0.035	0.031
HT4	0.246	0.194	0.342	0.164	0.281	0.262	**0.059**	**0.053**	0.096

1. LLR shows better performance than LDA and nearest neighbourhood approaches in almost all examples (one exception for LDA, one exception for 10-NN). In the examples where LLR and LDA have similar performance (HT1 and HT2) the optimal LLR localization parameter with respect to the cross validation score is found to be $k = n_L$. This indicates that local models are not necessary for these examples and LLR becomes a global procedure. For the few examples where nearest neighbourhood methods perform well (e.g. HT3) LLR performance is similar. LLR parameter selection here is found to favour very local models and so LLR becomes a nearest neighbourhood method.

2. Surprisingly the comparison to Neural Networks is in favour of LLR, despite the fact that in contrast to LLR neural networks can model interactions of covariates directly. Neural networks as used here perform distinctly better only in two examples. The same holds for simple trees which are no serious alternative.

3. In all the examples where observations are drawn from a multivariate normal distribution per class, with and without noise the performance of LLR is one of the best of all methods. Trees perform

Table 2: Ratios of mean error rate relative to the mean error rate of quadratic localized logistic regression for different simulated data examples and classification procedures (lLLR: linear localized logistic regression; LDA: linear discriminant analysis; NNet: neural networks with committee voting; 1-NN/10-NN: 1- and 10-nearest-neighbourhood classification; Tree: cross validated classification trees; Bag: bagging with trees; RF: random forests).

	lLLR	LDA	NNet	1-NN	10-NN	Tree	Bag	RF
multivariate normal								
HT1	0.96	0.91	0.97	1.36	1.04	1.45	1.25	1.20
HT2	0.97	1.04	1.19	3.39	2.40	1.55	1.53	1.82
F1	0.94	1.12	1.13	1.70	1.58	5.35	2.97	1.11
F2	0.91	0.98	1.15	1.61	1.38	5.44	2.29	0.94
non-overlapping connected								
F5	0.99	1.15	0.66	5.12	3.72	7.15	4.64	3.60
F4	7.32	21.3	12.4	15.9	19.5	14.6	10.0	8.44
F3	3.38	6.84	3.24	4.77	5.44	3.09	2.43	2.21
HT5								
no noise	1.72	8.78	2.26	3.11	3.94	3.47	2.46	2.23
some noise	3.89	8.87	4.27	4.98	5.78	3.73	2.58	2.57
more noise	5.42	8.77	6.89	6.01	6.61	3.78	2.60	3.13
fractioned class structure								
HT3	0.92	12.5	1.72	1.20	0.87	1.82	1.32	1.18
HT4	1.27	1.76	0.84	1.45	1.35	0.30	0.27	0.49

very badly, only random forests come close.

4. For the investigated data structures with non-overlapping class regions the only competitors to localizing techniques are advanced tree methodologies as bagging and random forest. For F3 and HT5 (some noise) random forests outperform linear localizing procedures. When quadratic terms are included into localizing LLR dominates distinctly which in these cases is due to the underlying quadratic structure.

5. For data with fractioned class structure with noise variables tree-based approaches perform very well in particular if noise variables are included. Although LLR performs better without noise variables it is outperformed if much noise is present. Advanced tree methodology as bagging and random forests clearly perform best in this case.

6. The comparison between linear and quadratic LLR shows that the quadratic version performs only slightly worse in examples where linear local models work well. This indicates that penalization and predictor selection succeed in preventing overfitting when quadratic local models have superfluous complexity. On the other hand there are several examples (e.g. F3 and HT5) where linear LLR is clearly outperformed by quadratic LLR and so the latter should be preferred.

   One my argue that these examples are artificially constructed to favor quadratic models, but it can be seen that even with quadratic structure in some examples (e.g. F4) linear LLR performs well compared to other procedures. So the examples illustrate for which kind of structure linear local models are sufficient and for which not.

4 Application to real data

In the previous section simulated examples have been used to investigate how different types of structure affect the performance of localized logistic regression (LLR). As simulated data are, by definition, always artificial in this section we will use real data sets to investigate real world performance.

We use the Australian credit data from the Statlog project (Michie et al., 1994) and the breast cancer and the sonar data from the UCI machine learning repository Blake and Merz (1998). One reason for this selection of data sets is that they have been used in recent work on boosting methods (Bühlmann and Yu, 2003) and so information on error rates is available for a class of procedures that is considered to perform very well.

For the Australian credit data the aim is to devise a rule for assessing applications for credit cards. The data set has 14 covariates and 690 observations. Due to confidentiality neither the meaning of the covariates nor the exact meaning of the two classes is known. For the use with LLR, LDA, neural networks and the nearest neighbourhood methods some variables had to be transformed from categorical to binary dummy variables, with categories that have a relative frequency below five percent being discarded. The sonar data contains 208 sonar patterns from either "mines" or "rocks" at various angles and under various conditions. Each pattern is a set of 60 numbers in the range 0.0 to 1.0. The aim is to classify an object as “mine” or “rock” given a pattern. The breast cancer data has nine predictors and 699 observations. The classes are “benign” and “malignant” and the covariates contain various cell characteristics.

Each data set has been split 50 times randomly into a 90% training and 10% test set and all procedures used in the simulation study have been applied. We used linear instead of quadratic LLR because it is much faster and shows sufficently good performance. Table 3 shows the error rates for the three data sets and all procedures used in the simulation study. In addition the error rates for several boosting procedures as given in Bühlmann and Yu (2003) are shown. For two data sets results for boosting with splines in addition to boosting with tree stumps are available.

For the Australian credit data nearest neighbourhood classification rules yield very bad performance while the rest of the procedures are well comparable. For the breast cancer example LLR, 10-nearest neighbourhood classification and random forests distinctly outperform the rest. It can be seen that LLR performs well for all three data sets. Special attention should be given to the superior performance for the sonar data. The relatively good performance of 1-nearest neighbourhood compared to 10-nearest neighbourhood classification hints at a very local data structure. The good performance of neural networks and that of boosting with splines compared to boosting with tree stumps indicates that there is some kind of linear structure. LLR is able to model local as well as linear structures. This combination might explain the superior performance.

Although averaging across various splits is preferable for the sonar data a specific 50% split is used because it is a reference suggested by Gorman and Sejnowski (1988) and has been used by Hastie and Tibshirani (1996). For this specific split one obtains error rates 0.106 (lLLR), 0.240 (LDA), 0.115 (neural networks), 0.087 (1-nearest neighbourhood), 0.288 (10-nearest neighbourhood), 0.269 (trees), 0.192 (bagging) and 0.173 (random forests). Thus also for the fixed splitting LLR performs very well. Hastie and Tibshirani (1996)obtained for their discriminant adaptive nearest neighbour classifier (DANN) the test error rate 0.048 which is better than the LLR procedure. When a finer grid is used for parameter selection the LLR error rate reduces to 0.010. This may be interpreted as an artifact, but it also shows that with some tuning of the parameter selection procedure the LLR results are well comparable to the results

Table 3: Error rates for real data and various classification procedures. The numbers are mean error rates for 50 random splits into a 90% training and 10% test set. (lLLR: linear localized logistic regression; LDA: linear discriminant analysis; NNet: neural networks with committee voting; 1-NN/10-NN: 1- and 10-nearest-neighbourhood classification; Tree: cross-validated classification trees; Bag: bagging with trees; RF: random forests; L2Boost, L2WCBoost and LogitBoost: various boosting algorithms with tree stump and spline base learners).

	Australian credit	breast cancer	sonar
lLLR	0.126	0.029	0.078
LDA	0.146	0.037	0.273
NNet	0.140	0.035	0.165
1-NN	0.325	0.039	0.181
10-NN	0.313	0.029	0.336
Tree	0.153	0.054	0.271
Bag	0.138	0.039	0.212
RF	0.125	0.028	0.164
L2Boost*	0.123	0.037	0.228
with spline*		0.036	0.178
L2WCBoost*	0.123	0.040	0.190
with spline*		0.043	0.168
LogitBoost*	0.131	0.039	0.158
with spline*		0.038	0.148

* from Bühlmann and Yu (2003)

for DANN and other procedures given by Hastie and Tibshirani (1996), which indicates good local adaptivity.

5 Concluding remarks

A localized discrimination procedure has been proposed which in combination with local selection of predictors shows promising results. Although a method cannot be expected to be best for all potential data structures the performance is surprisingly good over a wide range of data structures.

While it outperforms tree methodology as bagging and boosting for simple structures, the latter dominate at least the linear version for quadratically separated classes with many noise variables. For real data sets, the localizing methodology works very well with the best performance for two of the considered data sets. This shows the potential in statistical applications.

It should be noted that the method is not intended to compete with methods from machine learning which are designed for image processing or pattern recognition where highly fractioned class structures with many noise variables might occur. The focus is on statistical applications where also the relevance of variables might be of interest. For these applications it is a serious alternative to existing methodology which might supplement the statistical tool box for classification.

Acknowledgment
We thank the German Science Foundation DFG (Sonderforschungsbereich 386) for financial support.

References

[1] Albert, A. and J. A. Anderson: 1984, 'On the Existence of Maximum Likelihood Estimates in Logistic Regression Models'. *Biometrika* **71**(1), 1–10.

[2] Bellman, R. E.: 1961, *Adaptive Control Processes: A Guided Tour.* Princeton, NJ: Princeton University Press.

[3] Bishop, C. M.: 1995, *Neural Networks for Pattern Recognition.* Oxford: Clarendon Press.

[4] Blake, C. and C. Merz: 1998, 'UCI Repository of machine learning databases'.

[5] Breiman, L.: 1996, 'Bagging Predictors'. *Machine Learning* **24**(2), 123–140.

[6] Breiman, L.: 1999, 'Prediction Games and Arcing Algorithms'. *Neural Computation* **11**, 1493–1517.

[7] Breiman, L.: 2001, 'Random Forests'. *Machine Learning* **45**(1), 5–32.

[8] Breiman, L.: 2002, 'Manual On Setting Up, Using, And Understanding Random Forests V3.1'.

[9] Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone: 1984, *Classification and Regression Trees*. Wadsworth.

[10] Bühlmann, P. and B. Yu: 2003, 'Boosting With the L2Loss: Regression and Classification'. *Journal of the American Statistical Association* **98**(462), 324–339.

[11] Fan, J. and I. Gijbels: 1996, *Local Polynomial Modelling and its Applications*. London: Chapman & Hall.

[12] Fix, E. and J. L. Hodges: 1951, 'Discriminatory Analysis, Nonparametric Discimination, Consistency Properties'. Technical Report 4, United States Air Force, School of Aviation Medicine, Randolph Field, TX.

[13] Friedman, J. H.: 1994, 'Flexible metric nearest neighbor classification'. Technical report, Standford University.

[14] Friedman, J. H.: 2001, 'Greedy Function Approximation: A Gradient Boosting Machine'. *Annals of Statistics* **29**, 1189–1232.

[15] Friedman, J. H., T. Hastie, and R. Tibshirani: 2000, 'Additive Logistic Regression: A Statistical View of Boosting'. *Annals of Statistics* **28**, 337–407.

[16] Gorman, R. P. and T. J. Sejnowski: 1988, 'Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets'. *Neural Networks* **1**, 75–89.

[17] Hastie, T. and R. Tibshirani: 1996, 'Discriminant Adaptive Nearest Neighbor Classification'. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **18**(6), 607–615.

[18] Hastie, T., R. Tibshirani, and J. Friedman: 2001, *The Elements of Statistical Learning*. New York: Springer.

[19] Holm, S.: 1979, 'A Simple Sequentially Rejective Multiple Test Procedure'. *Scandinavian Journal of Statistics* **6**, 65–70.

[20] Ihaka, R. and R. Gentleman: 1996, 'R: A Language for Data Analysis and Graphics'. *Journal of Computational and Graphical Statistics* **51**(3), 299–314.

[21] Kauermann, G. and G. Tutz: 2000, 'Local Likelihood Estimates and Bias Reduction in Varying Coefficients Models'. *Journal of Nonparametric Statistics* **12**, 343–371.

[22] Kira, K. and L. A. Rendell: 1992, 'A Practical Approach to Feature Selection'. In: D. Sleeman and P. Edwards (eds.): *Machine Learning. Proceedings of the Ninth International Workshop (ML92).* San Mateo, Morgan Kaufmann.

[23] Kohavi, R. and G. H. John: 1998, 'The Wrapper Approach'. In: H. Liu and H. Motoda (eds.): *Feature Extraction, Construction and Selection. A Data Mining Perspective.* Dordrecht: Kluwer Academic Publishers, pp. 33–50.

[24] Le Cessie, S. and J. C. van Houwelingen: 1992, 'Ridge Estimators in Logistic Regression'. *Applied Statistics* **41**(1), 191–201.

[25] Loader, C.: 1999, *Local Regression and Likelihood.* New York: Springer.

[26] Michie, D., D. J. Spiegelhalter, and C. C. Taylor: 1994, *Machine Learning, Neural and Statistical Classification.* New York: Ellis Horwood.

[27] Powell, M. J. D.: 2002, 'UOBYQA: unconstrained optimization by quadratic approximation'. *Math. Program.* **92**, 555–582.

[28] Ripley, B. D.: 1996, *Pattern Recognition and Neural Networks.* Cambridge: Cambridge University Press.

[29] Schaal, S., S. Vijayakumar, and C. G. Atkeson: 1998, 'Local Dimensionality Reduction'. In: M. I. Jordan, M. J. Kearns, and S. A. Solla (eds.): *Advances in Neural Information Processing Systems 10.* Cambridge, MA: MIT Press.

[30] Venables, W. N. and B. D. Ripley: 1999, *Modern Applied Statistics With S-Plus.* Springer, 3rd edition.