Bundling Classifiers with an Application to Glaucoma Diagnosis

Dissertation

zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften
der Universität Dortmund

Dem Fachbereich Statistik
der Universität Dortmund
vorgelegt von

Torsten Hothorn
aus Dohna

Dortmund 2003

Prüfungskommission:	Prof. Dr. G. Trenkler (Vorsitzender)
	Prof. Dr. C. Weihs (Gutachter)
	Prof. Dr. W. Krämer (Gutachter)
	PD Dr. B. Lausen (Gutachter)
	Dr. G. Knapp (Beisitzer)
Tag der mündlichen Prüfung:	10. März 2003

How many lakes did you make in your life now?

Uwe Johnson: Jahrestage

Aus dem Leben von Gesine Cresspahl: 20. April, 1968

Danksagung

Die Aufnahme und Vollendung dieser Arbeit wäre ohne die direkte und indirekte Hilfe Vieler nicht denkbar gewesen. Meinem wissenschaftlichen Mentor, PD Dr. Berthold Lausen, gebührt mein Dank für die Betreuung dieser Arbeit, zahlreiche anregende und aufregende Diskussionen sowie gemeinsame thematische Abstecher. Für die gelegentlich notwendige Unterstützung bei Bemühungen, die Dinge im rechten Licht zu betrachten, sowie für die großzügige Gestaltungsfreiheit bei meiner Tätigkeit am Institut für Medizininformatik, Biometrie und Epidemiologie in Erlangen möchte ich Prof. Dr. Olaf Gefeller danken.

Prof. Dr. Claus Weihs gab den Anstoß für eine auch theoretische Betrachtung des Themas. Für diese Anregung sowie für die Begutachtung und Korrektur der Arbeit bedanke ich mich. Ebenso danke ich Prof. Dr. Walter Krämer für Gutachten und Korrekturhinweise und Prof. Dr. Götz Trenkler als dem Vorsitzenden der Prüfungskommision.

Meiner Kollegin Dr. Christine Vogel bin ich für Rettung in letzter Minute verpflichtet. Hinweise, die zu einer glücklicheren Strukturierung des theoretischen Teils der Arbeit geführt haben, sind mir dankenswerterweise

von Dipl.-Stat. Achim Zeileis gegeben worden. Für die kritische Kommentierung des von mir produzierten Codes (nicht nur per `R CMD check`) sowie lehrreiche Diskussionen über das Design von Benutzerschnittstellen danke ich Prof. Dr. Kurt Hornik. Kerstin Saler, M.A., soll bedankt sein für die Ausmerzung der gröbsten sprachlichen Ausrutscher.

Meinen ehemaligen und jetzigen Erlanger Kolleginnen und Kollegen, die ich mich scheue, sie an dieser Stelle in eine Reihenfolge zu bringen, möchte ich danken für Hilfe und Unterstützung, Motivation und Ablenkung, Kaffee und Kuchen sowie für wissenschaftliche und nichtwissenschaftliche Ausfüge, nicht nur auf den "Berg".

Das Fundament dieser Arbeit jedoch bilden meine Familie und Freunde, die mir immer mit Rat und Tat zur Seite standen und mich zur rechten Zeit, bewußt oder unbewußt, an die wirklich wichtigen Dinge erinnert haben.

Torsten Hothorn Erlangen, im März 2003

Contents

Abstract

The combination of classifiers of arbitrary type, for example classification trees, linear discriminant analysis, nearest neighbors or the logistic regression model, is most desirable for at least two reasons. Firstly, a combined classifier can be expected to improve any of the single classifiers with respect to misclassification error and, secondly, the need for an explicit selection of one of the competitors for a special classification problem, usually causing a method selection bias for error rate estimation, disappears.

In this work we propose to use the out-of-bag observations of a bootstrap sample, i.e. all observations of a learning sample which are not part of the bootstrap sample itself, to learn classifiers of arbitrary type. The predictions of those classifiers, for example predicted classes, estimated conditional class probabilities or linear discriminant values, are computed for the observations in the bootstrap sample and are used as predictors offered to a classification tree in addition to the original predictors. The classification tree implicitly selects the most informative predictors, either the original ones or transformations of them, in order to construct a classifier. In this sense, the classification tree "bundles" the additional classifiers. In

analogy to bagging, the procedure is sufficiently repeated and the class of a new observation is predicted by majority voting. Consequently, we call the procedure "bundling".

The superior performance of the proposed combined classifier is shown using standard benchmark classification problems from the UCI machine learning database. Moreover, we prove the almost sure risk consistency of bundling using results of data driven random partitions.

In the second part of this work, the methodology is used to classify eyes as either normal or glaucomatous, based on predictors derived from laser scanning images of the eye background. Glaucoma is one of the major reasons for blindness worldwide and an early detection of a glaucomatous change of the optic nerve head morphology is important. The performance of different candidates for glaucoma classification is evaluated by using a simulation model of the optic nerve head morphology. Bundling performs superior to other classifiers both in our simulation experiments as well as for the observations of a case-control study.

Finally, we illustrate how to combine classifiers via bundling within the R system for statistical computing, using the case-control study on glaucoma as example.

Zusammenfassung

Die Kombination mehrerer verschiedener Klassifikationsverfahren, wie zum Beispiel Klassifikationsbäume, lineare Diskriminanzanalyse, nächste Nachbarn oder logistische Regression, ist aus mindestens zwei Gründen wünschenswert. Zum einen ist zu erwarten, daß die Kombination verschiedener Verfahren zu einer Reduktion der Fehlerraten der einzelnen Komponenten führt. Zum anderen ist die explizite Schätzung der Fehlerrate eines jeden Verfahrens auf einem Datensatz, welche im allgemeinen zu einer Verzerrung bei der Schätzung der Fehlerrate des selektierten Verfahrens führt, nicht mehr notwendig.

In dieser Arbeit wird vorgeschlagen, die unterschiedlichen Klassifikatoren auf den sogenannten "out-of-bag" Beobachtungen anzulernen, also auf den Beobachtungen einer Lernstichprobe, welche nicht Teil einer Bootstrapstichprobe sind. Die Vorhersagen dieser Klassifikatoren, dies können die vorhergesagte Klasse, geschätzte bedingte Klassenwahrscheinlichkeiten oder die Werte von linearen Diskriminanzfunktionen sein, werden zusätzlich zu den Originalvariablen benutzt, um einen Klassifikationsbaum zu konstruieren. Das rekursive Partitionieren wählt implizit die für

die Vorhersage der Klassenzugehörigkeit informativsten Variablen oder eben deren Transformationen aus und "bündelt" die zusätzlichen Klassifikatoren in diesem Sinne. Wie beim bagging wird nun diese Prozedur ausreichend oft wiederholt und die Klasse einer neuen Beobachtung wird per Mehrheitsabstimmung über die multiplen Bäume bestimmt. Wir bezeichnen daher unseren Vorschlag als "bundling".

Die sehr guten Eigenschaften des kombinierten Klassifikators werden in einem Vergleich mittels mehrerer realer und künstlicher Standardklassifikationsprobleme nachgewiesen. Darüberhinaus zeigen wir unter Benutzung von Resultaten zu Zufallspartitionen, daß die Fehlerrate von bundling asymptotisch fast sicher gegen die Fehlerrate des Bayes-Klassifikators konvergiert.

Im zweiten Teil der vorliegenden Arbeit werden die methodischen Ergebnisse benutzt, um gesunde Augen von Augen mit einer beginnenden glaukomatösen Schädigung zu unterscheiden. Das Glaukom oder der grüne Star ist einer der häufigsten Gründe für eine Erblindung und eine möglichst frühe Diagnose daher wichtig. Die Fehlerraten verschiedener Klassifikatoren für die Glaukomdiagnose werden mittels eines Simulationsmodells des Sehnervenkopfes verglichen. Bundling erreicht sowohl in unseren Simulationsuntersuchungen als auch für die Daten einer Fall-Kontroll-Studie die kleinsten Fehlerraten aller untersuchten Verfahren.

Abschließend illustrieren wir die Kombination von Klassifikatoren in der statistischen Programmierumgebung R anhand der Daten der Fall-Kontroll-Studie.

Chapter 1

Introduction

Many medical decisions are based on examinations that involve medical imaging techniques, for example magnet resonance tomography (MRT) in radiology or laser scans of the eye background in ophthalmology. The medical diagnosis, i.e. the decision whether a subject suffers a special disease of not, using the information provided by such a medical imaging device requires the knowledge of an experienced physician and is time consuming and costly. Therefore, the development of automated decision systems is the basis for the application of such imaging techniques in situations, where a human examination of a large number of subjects is impossible due to financial and time constraints, for example in screening programs.

The challenge to biostatistics is to construct a rule which can be used to discriminate between healthy subjects and patients suffering a disease, where the only information to be used by this rule is a, possibly large, set

of numerical measurements derived from a medical image.

In a statistical framework this problem is known as discriminant analysis or classification problem, whereas in machine learning the term "supervised learning" is used. Typically the decision rule, called classifier, is constructed ("learnt" or "trained") by using a number of observations with known class labels describing the true state of the observation. In the simplest medical context the two classes may be either "healthy" or "ill". This set of observations is called learning sample. Once a classifier has been learnt, this rule can be used to predict the diagnosis of a new subject based on numerical measurements derived from an appropriate examination without human interaction. The performance of such a classifier is measured by the expected proportion of faultily predictions, called the misclassification error.

The construction of a good classifier based on a learning sample can be seen as a three step procedure. At first, we use the observations in the learning sample to construct different rules. In the second step we need to choose the best among them. This is usually done by selecting the rule with minimum estimated misclassification error. And at last, but not least, an honest estimate of the misclassification error of the selected procedure is required, for example to decide whether this classifier is good enough to be applied in practical situations or not. A common problem, especially in medical statistics, is that only a small learning sample with a large number of possible predictors is available and all three steps have to be performed by using this small learning sample.

Two main problems arise. First, we need to choose an appropriate classifier out of a number of possible candidates, for example linear or tree based classifiers, neural networks, nearest neighbors or support vector machines. And second, we need to estimate the misclassification error of the selected procedure. It is well known that the selection of a classification rule with minimum estimated misclassification error leads to biased estimates of its performance. Even with efficient estimates of misclassification error like the .632+ bootstrap estimator by Efron and Tibshirani (1997), the minimum of several estimators of misclassification error is a downward biased estimate of the true error rate. Nevertheless, different rules have to be taken into account. In many applications simple rules like naive Bayes, nearest neighbors or linear discriminant analysis perform comparably to more advanced classifiers (see for example Friedman, 1997). However, the individual classifiers perform well in certain situations and fail under other conditions.

One approach to solve both problems simultaneously, i.e. the method selection and error rate estimation problem, is to apply a combination of classifiers. Instead of selecting one single procedure, combining the competitors may improve classification rules. There are several approaches to the combination of different classifiers. A linear combination of the estimated conditional class probabilities is suggested by LeBlanc and Tibshirani (1996) and Mojirsheibani (1997), which is related to linear combinations of regression models (Breiman, 1996c; LeBlanc and Tibshirani, 1996). Merz (1999) uses correspondence analysis to combine the predictions of

different classifiers. Majority voting of the predictions of the different classifiers is introduced by Mojirsheibani (1999, 2002).

We propose a new procedure for the combination of classifiers of different kind. The basic idea is to add the outcome of arbitrary classifiers (linear discriminant variables, predicted conditional class probabilities or predicted classes) to the set of original predictors for bagging of classification trees (Breiman, 1996a, 1998). For the special case of linear combinations of the predictors, Hothorn and Lausen (2002a, 2003b) suggest a combination of linear discriminant analysis (LDA) and classification trees (CTREE, Breiman et al., 1984), called "double-bagging". LDA and CTREE are somewhat extreme models. The LDA assumes a spherical distribution of the predictors in each class. The classes are separable by hyperplanes in the sample space. In contrast, classification trees are nonparametric, i.e. do not assume a special distribution of the predictors. Basically, classification trees are constructed by a recursive search for both the cutpoint and predictor which separate the observations best with respect to an appropriately defined two sample statistic. Therefore, CTREE searches for rectangular partitions in the multivariate sample space, which may be seen as higher order interactions or homogeneous subgroups defined by some combination of binary splits of the predictors. Because of that it is natural to combine both ideas. Breiman et al. (1984), page 16, noted that it is "... of surprise that [LDA] does as well as it does ..." and consequently suggested to investigate linear combinations of the predictors in each node.

Classification trees are unstable in the sense that a small perturbation of

the learning sample, i.e. removing or adding some observations, may produce a completely different tree. The idea of aggregating the predictions of different trees, which were constructed on reweighted observations in the learning sample, lead to a stabilization and substantial reduction of the misclassification error in many applications. The most common procedures are boosting (Freund and Schapire, 1996; Schapire et al., 1998) and bagging (Breiman, 1996a, 1998). Boosting is based on an deterministic and iterative weighting scheme whereas bagging predicts the class of a new observation by majority voting of the predictions from trees which were constructed by using bootstrap samples of the original learning sample.

As in all statistical procedures which are based on the bootstrap, approximately $1/3$ of the observations are not part of a single bootstrap sample in bagging. Breiman (1996b) calls these observations "out-of-bag". Double-bagging for the combination of LDA and CTREE uses the out-of-bag sample to estimate the coefficients of a linear discriminant function. The corresponding linear discriminant variables computed for the bootstrap sample are used as additional predictors for the classification trees which allow a linear separation of the classes. This method performs comparably to LDA when the classes are linearly separated and comparable to bagging if the classes can be identified by partitions.

The proposal is not restricted to the combination of LDA and CTREE but can be extended to the combination of arbitrary classifiers. For each bootstrap sample, a number of classifiers is constructed using the out-of-bag observations only. The predictions of those classifiers, i.e. predicted

classes, estimated conditional class probabilities or linear discriminant values, are computed for the observations in the bootstrap sample and used as additional predictors for a classification tree. The trees implicitly select the most informative predictors and "bundle" in this sense the additional classifiers. The procedure is sufficiently repeated and a new observation is classified by averaging the predictions of the multiple trees. The idea is in the spirit of Breiman (2001b): instead of reducing the dimensionality we enlarge the number of possible predictors available to the classification trees and stabilize the procedure by bootstrap aggregation. It should be noted that this approach is completely contrary to the classical statistical practice, where the number of possible predictors is reduced by variable selection procedures (for example Miller, 2002) before classifiers are constructed.

Moreover, by using the suggested combination of classifiers we do not need to estimate the error rates of the individual classifiers because the classifiers are implicitly selected by classification trees. Therefore, an unbiased estimate of the misclassification error of the combined procedure can be computed for example with the use of cross-validation or the .632+ bootstrap.

Benchmark experiments and simulations show that the suggested combination of classifiers performs comparably to the best of the single classifiers or bagging with respect to misclassification error. Furthermore, the combined classifier improves the best of the single classifiers or bagging in a number of examples. It turned out in our experiments that the pro-

cedure performs comparably or even improves random forests (Breiman, 2001a), one of the best classifiers known today, for most of the benchmark problems.

Although the combined classifiers are shown to be practically useful by means of benchmark experiments, the asymptotic properties of the method are of theoretical interest. Using results on random partitions (Lugosi and Nobel, 1996) it can be shown that the misclassification error of the combined classifier converges almost surely to the Bayes error. The result holds independent of the choice of the additional classifiers and for a fixed number of bootstrap samples.

Our motivation for the development of combined classifiers origined from the problem of glaucoma classification for screening programs in the project "Automated Glaucoma Screening" as part of the Sonderforschungsbereich 539 "Glaucoma including Pseudoexfoliationsyndrome" at the Friedrich–Alexander–University Erlangen–Nuremberg. Glaucoma is an ocular disease which causes progressive damage in the optic nerve fibres and leads to visual field loss. The prevalence of this irreversible neurodegenerative disease is about 2.5% and glaucoma is the second leading cause of blindness worldwide (Coleman, 1999).

The detection of early glaucomatous damage in the eye is of major importance to permit an early treatment. Additionally, the development of screening programs for glaucoma relies on methods for the detection of glaucoma at an early stage. The visual field loss caused by loss of retinal nerve fibres can only be measured at an advanced stage of the disease. In

contrast to examinations of the visual field, laser scanning images of the papilla are able to detect early stages of glaucoma (Mikelberg et al., 1995; Mardin et al., 1999). We therefore focus on the development of a good classifier for glaucoma based on parameters derived from laser scanning images of the eye background. In routine examinations, 62 predictors are derived from a laser scanning image. As it is frequent in medical research, the number of available observations in our learning sample is rather limited: 98 normal eyes and 98 glaucomatous eyes of 196 subjects matched by age and gender are included in a case-control study for the construction of a classifier (Mardin et al., 2002; Hothorn et al., 2003). As already mentioned, this sample is too small for the training and method selection as well as error rate estimation. Hothorn and Lausen (2002b, 2003a) suggest to use a simulation model of the optic nerve head for investigating the performance of different classifiers to avoid the method selection bias. As already mentioned, the need for an explicit selection of classifiers by means of estimates of their misclassification error disappears for a combined classifier.

This work is organized as follows. Chapter 2 introduces the basic model of discriminant analysis. The methodology of combining classifiers is developed in Chapter 3. Before the strong risk consistency of the combined classifier is derived in Chapter 5, we discuss computational problems of the procedure as well as possible solutions in Chapter 4. We compare the performance of the proposed procedure with the performance of the individual classifiers and random forests (Breiman, 2001a) by some

benchmark problems from the UCI machine learning repository (Blake and Merz, 1998) in Chapter 6. Three procedures for estimating the error rate of a classifier are reviewed in Chapter 7.

The problem of glaucoma classification based on numerical predictors derived from laser scanning images of the eye background as well as the design of the case-control study is introduced in Chapter 8. The performance of linear and tree based classifiers for glaucoma classification and a combination of both are evaluated by means of a simulation study based on a model of the surface of the papilla in Chapter 9. In addition, we compare three error rate estimators under the same simulation setup. Finally, we use the results of the simulation experiments and construct a classifier for glaucoma diagnosis, using the data of the case-control study as learning sample and compute an honest estimate of its misclassification error in Chapter 10.

Some details of a free software implementation of combined classifiers in the R system for statistical computing as well as parallel statistical simulations using the Mosix load-balancing system are discussed in Chapter 11.

Chapter 2

The Discriminant Analysis Model

In this Chapter we consider the basic model for discriminant analysis problems: a p-dimensional vector of predictors $\mathbf{X} \in \mathbb{R}^p, \mathbf{X} = (X_1, \ldots, X_p)$ is observed and associated with a class label $Y \in \{1, \ldots, J\}$. The joint distribution function of the predictors and class labels is denoted by $\mathcal{F}$. We observe a learning sample $\mathcal{L}_n$ of n independent and identically distributed random samples from $\mathcal{F}$:

$$\mathcal{L}_n = \{(y_i, \mathbf{x}_i); i = 1, \ldots, n\}.$$

Furthermore, the marginal distribution function of the predictors $\mathbf{X}$ is denoted by $\mathcal{F}_\mathbf{X}$.

We try to predict the class for a new observation $\mathbf{x}_{\text{new}}$ based on the learning sample by a rule C, called classifier. A classifier is a function

$$C : \mathbb{R}^p \to \{1, \ldots, J\}$$

which maps the p-dimensional predictors into the class labels. The function C is a composition of two functions $C = g \circ c$ where

$$\begin{aligned} c &: \mathbb{R}^p \to \mathbb{R}^{(J-1)} \text{ and} \\ g &: \mathbb{R}^{(J-1)} \to \{1, \ldots, J\}. \end{aligned}$$

The value $c(\mathbf{x}_{\text{new}})$ may be the vector of the conditional class probability estimators and g the argmax function. For the linear discriminant analysis, $c(\mathbf{x}_{\text{new}})$ are the discriminant variables and g is a function defining the class. The classifier is trained using the learning sample $\mathcal{L}_n$, and we denote the dependence of the classifier C on the learning sample by writing

$$C(\mathbf{x}_{\text{new}}; \mathcal{L}_n) = g(c(\mathbf{x}_{\text{new}}; \mathcal{L}_n)).$$

By convention, $c_j(\mathbf{x}_{\text{new}}; \mathcal{L}_n)$ is the jth element of the vector $c(\mathbf{x}_{\text{new}}; \mathcal{L}_n) \in \mathbb{R}^{(J-1)}$.

The performance of a classifier is measured by the misclassification error, i.e. the expected loss

$$L(C) = P_{\mathcal{F}}(C(\mathbf{X}) \neq Y).$$

The Bayes classifier is the classifier with minimum loss and is of the form

$$C^{\text{Bayes}}(\mathbf{x}) = \underset{j=1,\ldots,J}{\operatorname{argmax}} P_j(\mathbf{x})$$

where $P_j(\mathbf{x}) = P(Y = j | \mathbf{X} = \mathbf{x})$ are the conditional class probabilities (cf. Ripley, 1996). The misclassification error of the Bayes classifier is called Bayes error

$$L_{\text{Bayes}} = L(C^{\text{Bayes}}).$$

Since this misclassification error is hardly measurable in realistic setups, we focus on the conditional error rate, where the condition is on the learning sample $\mathcal{L}_n$:

$$L(C(\cdot; \mathcal{L}_n)) = P_{\mathcal{F}}(C(\mathbf{X}; \mathcal{L}_n) \neq Y | \mathcal{L}_n).$$

Resampling based estimators of the misclassification error are given in Chapter 7.

The asymptotic properties of a classifier C are of theoretical interest. A classifier C is said to be strongly or almost surely risk consistent if the conditional error rate tends to the Bayes error with probability one as the sample size n of the learning sample $\mathcal{L}_n$ tends to infinity:

$$L(C(\cdot; \mathcal{L}_n)) \to L_{\text{Bayes}} \text{ a.s. }.$$

Chapter 3

Combining Classifiers

The major contribution of this work, a combination of classifiers of arbitrary type, is given in this Chapter. The aim is to combine a main classifier $C^{\text{main}} = g^{\text{main}} \circ c^{\text{main}}$ and K additional classifiers $C^k = g^k \circ c^k; k = 1, \ldots, K$ of arbitrary type. We use superscripts to distinguish between the different classifiers, for example linear discriminant analysis, nearest neighbors or any other kind of classifier.

3.1 Bundling

The combined classifier is defined as the main classifier trained on the original p predictors and additional $K(J-1)$ transformations of the original predictors in the combined learning sample:

$$C^{\text{comb}}(\mathbf{x}_{\text{new}}; \mathcal{L}_n) = C^{\text{main}}(\mathbf{x}_{\text{new}}; \mathcal{L}_{\text{comb},n}),$$

where the combined learning sample $\mathcal{L}_{\text{comb},n}$ is given by

$$\mathcal{L}_{\text{comb},n} = \left\{ \left(y_i, \mathbf{x}_i, c^1(\mathbf{x}_i), \ldots, c^K(\mathbf{x}_i) \right); i = 1, \ldots, n \right\}.$$

We will choose classification trees as main classifier. The motivation for adding transformations to the set of original predictors is that classification trees allow for rectangular splits of the form $X_i \leq \xi$, but classification trees based on the combined learning sample $\mathcal{L}_{\text{comb},n}$ allow for more general splits of the form $c_j^k(\mathbf{X}) \leq \xi, \xi \in \mathbb{R}$. For example, if the discriminant variables of a linear discriminant analysis are used as additional predictors, linear splits of the form $\alpha^\top \mathbf{X} \leq \xi$ for $\alpha \in \mathbb{R}^p$ are possible. Note that in this framework, $c^k(\mathbf{x}); k = 1, \ldots, K$ may or may not depend on the learning sample $\mathcal{L}_n$.

One variant of random forests (Forest-RC, Breiman, 2001a) uses random linear combinations of the predictors. In contrast to random linear combinations we will add possibly non-linear transformations $c^k(\mathbf{X})$ of the predictors that depend on the data. However, an additional learning sample for training c^k is often not available and sample splitting is inefficient for small learning samples. Computing c^k and c^{main} using the same data is not appropriate. In this case, c^{main} is likely to select transformations derived from the individual classifier c^k which overfits the data the most. Therefore, we use the out-of-bag samples for the bootstrap aggregated combined classifiers as independent learning samples as follows.

Let $\mathcal{L}_n^*$ denote a bootstrap sample of size n from the empirical distribu-

tion function $\hat{\mathcal{F}}$:

$$\mathcal{L}_n^* = \{(y_i^*, \mathbf{x}_i^*); i = 1, \ldots, n\} \sim \hat{\mathcal{F}}.$$

In standard bagging, the multiple trees trained on the bootstrap samples $\mathcal{L}_n^*$ are aggregated by class majority voting, i.e. voting of the class predictions for a new observation. However, Breiman (1996a) mentions that averaging the conditional class probability estimators and choosing the class with highest average conditional class probability leads roughly to the same results. We therefore define the bootstrap aggregated main classifier by the expectation of c^{main} with respect to $\hat{\mathcal{F}}$:

$$\begin{aligned} C^{\text{bagmain}}(\mathbf{x}_{\text{new}}; \mathcal{L}_n) &= g^{\text{main}}\left(\mathrm{E}_{\hat{\mathcal{F}}} c^{\text{main}}(\mathbf{x}_{\text{new}}; \mathcal{L}_n^*)\right) \\ &= g^{\text{main}}\left(\int c^{\text{main}}(\mathbf{x}_{\text{new}}; \mathcal{L}_n^*) d\hat{\mathcal{F}}(\mathcal{L}_n^*)\right) \end{aligned}$$

and approximate it by a finite number of B bootstrap samples

$$C_B^{\text{bagmain}}(\mathbf{x}_{\text{new}}; \mathcal{L}_n) = g^{\text{main}}\left(B^{-1} \sum_{b=1}^{B} c^{\text{main}}\left(\mathbf{x}_{\text{new}}; \mathcal{L}_n^{*(b)}\right)\right). \quad (3.1)$$

In each bootstrap sample, approximately one third of the original observations are left out. Breiman (1996b) calls the observations in $\mathcal{L}_n \setminus \mathcal{L}_n^*$, i.e. the observations which are not element of the bootstrap sample, "out-of-bag" and uses this additional independent sample for improved estimators of the conditional class probabilities. Another application is the use of the out-of-bag sample for estimating the prediction error, see Section 7. Rao and Tibshirani (1997) discuss a weighted prediction, where the weights are determined from the out-of-bag samples. In contrast to these

suggestions, we will use the out-of-bag observations $\mathcal{L}_n \setminus \mathcal{L}_n^*$ for training of additional classifiers.

For a bootstrap sample $\mathcal{L}_n^*$, the combined learning sample is defined by

$$\mathcal{L}^*_{\text{comb},n} = \{(y_i^*, \mathbf{x}_i^*, c^1(\mathbf{x}_i^*; \mathcal{L}_n \setminus \mathcal{L}_n^*), \ldots, c^K(\mathbf{x}_i^*; \mathcal{L}_n \setminus \mathcal{L}_n^*)); i = 1, \ldots, n\},$$

i.e. the out-of-bag sample is now used as independent learning sample for $c^k, k = 1, \ldots, K$. The learning sample $\mathcal{L}^*_{\text{comb},n}$ consists of the original predictors as well as $K(J-1)$ transformations of them, induced by the independently constructed $c^1, \ldots, c^K$.

The bootstrap aggregated combined classifier is now given by

$$\begin{aligned} C^{\text{bagcomb}}(\mathbf{x}_{\text{new}}; \mathcal{L}_n) &= g^{\text{main}}\left(\mathrm{E}_{\hat{\mathcal{F}}} c^{\text{comb}}(\mathbf{x}_{\text{new}}; \mathcal{L}_n^*)\right) \text{ where} \\ c^{\text{comb}}(\mathbf{x}_{\text{new}}; \mathcal{L}_n^*) &= c^{\text{main}}(\mathbf{x}_{\text{new}}; \mathcal{L}^*_{\text{comb},n}) \end{aligned} \tag{3.2}$$

and again we approximate C^{bagcomb} by a finite number of B bootstrap replications:

$$C_B^{\text{bagcomb}}(\mathbf{x}_{\text{new}}; \mathcal{L}_n) = g^{\text{main}}\left(B^{-1}\sum_{b=1}^{B} c^{\text{comb}}\left(\mathbf{x}_{\text{new}}; \mathcal{L}_n^{*(b)}\right)\right).$$

Each of the B classification trees works as a main classifier that selects and combines the predictions of the additional classifiers $c^1, \ldots, c^K$. Consequently we will call the procedure "bundling".

3.2 Double-Bagging

Bundling is a generalization of a combination of linear discriminant analysis and bagging ("double-bagging", Hothorn and Lausen, 2002a, 2003b).

Double-bagging uses the values of the linear discriminant functions trained on the out-of-bag sample as additional predictors for bagging classification trees only, i.e. is a special case of bundling with $K = 1$. In the two class problem, one additional predictor is added to the set of original predictors. This predictor is simply a linear combination of the original predictors in the bootstrap sample, its coefficients are estimated using the out-of-bag observations only. Since the combination of LDA and CTREE is the simplest configuration and combines two very popular classifiers, we will use double-bagging in a simulation study of glaucoma classifiers in Chapter 8.

Chapter 4

Computational Aspects of Bundling

In our experiments using benchmark datasets and simulation studies in Chapter 6, we compute LDA, nearest neighbors and the multinomial or logistic regression model as additional classifiers. Since these classifiers are trained using the out-of-bag observations only, this subset of the learning sample may be too small. For example, in the case-control study for the classification of glaucoma based on laser scanning images of the eye background in Chapter 8, the out-of-bag sample contains, on average, 72 observations. The estimation of a covariance matrix for the LDA with 62 predictors is usually infeasible. We suggest two strategies to deal with this problem. One possibility is to reduce the number of possible predictors. For the incorporation of linear combinations of the original predictors, we use a stabilized LDA (sLDA, Läuter, 1992) based on low dimensional PC-q

scores instead of the original predictors. The other possibility is to enlarge the out-of-bag sample by using a different resampling scheme.

4.1 Stabilized Linear Discriminant Analysis

One rather general and successful method of dimension reduction in multivariate analysis is based on the theory of left-spherically distributed random variables (Fang and Zhang, 1990). For example, in the framework of multivariate analysis of variance, Läuter et al. (1998) show that the null distribution of Hotellings T^2 statistic remains the same for low dimensional linear combinations of the multivariate response variables derived from the total sum of products matrix. The power of the associated test is increased compared to the power of Hotellings T^2 test when the number of responses is large. In fact, the basic idea was developed in the discriminant analysis framework (Läuter, 1992).

Let $\bar{\mathbf{x}}$ denote the p-dimensional mean vector over all n observations and let

$$\mathbf{W} = \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top$$

denote the total sum of products matrix as defined in Läuter et al. (1998). The stabilized linear discriminant analysis uses the PC-q scores (Läuter, 1992; Läuter et al., 1998; Kropf, 2000), i.e. a q-dimensional linear combination $\tilde{\mathbf{x}}_i = \mathbf{x}_i \mathbf{D}_q$ (with $q < p$), instead of the original p-dimensional predictors $\mathbf{x}_i$. The matrix $\mathbf{D}_q$ is the $p \times q$ matrix of eigenvectors with eigenvalues greater one in the eigenvalue problem $\mathbf{WD} = \mathrm{Diag}(\mathbf{W})\mathbf{D}\Lambda$, where Λ is

the diagonal matrix of the p eigenvalues and $\mathbf{D}$ is the $p \times p$ matrix of the corresponding eigenvectors. Theorem 2 of Läuter et al. (1998) states that $\tilde{\mathbf{x}}_i$ are left-spherically distributed random variables. A stabilized linear discriminant analysis can therefore be computed based on the q-dimensional linear combination $\mathbf{x}_i\mathbf{D}_q$ of the original predictors.

4.2 Subundling

Another possibility is to give more weight to the out-of-bag sample. Instead of sampling n out of n with replacement, Bühlmann and Yu (2002) suggested subsampling ("subagging"), i.e. sampling 50% of the data without replacement. Consequently, we call this procedure "subundling". This modification ensures that the out-of-bag sample always contains half of the observations: the learning samples for the training of the additional classifiers, i.e. the out-of-bag samples, are now large enough.

Chapter 5

Strong Risk Consistency of Bundling

The asymptotic properties of bundling are of theoretical interest. We will show that the misclassification error of bundling tends to the misclassification error of the Bayes classifier as the number of observations in the learning sample tends to infinity. This result holds for arbitrary additional classifiers as well as for a fixed number of bootstrap samples under rather weak conditions.

To establish the strong risk consistency of C_B^{bagcomb} we use results on random partitions by Lugosi and Nobel (1996). Let $\pi_n(\mathcal{L}_n) = \{A_{n,1}, \ldots, A_{n,r}\}$ denote a data-driven partition of $\mathbb{R}^p$, i.e.

$$\bigcup_{i=1}^{r} A_{n,i} = \mathbb{R}^p \text{ and } A_{n,i} \bigcap A_{n,j} = \emptyset \text{ for all } i \neq j$$

depending on a learning sample $\mathcal{L}_n$. The unique cell containing $\mathbf{x}$ is denoted by $\pi_n[\mathbf{x}]$.

In the following, we restrict ourselves to classification trees, i.e. tree structured partition based main classifiers C^{main} of the form

$$c^{\text{main}}(\mathbf{x}) = \left(\frac{\sum_{i=1}^{n} I(\mathbf{x}_i \in \pi_n[\mathbf{x}], y_i = j)}{n\mu(\pi_n[\mathbf{x}])} \right)_{j=1,\ldots,(J-1)} \tag{5.1}$$

where μ is the probability measure of $\mathbf{X}$, $I(\cdot)$ is the indicator function and

$$C^{\text{main}}(\mathbf{x}) = g^{\text{main}}(c^{\text{main}}(\mathbf{x})) = \underset{j=1,\ldots,J}{\operatorname{argmax}}\, c_j^{\text{main}}(\mathbf{x})$$

where by convention $c_J^{\text{main}}(\mathbf{x}) = 1 - \sum_{j=1}^{J-1} c_j^{\text{main}}(\mathbf{x})$. Let $\operatorname{diam}(A)$ denote the diameter of a subset A of $\mathbb{R}^p$

$$\operatorname{diam}(A) = \sup_{s,t \in A} ||s - t||_2.$$

We now establish the strong risk consistency of the combined classifier C^{comb} and the bootstrap aggregated combined classifier C_B^{bagcomb}.

Theorem 1. *Let $\{\pi_1, \pi_2, \ldots\}$ be a sequence of tree structured partitioning rules for $\mathbb{R}^p$ induced by C^{comb}. Suppose that for every sequence of learning samples $\mathcal{L}_n$, each cell of the partition $\pi_n(\mathcal{L}_n)$ contains at least h_n of $\mathbf{x}_1, \ldots, \mathbf{x}_n$, where*

$$\frac{h_n}{\log(n)} \to \infty.$$

If in addition for all $\gamma > 0$ and $\delta \in (0, 1)$:

$$\inf_{\mathbf{S} \subseteq \mathbb{R}^p : \mu(\mathbf{S}) \geq 1-\delta} \mu\{\mathbf{x} \in \mathbb{R}^p | diam(\pi_n[\mathbf{x}] \cap \mathbf{S}) > \gamma\} \to 0 \text{ a.s.}$$

then C^{comb} is strongly risk consistent.

The last assumption is a shrinking cell condition depending on the probability measure of the predictors $\mathbf{X}$. The proof of Theorem 1 is a simple extension to the proof of Theorem 3 of Lugosi and Nobel (1996), where the strong risk consistency of classification trees is shown. Basically, we show that adding $K(J-1)$ additional predictors does not affect the strong risk consistency of classification trees.

Proof. For a family of partitions $\mathcal{A}$ of $\mathbb{R}^p$, the maximal cell count is given by

$$m(\mathcal{A}) = \sup_{\pi \in \mathcal{A}} |\pi|$$

where $|\pi|$ denotes the number of cells in π. The complexity of $\mathcal{A}$ is measured by the growth function

$$\Delta_n(\mathcal{A}) = \max_{\mathcal{T}_n = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathbb{R}^{n \times p}} \left\{ \text{number of different sets in } \left\{ (A_1 \cap \mathcal{T}_n, \ldots, A_p \cap \mathcal{T}_n) | (A_1, \ldots, A_r) \in \mathcal{A} \right\} \right\}.$$

Under the three conditions

a) $n^{-1} m(\mathcal{A}_n) \to 0$

b) $n^{-1} \log(\Delta_n(\mathcal{A}_n)) \to 0$

c) for all $\gamma > 0$ and $\delta \in (0, 1)$:

$$\inf_{\mathbf{S} \subseteq \mathbb{R}^p : \mu(\mathbf{S}) \geq 1 - \delta} \mu\{\mathbf{x} \in \mathbb{R}^p | \text{diam}(\pi_n[\mathbf{x}] \cap \mathbf{S}) > \gamma\} \to 0 \text{ a.s.}$$

the strong risk consistency of partition based classifiers follows from the L_1 consistency of the associated conditional class probability estimators (Theorem 2 of Lugosi and Nobel, 1996).

Now let $\mathcal{A}_n$ denote the set of all possible partitions of $\mathbb{R}^p$ induced by π_n or c^{main}, respectively. The following facts are simple extensions to the proof of Theorem 3 of Lugosi and Nobel (1996). By assumption, each partition induced by a classification tree contains at most n/h_n cells, therefore $n^{-1}m(\mathcal{A}_n) \leq h_n^{-1} \to 0$. Each partition $\pi_n(\mathcal{L}_n)$ is based on not more that $m(\mathcal{A}_n) = n/h_n$ hyperplane splits in $X_1, \ldots, X_p, c^1(\mathbf{X}), \ldots, c^K(\mathbf{X})$. Each split can dichotomize $n \geq 2$ points in $\mathbb{R}^{p+K(J-1)}$ in at most $n^{p+K(J-1)}$ different ways (this upper bound is based on Cover, 1965). Therefore,

$$\Delta_n(\mathcal{A}_n) \leq n^{(p+K(J-1))^{n/h_n}} \text{ and}$$
$$\frac{1}{n}\log(\Delta_n(\mathcal{A}_n)) \leq \frac{(p+K(J-1))\log(n)}{h_n} \to 0.$$

Conditions a) and b) are satisfied and the strong risk consistency of C^{comb} follows from Theorem 2 of Lugosi and Nobel (1996). □

Theorem 2. *Under the assumptions of Theorem 1, $C_B^{bagcomb}$ is strongly risk consistent.*

The strong risk consistency of the bootstrap aggregated combined classifier C_B^{bagcomb} for a fixed and finite number of bootstrap replications B is proved by showing that bagging of any strongly risk consistent partition based classifier is again risk consistent, therefore we proof the following lemma.

Lemma 1. *Bagging of any strong risk consistent partition based classifier is again strongly risk consistent.*

Proof. Let $\hat{P}_j(\mathbf{x}; \mathcal{L}_n)$ denote a partition based estimator of the conditional class probability $P_j(\mathbf{x}) = P(Y = j|\mathbf{X} = \mathbf{x})$, i.e. an estimator of the form

(5.1):

$$\hat{P}_j(\mathbf{x}; \mathcal{L}_n) = \frac{\sum_{i=1}^{n} I(\mathbf{x}_i \in \pi_n[\mathbf{x}], y_i = j)}{n\mu(\pi_n[\mathbf{x}])}.$$

It holds for every class $j \in \{1, \ldots, J\}$ that the conditional class probability estimators are L_1 consistent, i.e.

$$\int |P_j(\mathbf{x}) - \hat{P}_j(\mathbf{x}; \mathcal{L}_n)| d\mathcal{F}_{\mathbf{X}}(\mathbf{x}) \to 0 \text{ a.s.}$$

(Proof of Theorem 2 of Lugosi and Nobel, 1996). Bagging of tree structured partition based main classifiers (cf. 5.1) as defined in (3.1) averages the estimates of the conditional class probabilities based on B bootstrap replications of the learning sample:

$$\hat{P}_j^*(\mathbf{x}; \mathcal{L}_n) = B^{-1} \sum_{b=1}^{B} \hat{P}_j\left(\mathbf{x}; \mathcal{L}_n^{*(b)}\right).$$

Now fix j for the moment. We have

$$\begin{aligned}
& \int \left| P_j(\mathbf{x}) - \hat{P}_j^*(\mathbf{x}; \mathcal{L}_n) \right| d\mathcal{F}_{\mathbf{X}}(\mathbf{x}) \\
= & \int \left| P_j(\mathbf{x}) - B^{-1} \sum_{b=1}^{B} \hat{P}_j\left(\mathbf{x}; \mathcal{L}_n^{*(b)}\right) \right| d\mathcal{F}_{\mathbf{X}}(\mathbf{x}) \\
\leq & \int B^{-1} \sum_{b=1}^{B} \left| P_j(\mathbf{x}) - \hat{P}_j\left(\mathbf{x}; \mathcal{L}_n^{*(b)}\right) \right| d\mathcal{F}_{\mathbf{X}}(\mathbf{x}) \\
= & B^{-1} \sum_{b=1}^{B} \left(\underbrace{\int \left| P_j(\mathbf{x}) - \hat{P}_j\left(\mathbf{x}; \mathcal{L}_n^{*(b)}\right) \right| d\mathcal{F}_{\mathbf{X}}(\mathbf{x})}_{\to 0 \text{ a.s. for every } j} \right) \to 0 \text{ a.s. for every } j.
\end{aligned}$$

In all B bootstrap samples, the estimated conditional class probabilities are L_1 consistent (Theorem 2 of Lugosi and Nobel, 1996) and therefore the finite sum tends to zero as well. This result implies the L_1 consistency

of the averaged estimators of the conditional class probabilities and the strong risk consistency of the associated classifier follows from Theorem 1 of Devroye and Györfi (1985), page 254. □

Proof of Theorem 2. Recall that both C^{main} (5.1) and C^{comb} (3.2) are based on classification trees. The strong risk consistency of both procedures follows from Theorem 3 of Lugosi and Nobel (1996) and Theorem 1. The strong risk consistency of C_B^{bagcomb} follows from Lemma 1 and the proof is complete. □

Note that the results hold true for subundling. If we sample 50% of the observations without replacement and denote this sample by $\mathcal{L}^*_{\lfloor n/2 \rfloor}$ we have

$$\int \left| P_j(\mathbf{x}) - \hat{P}_j\left(\mathbf{x}; \mathcal{L}^*_{\lfloor n/2 \rfloor}\right) \right| d\mathcal{F}_{\mathbf{X}}(\mathbf{x}) \to 0$$

as the sample size n of the learning sample $\mathcal{L}_n$ tends to infinity by the same arguments as for the proof of Theorem 2.

Chapter 6

Benchmark Experiments

In this Chapter we illustrate the performance of the combination of classifiers via bundling using three artificial, four small and three larger benchmark problems. The experiments were conducted with the `ipred` package (Peters et al., 2002, see Chapter 11) in the R system for statistical computing (version 1.5.1, Ihaka and Gentleman, 1996, `http://www.R-project.org`).

The breast cancer, ionosphere, diabetes, glass, satellite, shuttle and DNA datasets from the UCI machine learning repository (Blake and Merz, 1998) are assembled in the R package `mlbench` (Leisch and Dimitriadou, 2001). The code for generating the twonorm, threenorm and ringnorm data is part of the `mlbench` package as well. The artificial problems are defined as in Breiman (1998):

Twonorm: A two class problem with $p = 20$ predictors, drawn from a multivariate normal distribution with unit covariance matrix. The mean vectors are $(a, \ldots, a)$ and $(-a, \ldots, -a)$ with $a = 2/\sqrt{20}$.

Threenorm: A two class problem with $p = 20$ predictors, drawn from a multivariate normal distribution with unit covariance matrix. The observations of class one are drawn with equal probability with means $(a, \ldots, a)$ and $(-a, \ldots, -a)$ whereas the observations in class two are drawn with mean $(a, -a, a, \ldots, -a)$ where $a = 2/\sqrt{20}$.

Ringnorm: A two class problem with $p = 20$ predictors, drawn from a multivariate normal distribution with four times the unit covariance matrix for class one and unit covariance matrix for class two. The mean vectors are $(0, \ldots, 0)$ for class one and $(a, \ldots, a)$ with $a = 2/\sqrt{20}$ for class two.

We study bundling of three individual classifiers: stabilized linear discriminant analysis (sLDA, see Section 4.1), k nearest neighbors (k-NN, with $k = 5$ and $k = 10$) as well as logistic regression (LR). The multinomial model is used for problems with more than two classes. The values of the linear discriminant functions of the stabilized LDA as well as the predicted conditional class probabilities of nearest neighbors and the logistic regression model are combined. For bagging and bundling, 50 unpruned trees are used. We additionally report the error rates of random forests with 50 trees (Forest-RI, Breiman, 2001a), R package `randomForest` (Liaw and Wiener, 2002, version 3.3-2), where the number of randomly selected predictors in each node is chosen as the ceiling of $\log_2(p + 1)$.

The misclassification error for the artificial problems is estimated by the average of 100 simulation runs, where the learning samples are of size 300 and the error rate is computed using one single test sample of size

Data set	n	Test size	p	J
Breast Cancer	699	-	9	2
Ionosphere	351	-	34	2
Diabetes	768	-	8	2
Glass	214	-	9	6
Satellite	4435	2000	36	6
Shuttle	5803	5802	9	7
DNA	2000	1186	180	3
Twonorm	300	18000	20	2
Threenorm	300	18000	20	2
Ringnorm	300	18000	20	2

Table 6.1: Sample size of the learning samples, the number of predictors p and number of classes J for the benchmark datasets under consideration.

18000. For the larger datasets, a test sample is selected randomly for the larger problems. Table 6.1 reports the sample sizes of learning and test samples for the real world applications. The misclassification error for the smaller problems is estimated by averaging the misclassification error of ten independent runs of 10-fold cross-validation (see Section 7.1 also).

The simulated or estimated misclassification errors for the artificial and real world benchmark datasets are given in Table 6.2. A graphical representation of the simulation results for the artificial problems using boxplots is shown in Figure 6.1.

Bundling of sLDA, k-NN and the logistic regression model is at least as good as any of the single classifiers except for the diabetes data, where the

logistic regression model outperforms all other classifiers. The estimated misclassification error of bagging is, except for the glass data, larger than the misclassification error of the combined classifier. Subundling does not lead to a major improvement, the misclassification error of bundling and subundling are comparable in most problems. The boxplots of the simulated misclassification error for the artificial problems in Figure 6.1 show that bundling and subundling have a smaller variance than bagging for the three examples. For the twonorm and threenorm problem, the errors of the combined classifiers are respectively comparable to the error of sLDA or 10-NN. Surprisingly, bundling improves bagging also for the ringnorm problem, where all additional classifiers perform poor in general.

From this results we can conclude that the training and error rate estimation for each of the single classifiers is unnecessary because bundling "inherits" the properties of the best of the single classifier or even improves upon it. As discussed in the Introduction, any method selection bias can be avoided. Moreover, the combined classifiers perform as well ask random forests, one of the best classifiers known today, or even outperform random forests in a number of problems: twonorm, threenorm, ringnorm and DNA.

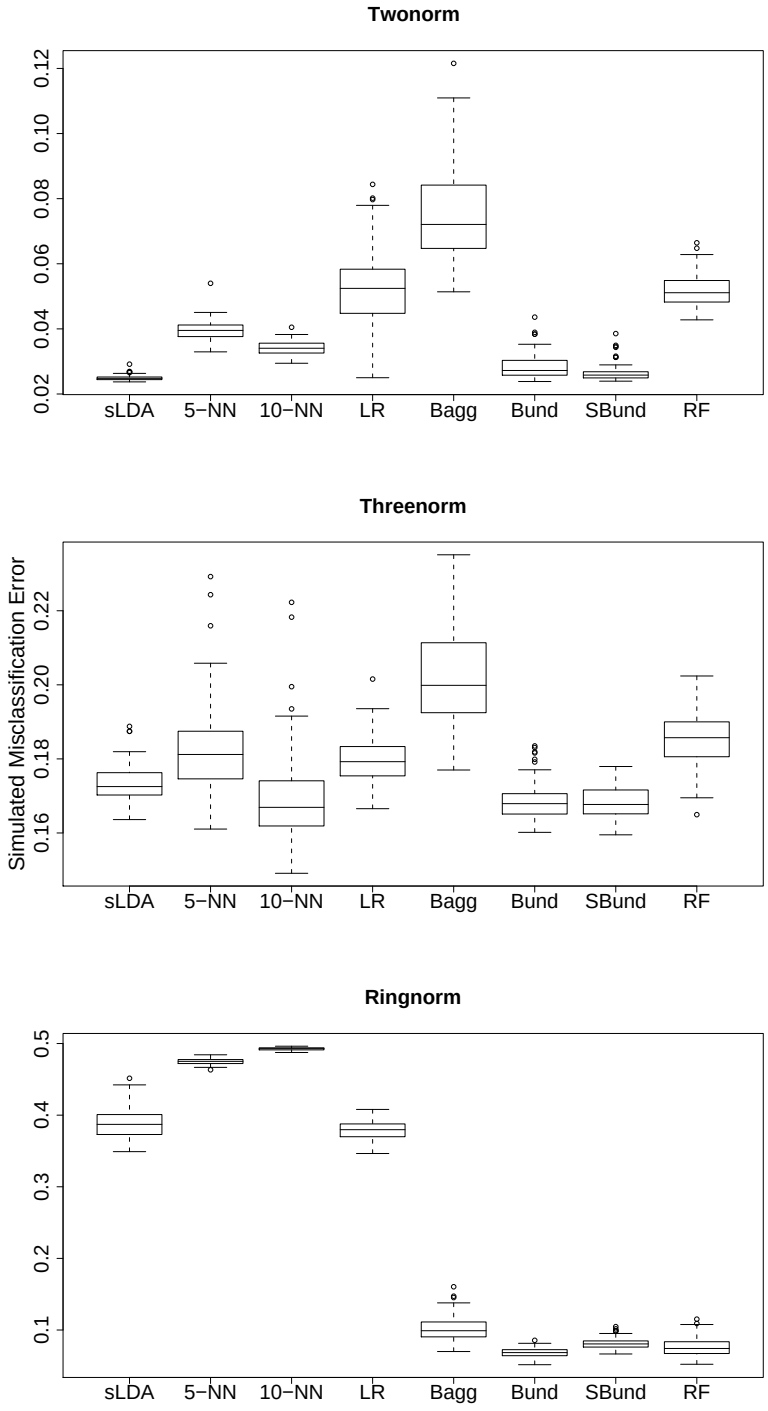


Figure 6.1: Misclassification error of 100 simulation runs for the artificial problems.

	sLDA	5-NN	10-NN	LR	Bagg	Bund	SBund	RF
Twonorm	2.5	4.0	3.4	5.2	7.5	2.8	2.6	5.2
Threenorm	17.3	18.2	16.9	18.0	20.2	16.9	16.8	18.6
Ringnorm	38.8	47.5	49.3	38.0	10.2	6.8	8.1	7.6
BreastCancer	3.3	6.4	8.3	7.4	4.0	3.0	3.1	3.0
Ionosphere	13.8	15.6	16.6	11.8	8.3	6.5	6.6	6.6
Diabetes	27.3	28.5	26.0	22.4	24.3	24.1	24.7	23.8
Glass	42.4	32.7	37.3	35.7	23.2	24.9	26.3	22.1
Satellite	19.3	8.7	9.6	19.2	8.4	7.4	7.7	7.4
Shuttle	8.2	0.4	0.6	2.9	0.1	0.1	0.1	0.1
DNA	8.1	19.5	16.2	10.4	4.5	3.1	3.1	5.6

Table 6.2: Estimated misclassification errors for the benchmark datasets and artificial problems for stabilized LDA (sLDA), nearest neighbors (k-NN), the logistic regression model (LR) and bagging (Bagg). Bundling (Bund) and subundling (SBund) both combine sLDA, k-NN ($k = 5$ and $k = 10$) and LR. Random forests (RF), bagging and (su)bundling were computed using 50 trees. The misclassification errors of the best procedures are underlined.

The benchmark experiments presented in this Chapter are limited with respect to both the number of problems and classifiers under test. Moreover, the influence of the choice of the additional classifiers for bundling was not studied. The number of 50 trees used for bagging, bundling and random forests is rather small due to computational limitations of bundling (see Section 11.1 for some details).

A more extensive comparison of 16 different classifiers (including support vector machines, neural networks and random forests) for 21 two class classification problems can be found in Meyer et al. (2003). This benchmark study includes bundling of sLDA as one of the competitors and shows the overall good performance of bundling: in 7 out of 21 classification problems, bundling is superior or shares the lowest misclassification error with a subset of the 16 classifiers under test (Meyer et al., 2003, Tables 3-6).

Chapter 7

Error Rate Estimators

For the estimation of the misclassification error of ensemble procedures like bagging we have to choose an appropriate error rate estimator. In this Chapter we review three commonly used error rate estimators: cross-validation, the .632+ bootstrap (Efron and Tibshirani, 1997) and the out-of-bag estimator for bagging (Breiman, 1996b). We will evaluate their performance, especially for glaucoma classification, using the problem specific simulation model in Chapter 9. The notation in this Chapter is similar to the one used by Efron and Tibshirani (1997). For a more detailed overview on error rate estimation we refer to Schiavo and Hand (2000).

Let Q denote the misclassification loss function

$$Q(y, \hat{y}) = \begin{cases} 1 & \text{if } y \neq \hat{y} \\ 0 & \text{if } y = \hat{y}. \end{cases}$$

Recall that, as defined in Chapter 2, the conditional true error rate Err is

the expected loss of a classifier C

$$\text{Err} = L(C(\cdot; \mathcal{L}_n))$$

and the following procedures can be used to estimate Err.

7.1 Cross-Validation

The apparent or resubstitution error rate is given by

$$\widehat{\text{Err}}^{(a)} = \frac{1}{N} \sum_{i=1}^{N} Q\left(y_i, C(\mathbf{x}_i; \mathcal{L}_n)\right)$$

and the leave-one-out bootstrap is defined as

$$\widehat{\text{Err}}^{(1)} = \frac{1}{N} \sum_{i=1}^{N} \text{E}_{\hat{\mathcal{F}}_{(i)}} Q\left(y_i, C(\mathbf{x}_i; \mathcal{L}_n^*(i))\right),$$

where $\hat{\mathcal{F}}_{(i)}$ is the empirical distribution function with the ith observation removed and $\mathcal{L}_n^*(i)$ is a bootstrap sample from $\hat{F}_{(i)}$. Especially, leave-one-out cross-validation estimates the conditional true error rate by

$$\widehat{\text{Err}}^{(cv1)} = \frac{1}{N} \sum_{i=1}^{N} Q\left(y_i, C(\mathbf{x}_i; \mathcal{L}_n(i))\right)$$

where $\mathcal{L}_n(i)$ is the learning sample with the ith observation removed. We use 10-fold cross-validation $\widehat{\text{Err}}^{(cv10)}$ where the learning sample is split into 10 approximately equal sized parts, either deterministically or randomly. The class labels for observations in one part are predicted by a classifier trained on the remaining parts and the misclassification rate is averaged over all 10 possible segmentations into learning and test sample.

There are several extensions to improve this estimator. Kohavi (1995) observed a reduction of the variance of the estimator when stratified sampling is used, i.e. when the proportions of the classes are preserved. Iterating $\widehat{\text{Err}}^{(cv10)}$ various times and using the mean or median of the estimators is a procedure usually used in benchmark studies (cf. Chapter 6).

7.2 The .632+ Bootstrap

The .632+ estimator (Efron and Tibshirani, 1997) takes the amount of over-fitting of a classifier C into account, that is the difference between the apparent error rate and the leave-one-out bootstrap estimator $\widehat{\text{Err}}^{(1)} - \widehat{\text{Err}}^{(a)}$. The amount of over-fitting is scaled by the no-information error rate $\gamma = \text{E}_{\mathcal{F}_0} Q\left(Y, C(\mathbf{X}; \mathcal{L}_n)\right)$, where the expectation is with respect to the distribution $\mathcal{F}_0$ with the same marginal distributions of response and predictors as $\mathcal{F}$ but under the null hypothesis that predictors and response are independent. In the two-class problem, the no-information error rate is estimated by

$$\hat{\gamma} \quad = \quad \hat{p}(1-\hat{q}) + (1-\hat{p})\hat{q}$$

where

$$\hat{p} = \frac{1}{N}\sum_{i=1}^{N} Q(y_i, 1) \text{ and } \hat{q} = \frac{1}{N}\sum_{i=1}^{N} Q\left(C(\mathbf{x}_i; \mathcal{L}_n), 1\right)$$

are the proportions of responses and predictions equaling 1. The over-fitting rate is given by

$$\hat{R} \quad = \quad \frac{\widehat{\text{Err}}^{(1)} - \widehat{\text{Err}}^{(a)}}{\hat{\gamma} - \widehat{\text{Err}}^{(1)}}$$

and the .632+ estimator is a weighted average of $\widehat{\text{Err}}^{(1)}$ and $\widehat{\text{Err}}^{(a)}$:

$$\widehat{\text{Err}}^{(.632+)} = (1 - \hat{w})\ \widehat{\text{Err}}^{(a)} + \hat{w}\ \widehat{\text{Err}}^{(1)} \text{ with } \hat{w} = \frac{.632}{1 - .368\hat{R}}.$$

7.3 The Out-of-Bag Estimator

An error rate estimator for C_B^{bagmain} (3.1) can be computed by using the out-of-bag sample. Every observation in the out-of-bag sample is classified by the classifier trained on the bootstrap sample and the predictions are aggregated over all bootstrap samples, in more detail:

a) Draw B random samples $\mathcal{L}_n^{*(1)}, \ldots, \mathcal{L}_n^{*(B)}$ of size N with replacement from $\mathcal{L}_n$.

b) Construct the classifier C using the bootstrap sample $\mathcal{L}_n^{*(b)}$. Predict the responses of all observations $\mathbf{x}_i$ in the out-of-bag sample $\mathcal{L}_n \setminus \mathcal{L}_n^{*(b)}$ by

$$\hat{y}_i^{(b)} = \begin{cases} C\left(\mathbf{x}_i; \mathcal{L}_n^{*(b)}\right) & \text{if } \mathbf{x}_i \in \mathcal{L}_n \setminus \mathcal{L}_n^{*(b)} \\ 0 & \text{otherwise.} \end{cases}$$

c) Iterate step 2) for all $b = 1, \ldots, B$ bootstrap samples.

For every observation $\mathbf{x}_i \in \mathcal{L}_n$ the predictions $\hat{y}_i^{(b)}$ are aggregated by majority voting over all $\hat{y}_i^{(b)} \neq 0$ for $b = 1, \ldots, B$. The aggregated prediction for observation i is denoted by $\hat{y}_i$. The out-of-bag estimate is the proportion

of incorrect aggregated predictions

$$\widehat{\text{Err}}^{(oob)} = \frac{1}{N}\sum_{i=1}^{N} Q(y, \hat{y}_i).$$

Breiman (2001a) argues that $\widehat{\text{Err}}^{(oob)}$ is biased upwards because only $B/3$ predictions are available for majority voting for each observation. Bylander (2002) proposes a correction for the two class problem.

Chapter 8

Glaucoma Classification by Laser Scanning Images

The problem of glaucoma classification for screening programs has lead to the development of the methodology presented in the previous Chapters. The aim of the application of this methodology is to develop and evaluate a classifier which can be used to decide whether a subject suffers glaucoma or not. The decision should be based on numerical measurements derived from a laser scanning examination of the eye background. Some details of the glaucoma disease are given in the first part of this Chapter. In the second part we focus on some technical details of the measurement device, a confocal laser scanner of the eye background. We also describe the predictors that are derived from such an image in order to classify subjects. Finally, the design of a case-control study, i.e. the buildup of a learning sample for the classifiers, is introduced.

8.1 Glaucoma

Glaucoma is an ocular disease which causes progressive damage in the optic nerve fibres and leads to visual field loss. This irreversible neurodegenerative disease is the second leading cause of blindness worldwide and is most common in elderly people (Coleman, 1999; Weih et al., 2001).

The loss of retinal nerves can be observed by examination of the morphology of the optic nerve head or the papilla. The optic nerve bundles the retinal nerves and connects the eye and the visual centre of the brain. It enters the eye at the papilla by passing through a hole in the sclera.

The diagnosis of glaucoma is based on examination of the visual field (perimetry), intra ocular pressure (IOP) and optic nerve head morphology. The latter can be examined by stereo optic disc photographs or optic nerve head tomography. Most published discriminant analysis models are based on visual field examinations. Logistic regression models were used for the classification of glaucoma based on visual field measurements by Hilton et al. (1996). The prediction of glaucomatous changes based on repeated visual field examinations by multivariate time series is discussed in Swift and Liu (2002). A comparison of some classifiers based on standard automated perimetry can be found in Chan et al. (2002).

However, before a visual field defect can be measured, loss and damage of the nerve fibre layer can be observed in the area of the optic nerve head (Mikelberg et al., 1995; Mardin et al., 1999). In the following we focus on the classification of glaucoma by measurements derived from laser

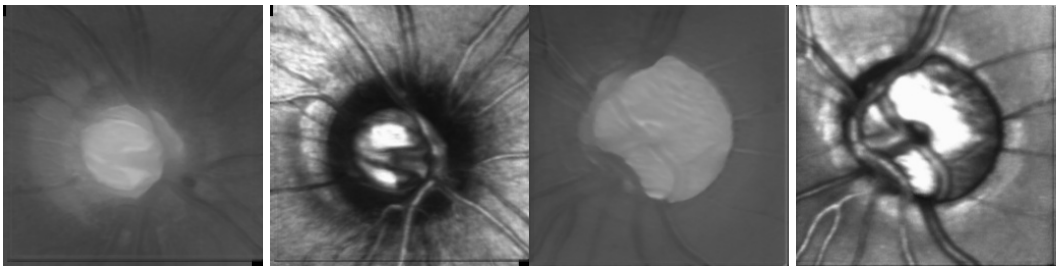
scanning images of the optic nerve head, using the Heidelberg Retina Tomograph (HRT, Heidelberg Engineering, 1997).

8.2 Laser Scanning of the Eye Background

The HRT is a non-invasive confocal scanning laser system for imaging of the eye background. The HRT provides a three-dimensional image computed from an image series of 32 images that are made from the different depth planes. The laser beam is focused on the examined retina point and the reflected light is detected by a photo diode. The retinal tissue is scanned and digitized in two dimensions sequentially.

Blood vessels and the surface of the papilla are visible in the HRT images. From the image series, the so-called topography image and the mean image can be computed. The topography image is a 2.5-dimensional image in which the pixels' grey values represent the depth coordinate (Figure 8.1, parts A and C). The gray-value of pixels on the mean image corresponds to the average intensity of three repeated laser scans (Figure 8.1, parts B and D). The important characteristics of the normal and glaucomatous eye can be observed in Figure 8.1, i.e. the excavation of the papilla due to loss of retinal nerve fibres.

In order to be able to classify a subject as either normal or glaucomatous, numerical predictors describing the morphology, i.e. the size, volume and depth of the excavation of the papilla, are derived from the laser scanning images. The examiner determines the margin of the papilla man-



A B C D

Figure 8.1: Topography (A) and mean image (B) of a normal and topography (C) and mean image (D) of a glaucomatous eye.

ually, which is called contour line. One measure of the size of the papilla is the area global, that is the area within the contour line. The thickness of the retinal nerve fibre layer itself cannot be measured by the HRT. However, the depth coordinate where the nerve fibre layer adjoins to the sclera is of special interest. The thickness of the nerve fibre layer is assumed as constant 50μm at a segment between -10 and -4 degrees temporal of the contour line. No blood vessels are located in this segment that could influence the height of the contour line. Therefore the reference plane, i.e. the location of the sclera, is defined as the mean height of the contour line in this small temporal segment minus 50μm (Burk et al., 2000). The area below reference as well as the volume below and above reference are used as additional features to describe the excavation of the papilla. The mean height of the retinal nerve fibre layer is computed with respect to the reference plan.

8.3 Case-Control Study – Design

The observations of this study (Mardin et al., 2002; Hothorn et al., 2003) include 98 glaucoma and 98 normal subjects from the "Erlangen Glaucoma Registry" of the Sonderforschungsbereich 539. The database was reimplemented using the `mysql` database engine (`http://www.mysql.org`) for two reasons: direct access to the raw images of the HRT examinations was needed and an interface to the R system exist for the `mysql` database engine (see Hothorn et al., 2001b, for details).

The data of the first examination of each subject were used in the study, whereas the diagnosis was based on the last examination whenever longitudinal data were available in order to take visual field long-time fluctuation into account. The study included 98 normal optic discs of 98 subjects with a median age of 56 years and 98 eyes of 98 chronic open-angle glaucoma patients with a median age of 56 years. Additional clinical characteristics of the normal and glaucoma group are given in Table 8.1. Per subject and patient, only one eye was selected. These eyes were the more advanced glaucomatous in patients and the better eye with the better visual field performance in normals. Glaucoma patients and normal subjects were assessed by clinical examination, visual field evaluation (perimetry) and optic nerve head analysis.

A glaucomatous visual field defect was clinically defined when the mean visual field defect (MD) was higher than 2.8 dB and corrected loss variance (CLV) was greater than 4.0 dB^2. The subjects of the normal

	Normal	Glaucoma	P-value
Total Number (n)	98	98	-
Age (years)	56 (50, 61)	56 (50, 61)	-
Gender (F/M)	57/41	57/41	-
Area global (mm^2)	2.46 (2.07, 3.07)	2.54 (2.21, 2.87)	0.577
Maximum IOP (mmHg)	20 (16, 25)	22 (20, 28)	0.001
IOP during examination (mmHg)	16 (15, 19)	16 (14, 19)	0.792
Visual Field corrected loss variance (CLV, dB^2)	1.35 (0.5, 2.3)	31.4 (9.5, 59.4)	< 0.001
Visual field mean defect (MD, dB)	1.4 (0.4, 2.7)	6.4 (4.52, 0.65)	< 0.001

Table 8.1: Clinical characteristics. For the continuous variables, the median as well as the first and third quartile in parentheses are given. The *P*-values of the Wilcoxon statistic are reported as a measure of separation between the two groups.

group were either recruited from the administrative staff of the hospital, being examined for a routine ocular check-up examination, or came to the hospital for exclusion of glaucoma due to large optic discs or suspected ocular hypertension.

Age is known to be a major risk factor for glaucoma (e.g. Coleman, 1999). To prevent bias by age, cases and controls were matched by age (maximum difference: 1 year) and additionally by gender. The normal group was defined as subjects with normal optic discs and visual fields.

Subjects with elevated intra ocular pressure (IOP) higher than 21 mmHg or a large papilla (macro papilla) were included in the normal group. The glaucoma group includes patients with primary open-angle glaucoma and patients with normal pressure glaucoma.

Chapter 9

Simulation Experiments for Glaucoma Classification

For discriminant analysis problems with small learning samples, Hothorn and Lausen (2003a) suggest and apply a general strategy to avoid a method selection bias by using problem specific simulation models for the comparison and selection of classifiers. Since the learning sample for glaucoma classification as defined in Section 8.3 is too small for the training and selection of classifiers as well as error rate estimation, we use a simulation model of the HRT measurements to investigate the performance of linear and tree based classifiers for glaucoma based on the morphology of the papilla. Moreover, the accuracy of three estimators of the misclassification error given in Chapter 7 is investigated.

9.1 Simulation Model

The model introduced by Swindale et al. (2000) is used to describe the surface of the optic nerve head:

$$\mu_\beta(u,v) = \\ -\left(\frac{w}{1+e^{(r-r_0)/s}} + a(u-u_0) + b(v-v_0) + c(u-u_0)^2 + d(v-v_0)^2 + w_0\right)$$

with $r = \|(u-u_0, v-v_0)\|_2$, the usual 2-norm, and parameter vector

$$\beta = (w, w_0, r_0, s, u_0, v_0, a, b, c, d).$$

For the interpretation of the different parameters we refer to the original paper. The arguments u and v vary between 0 and 3mm, which is the area scanned by the HRT. The parameter vector $\hat{\beta}_{\text{norm}}$ is estimated from the normal subjects in the study and $\hat{\beta}_{\text{glau}}$ is estimated from the glaucoma subjects. The estimation of the parameter vectors from the case-control study introduced in Section 8.3 helps to achieve a more realistic model. For our set of observations the estimated parameter vectors are given in Table 9.1.

Progression in glaucoma is derived from the weighted average of the

β	w	w_0	r_0	s	u_0	v_0	a	b	c	d
$\hat{\beta}_{\text{norm}}$	0.75	0.80	0.54	0.12	1.42	1.44	0.05	0.03	0.05	-0.05
$\hat{\beta}_{\text{glau}}$	0.69	0.80	0.66	0.10	1.42	1.44	-0.01	0.01	-0.01	-0.07

Table 9.1: Simulation Model. The parameters are estimated from the normal and glaucoma population described the previous Chapter.

normal and glaucoma surfaces

$$\mu_\delta(u,v) = \delta\mu_{\hat{\beta}_{\text{glau}}}(u,v) + (1-\delta)\mu_{\hat{\beta}_{\text{norm}}}(u,v)$$

with $\delta \geq 0$ and $(u,v) \in [0,3]^2$. Figure 9.1 shows the surface of the optic nerve head for a normal and a glaucomatous eye.

Given a surface μ_δ with progression δ we simulate the predictors derived by the HRT as follows. The surface is computed on a grid $\mathbf{M}$ over $[0,3]^2$:

$$\mathbf{M} = \left\{ \frac{3i}{k}, i = 0, \ldots, k \right\}^2, k = 40.$$

To model the measurement error by the HRT we add independent and identically distributed (iid) normal errors with variance σ^2 to the surface at each point of the grid $\mathbf{M}$

$$x_\mu(u,v) \overset{\text{iid}}{\sim} \mathcal{N}(\mu_\delta(u,v), \sigma^2), \quad (u,v) \in \mathbf{M},$$

where $\mathcal{N}$ denotes the normal distribution. The papilla is modeled as the set of all points on the grid $\mathbf{M}$ within a circle of radius ρ around the centre (u_0, v_0):

$$\mathbf{Pa} = \left\{ (u,v) \in \mathbf{M} \quad \middle| \quad \|(u-u_0, v-v_0)\|_2 \leq \rho \right\},$$

where ρ is estimated from the study population. The following subsets of the papilla are defined by four sectors of 90 degrees each (temporal,

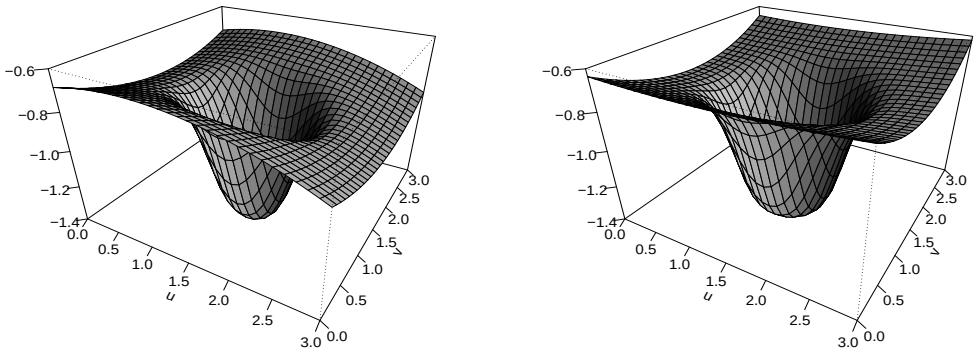


Figure 9.1: Surface of the optic nerve head model for normal (left, $\delta = 0$) and glaucomatous (right, $\delta = 1$) eyes with parameters estimated from the study population. The excavation of the papilla of the glaucomatous eye is wider and the border is flat compared to the normal papilla.

superior, inferior and nasal):

$$\begin{aligned}
\mathbf{Pa}_{\text{temp}} &= \left\{(u,v) \in \mathbf{Pa} \;\middle|\; u > v \wedge (3-u) > v\right\} \\
\mathbf{Pa}_{\text{nasal}} &= \left\{(u,v) \in \mathbf{Pa} \;\middle|\; u < v \wedge (3-u) < v\right\} \\
\mathbf{Pa}_{\text{inf}} &= \left\{(u,v) \in \mathbf{Pa} \;\middle|\; u > v \wedge (3-u) < v\right\} \\
\mathbf{Pa}_{\text{sup}} &= \left\{(u,v) \in \mathbf{Pa} \;\middle|\; u < v \wedge (3-u) > v\right\}.
\end{aligned}$$

The reference plane of the HRT is defined by x_{ref}, the mean height on the

contour line temporal minus 0.5mm (cf. Section 8):

$$x_{\text{ref}} = \left(|\mathbf{Pa}_c \cap \mathbf{Pa}_{\text{temp}}|^{-1} \sum_{(u,v)\in \mathbf{Pa}_c \cap \mathbf{Pa}_{\text{temp}}} x_\mu(u,v) \right) - 0.5.$$

$|\cdot|$ denotes the cardinality of a set and $\mathbf{Pa}_c$ is the contour line around the papilla including 2 cells of the grid

$$\mathbf{Pa}_c = \left\{ (u,v) \in \mathbf{Pa} \quad \middle| \quad \rho - 2\frac{3}{k} \leq \|(u-u_0, v-v_0)\|_2 \leq \rho \right\}.$$

The contour line is used to delimit the papilla from the rest of the image.

The following predictors are used by the HRT to describe the surface of the optic nerve head: Volume above reference x_{Var}, volume below reference x_{Vbr}, rim area x_{Ar}, area global x_{A}, mean height in contour x_{Mhc} and third moment x_{Tm}. Similar measures in the simulation model are defined either globally or within one of the four sectors $\mathbf{Pa}_{\text{temp}}, \mathbf{Pa}_{\text{nasal}}, \mathbf{Pa}_{\text{sup}}$ and $\mathbf{Pa}_{\text{inf}}$ as follows:

$$x_{\text{A}} \quad = \quad |\mathbf{Pa}| \, (3/k)^2$$

is the size of the papilla, i.e. the area within the contour line $\mathbf{Pa}_c$,

$$x_{\text{Ar}}(\mathbf{Q}) \quad = \quad \sum_{(u,v)\in\mathbf{Q}} I\left(x_\mu(u,v) > x_{\text{ref}}\right) (3/k)^2$$

describes the area above the contour line,

$$x_{\text{Var}}(\mathbf{Q}) \quad = \quad \sum_{(u,v)\in\mathbf{Q}} (x_\mu(u,v) - x_{\text{ref}}) I\left(x_\mu(u,v) > x_{\text{ref}}\right) (3/k)^2$$

measures the volume of the cup above the reference plane x_{ref},

$$x_{\text{Vbr}}(\mathbf{Q}) \quad = \quad \sum_{(u,v)\in\mathbf{Q}} (x_\mu(u,v) - \min_{(u,v)\in\mathbf{Pa}} (x_\mu(u,v)) I\left(x_\mu(u,v) \leq x_{\text{ref}}\right) (3/k)^2$$

is the volume of the cup below the reference plane x_{ref},

$$x_{\text{Mhc}}(\mathbf{Q}) = |\mathbf{Q} \cap \mathbf{Pa}_c|^{-1} \sum_{(u,v) \in \mathbf{Q} \cap \mathbf{Pa}_c} (x_\mu(u,v) - x_{\text{ref}})$$

gives the mean height of the surface at the contour line and

$$x_{\text{Tm}}(\mathbf{Q}) = |\mathbf{Q}|^{-1} \sum_{(u,v) \in \mathbf{Q}} \left(x_\mu(u,v) - |\mathbf{Q}|^{-1} \sum_{(u,v) \in \mathbf{Q}} x_\mu(u,v) \right)^3$$

is a measurement of the slope of the cup. The rim area x_{A} is only measured globally, the remaining measurements $x_{\text{Ar}}(\mathbf{Q})$, $x_{\text{Var}}(\mathbf{Q})$, $x_{\text{Vbr}}(\mathbf{Q})$, $x_{\text{Mhc}}(\mathbf{Q})$ and $x_{\text{Tm}}(\mathbf{Q})$ are computed from the noisy surface for either the papilla $\mathbf{Pa}$ or one of the four sectors

$$\mathbf{Q} \in \{\mathbf{Pa}_{\text{temp}}, \mathbf{Pa}_{\text{nasal}}, \mathbf{Pa}_{\text{sup}}, \mathbf{Pa}_{\text{inf}}\}.$$

Therefore, 26 predictors are available for each observation in our simulation setup.

9.2 Simulation Setup

In order to investigate the performance of possible candidates for glaucoma classification, the features derived from laser scanning image data are simulated by using the model of the optic nerve head morphology introduced in the previous Section.

We are primarily interested in a comparison of linear and tree based classifiers. Therefore, linear discriminant analysis (LDA), classification trees (CTREE, Breiman et al., 1984) and bagging of classification trees (Breiman, 1996a, 1998) are used as classifiers. As a classifier that combines

linear and tree based classifiers we choose bundling of LDA ("double-bagging", Section 3.2).

Classification trees are computed with the package `rpart`, version 3.1-3 (Therneau and Atkinson, 1997). The tree growing is stopped if a node contains less than 20 observations. This choice is in the order of $\sqrt{N}$. Additionally, we stop the tree building if none of the possible splits decreases the overall misclassification error by a factor of at least 0.01. That is, every split must lead to at least two additional correctly classified observations.

The first setup simulates the measurement error of the HRT itself for repeated examination of one normal and one glaucomatous eye. Three levels of progression for the glaucomatous eye are used: $\delta = 0.1, 0.5$ and 0.9. The measurement error is fixed at $\sigma = 0.2$.

The second setup is more suitable for comparing the performance of classifiers in a human population with varying morphology of the optic nerve head and simulates a population with three subgroups with respect to the area global, i.e. the size of the papilla (small: 1.72 mm^2, medium: 2.61 mm^2 and large: 4.01 mm^2). Furthermore, we assume that the contour line determined by the examiner is too wide in small papillae and too narrow in large papillae, each by an amount of 10% of the radius ρ. Again, the measurement error is fixed at $\sigma = 0.2$ and the level of progression varies for $\delta = 0.1, 0.5$ and 0.9.

Finally, we study the influence of the measurement error of the HRT on the classification of glaucoma for a clinical population. In the third setup we partition every subgroup with respect to the size of the area global into

three equally sized subgroups with respect to the level of progression $\delta = 0.2, 0.5$ and 0.8. The measurement error of the HRT varies for $\sigma = 0.1, 0.2$ and 0.3 in this situation.

It should be noted that both σ (standard deviation of the measurement error) and δ (level of progression) determine the distance between given class means. Consequently, the simulated error rates depend only on the ratio σ/δ. The classifiers are trained using a learning sample of size 200 (100 normal and 100 glaucoma surfaces) and tested using a test sample of the same size. We use 1000 Monte-Carlo replications.

9.3 Comparison of Classifiers

In the first setup, where the repeated examination of two eyes is simulated, LDA performs best. Bagging suffers a much higher error rate but improves the even larger error rate of CTREE. It may be assumed that the two classes are separable by linear combinations of the predictors. Bundling improves bagging significantly and its error rate is comparable to LDA. This result illustrates the intention of a combination of LDA and bagging: bundling "inherits" the properties of LDA when the classes are separable by a hyperplane. Note that for progression $\delta = 0.9$ LDA and bundling classify all observations correctly while bagging has a misclassification error rate of 6.3%.

CTREE outperforms LDA in the second setup, which simulates subjects with varying morphology of the optic nerve head in both groups.

Classification trees are able to identify the subgroups with respect to the size of the optic disc. Bagging reduces the simulated misclassification error of CTREE significantly. Bundling has the lowest misclassification error among all studied classifiers in this situation.

For the third setup, bundling attains the lowest simulated error rates, too. The subgroups with respect to both the size of the optic disc and progression are identified by classification trees.

To summarize the outcome of the simulation studies, we observe the same results as for the benchmark experiments in Chapter 6: the classifier which combines LDA and bagging performs at least comparable to the best of both methods or even leads to an improvement with respect to misclassification error.

While the linear classifier performs best for the repeated examination of a fixed morphology, the tree based classifiers are able to identify clinical subgroups, which are likely to be a problem in clinical setups. In fact, difficulties with linear classifiers in the presence of clinical subgroups with respect to the size of the papilla have been described in Iester et al. (1997). Bundling is able to adapt itself to the structure of the learning sample and its misclassification error is comparable the best classifier: either LDA or bagging.

	Setup 1			
δ	LDA	CTREE	Bagging	Bundling
0.1	0.376	0.470	0.452	0.431
0.5	0.027	0.223	0.158	0.031
0.9	0.000	0.110	0.063	0.000
	Setup 2			
δ	LDA	CTREE	Bagging	Bundling
0.1	0.443	0.459	0.440	0.437
0.5	0.246	0.216	0.155	0.142
0.9	0.147	0.137	0.077	0.059
	Setup 3			
σ	LDA	CTREE	Bagging	Bundling
0.3	0.337	0.340	0.298	0.288
0.2	0.278	0.271	0.216	0.206
0.1	0.178	0.147	0.101	0.101

Table 9.2: Model of glaucoma classification: Simulated error rates in three setups for linear discriminant analysis (LDA), classification trees (CTREE), bagging of classification trees and bundling of LDA and CTREE.

	Err	$\widehat{\text{Err}}^{(oob)}$	$\widehat{\text{Err}}^{(cv10)}$	$\widehat{\text{Err}}^{(.632+)}$
	Setup 1, $\delta = 0.5$			
Exp	0.168	0.178	0.173	0.164
RMS	-	0.026	0.026	0.016
Bias	-	0.010	0.005	-0.004
SD	-	0.024	0.026	0.015
	Setup 2, $\delta = 0.5$			
Exp	0.168	0.184	0.196	0.176
RMS	-	0.029	0.041	0.018
Bias	-	0.015	0.028	0.008
SD	-	0.025	0.030	0.017
	Setup 3, $\sigma = 0.1$			
Exp	0.107	0.121	0.137	0.111
RMS	-	0.024	0.039	0.015
Bias	-	0.014	0.030	0.004
SD	-	0.020	0.024	0.014

Table 9.3: Error rate estimators. Characteristics of 10-fold cross-validation, .632+ and out-of-bag estimator. Expectation (Exp), standard deviation (SD), bias and root-mean-squared error (RMS) are given.

9.4 Comparison of Error Rate Estimators

The error rate estimators are investigated in three selected parameter configurations: progression $\delta = 0.5$ in setup 1 and 2 and measurement error $\sigma = 0.1$ in setup 3. The error rate Err for each of the three situations is simulated and reported in Table 9.3.

Additional 100 learning samples of size 200 are created and the error rate is estimated for each of those 100 learning samples by $\widehat{\text{Err}}^{(cv10)}$, $\widehat{\text{Err}}^{(.632+)}$ and $\widehat{\text{Err}}^{(oob)}$. The mean, standard deviation, bias and root-mean-squared error (RMS) are given in Table 9.3. With respect to RMS, bias and standard deviation, the .632+ estimator performs best. The cross-validated error and the out-of-bag estimator suffer a higher standard deviation. The bias of the out-of-bag estimator is obvious.

Chapter 10

Case-Control Study – Results

The case-control study introduced in Section 8.3 is designed for the construction of classifiers for glaucoma screening with a binary outcome. The aim is to classify an eye as either "normal" or "glaucomatous" based on predictors describing the morphology of the papilla only. Those predictors are derived from a confocal laser scanning device (HRT, Section 8.2), since early glaucomatous changes can be detected by this examination.

The learning sample consists of 62 predictors (cf. Appendix B) for 196 observations and is therefore too small for learning, selection and error rate estimation of different classifiers. We therefore used a simulation model of the papilla to investigate the performance of linear and tree based classifiers for glaucoma diagnosis. From the investigations in Chapter 9 we can conclude that a combination of classifiers via bundling performs always comparably to the best of the single classifiers, while the performance of the single classifiers depends on the situation under test. More-

over, the .632+ estimator performed best with respect to bias and variance and therefore will be used for estimating the misclassification error.

Although we used bundling with LDA only for our simulation experiments, we choose bundling of sLDA, nearest neighbors (k-NN, $k = 5, k = 10$) and the logistic regression model as a more general classifier for glaucoma classification. In contrast to the simulation and benchmark experiments, 100 trees are used for bagging, bundling and random forests.

The estimated misclassification error of bundling is 11%. This estimate is honest in the following sense: it does not suffer any method or variable selection bias, because no data driven method or variable selection procedure has lead to the choice of this classifiers. Moreover, the .632+ bootstrap is an unbiased estimator of misclassification error. For the sake of comparison, Table 10.1 reports the estimated misclassification errors of the single classifiers and random forests. The classical LDA clearly suffers a dimension problem. Moreover, some of the predictors are by definition linear combinations of others, causing problems with multicollinearity. Obviously, the stabilized LDA does not suffer those problems and leads to a substantial improvement compared with the classical LDA. The misclassification error of bagging is much lower than the error of any of the classical procedures. Random forests perform comparably to bundling. The results are consistent with the results of the simulation studies in Chapter 9. Since patients suffering normal tension glaucoma and normals with ocular hypertension or macro papilla are included in the glaucoma or normal group of the study, the good performance of tree based classifiers (bag-

LDA	sLDA	5-NN	10-NN	LR	Bagg	Bund	RF
0.229	0.161	0.207	0.201	0.173	0.137	0.11	0.114

Table 10.1: Glaucoma classification: .632+ estimator (50 bootstrap replications) for LDA, stabilized LDA (sLDA), k nearest neighbors (k-NN, $k = 5, k = 10$) and the logistic regression model (LR). Bagging (Bagg), bundling (Bund) and random forests (RF) were computed using 100 trees each.

ging, bundling and random forests) is most likely due to their ability to identify those clinical subgroups.

Chapter 11

Software

In this Chapter we present some details of an implementation of bundling in the R system for statistical computing. Moreover, we discuss our experiences with parallel statistical simulations using a Linux cluster with the Mosix kernel extension.

11.1 The R System

The R system is basically an open source implementation of the S language developed at AT&T Bell Laboratories by Rick Becker, John Chambers and Allan Wilks (Becker et al., 1988; Chambers and Hastie, 1992).

The core of R is an interpreted computer language which borrows some concepts from Scheme (e.g. Abelson et al., 1996). The language allows branching and looping as well as modular programming using functions. It is possible for the user to interface to procedures written in C, C++ or Fortran for efficiency.

R has a powerful packaging system which provides efficient tools for maintaining code, documentation, examples as well as for quality testing. A huge set of packages is distributed on the Comprehensive R Archive Network. Bundling as proposed in this work is implemented in the `ipred` package (Peters et al., 2002), the software is released under General Public License and can be downloaded from `http://CRAN.R-project.org`. Details of the implementation are given in the next Section.

11.2 Bundling in R

Trees for nominal, continuous and censored responses are available in the R system via the `rpart` package (Therneau and Atkinson, 1997). The `rpart` function can easily be used for bagging classification trees: simply call `rpart` for bootstrap samples of the learning sample and concatenate the resulting `rpart` objects into a list. The prediction of a new observation is easy, too. Predict the class of the new observation for each tree in the list and aggregate the predictions by majority voting (for example using `table`).

Two main difficulties arise. For bundling, we need to compute arbitrary, user-specified additional classifiers for each out-of-bag sample and compute their predictions both for the bootstrap sample as well as for any new observation to classify. Therefore, we need to save the additional classifiers for each bootstrap sample. Another major problem is speed. Calling `rpart` 50 times, say, leads to repeated unnecessary computations:

formula evaluation, determination of the measurement scale for each predictor and so on. Unfortunately, there is a trade-off between a flexible implementation and speed. We therefore decided to speed up bagging by a modification of the `rpart` routine and to generalize bundling at the price of efficiency.

The implementation of the `rpart` routine currently does not separate the evaluation of formula objects and the construction of appropriate design matrices from the tree construction itself which leads to unnecessary computations if multiple trees are constructed for reweighted observations in the learning sample. Therefore, the `ipred` package implements a modified version called `irpart` which grows multiple trees without reevaluating formula objects in order to save computing time.

Both bagging and bundling are implemented in the generic `ipredbagg` which dispatches on the class of the response: methods for factors (classification) and numeric responses (regression) as well as responses of class `Surv` (censored data) currently exist. A formula based interface to `ipredbagg` is offered by `bagging`, a generic itself which dispatches on the `data` argument.

```
bagging(formula, data, subset, na.action, ...)
```

Currently only a method for data frames is implemented.

As mentioned in the previous Section, the interface to bundling was designed to allow users to specify additional classifiers in a flexible and easy way. Basically, for each classifier a list with two elements is required: `model` and `predict`, where `model` specifies a function for training of the

classifier and `predict` is a function for computing predictions. We require at least two arguments for `model`: `formula` and `data`. For `predict`, exactly two arguments are allowed: `object` and `newdata`. If more than one additional classifier is to be used, a list of lists with `model` and `predict` elements can be defined for bundling via the `comb` argument.

For the experiments in this work, we used the values of the linear discriminant variables of a stabilized linear discriminant analysis, the predicted classes of nearest neighbors ($k = 5$ and $k = 10$) and the estimated conditional class probability derived from the logistic regression model as additional predictors. The associated list can be defined along the following lines.

```
R> mybundle <- list(
    # stabilized LDA
    list(model=slda, predict=function(object, newdata)
                       predict.slda(object, newdata)$x),
    # 5-NN
    list(model=function(...) ipredknn(..., k=5),
         predict=predict.ipredknn),
    # 10-NN
    list(model=function(...) ipredknn(..., k=10),
         predict=predict.ipredknn),
    # LR or multinomial model, resp.
    list(model=function(...) multinom(...,trace=FALSE),
         predict=function(obj, newdata)
           predict.multinom(obj, newdata, type="prob"))
  )
```

The function `ipredknn` is a formula based interface to `knn` in package `class` from the `VR` bundle (Venables and Ripley, 1999).

For each bootstrap sample, the additional classifiers are trained and

one single function `bfct` for prediction is created in the environment of the current bootstrap sample. Every time `bfct(newdata)` is called, the predictions of the additional classifiers are computed in the corresponding environment ("lexical scoping", Gentleman and Ihaka, 2000) and an explicit knowledge of those objects in not needed.

The resulting user-interface is very compact. Bundling of sLDA, nearest neighbors and the logistic regression model using 100 bootstrap sample for the data from the case-control study (Chapter 8) can be computed along the following lines of code.

```
R> library(ipred)
R> data(GlaucomaM)
R> mod <- bagging(Class ~ ., data=GlaucomaM, comb=mybundle,
                  nbagg=100)
```

The function `bagging` returns an object of class `classbagg` (for classification problems) which stores all the information necessary to predict the class of a new observation via a method to the generic `predict`.

```
R> predict(mod, newdata=newobs)
  [1] normal   normal   normal   glaucoma ...
```

The complete manual page for `bagging` is given in Appendix A. The function `errorest` can be used to estimate the misclassification error, for example by the .632+ bootstrap, the results for the case-control study are given in Table 10.1.

```
R> paraest <- control.errorest(nboot=50)
R> errorest(Class ~ ., data=GlaucomaM, model=bagging,
            nbagg=100, comb=mybundle,  est.para=paraest,
            estimator="632plus")
```

```
Call:
errorest.data.frame(formula = Class ~ ., data = GlaucomaM,
    model = bagging, estimator = "632plus", est.para = paraest,
    nbagg = 100, comb = mybundle)

        .632+ Bootstrap estimator of misclassification error
        with 50 bootstrap replications

Misclassification error:  0.1098
```

11.3 Parallel Statistical Simulations

The training and evaluation of resampling based classifiers like bagging and bundling in simulation studies and benchmark experiments is time consuming, even with up-to-date hardware. Moreover, there is a trade off between a flexible and efficient implementation as discussed in the previous Section. However, the multiple trees in bagging or bundling are, by definition, independent of each other. The same is true for statistical simulations or the estimation of misclassification error by the average of ten runs of 10-fold cross-validation. Therefore, the computations can easily run on multiple processors or nodes in a cluster in a parallel way.

The `rpvm` package (Li and Rossini, 2001) provides an interface to PVM (Parallel Virtual Machine, `http://www.csm.ornl.gov/pvm/pvm_home.html`), which allows R users to spawn separate R processes from within R. This approach requires direct modifications to the R code. One alternative is to use a cluster of PC-nodes running Linux with the Mosix kernel ex-

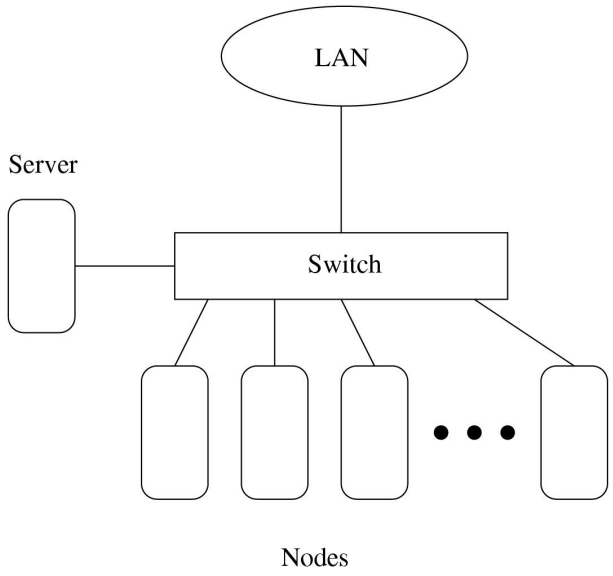


Figure 11.1: Mosix-Cluster: A 100MBit-Switch substitutes the system bus and connects the cluster to the local area network (LAN).

tension as we did for our simulations. The Mosix system (Barak and La'adan, 1998, http://www.mosix.org) monitors processes and distributes them across all registered nodes in a cluster to achive an optimal performance. Running simulations in a parallel way is simple: start different R processes and the Mosix system will distribute them among the available nodes. Because the network serves as a "system bus", a separate switch is used to connect all nodes in a cluster. We build up a cluster using one server and 14 diskless nodes, see Figure 11.1. The system directories of the nodes are mounted from the server by `nfs-root`. Beside the saving of the harddisk costs, only the server needs to be maintained and new nodes can be added to the cluster without much effort.

However, two problems arise. To enable R processes to migrate, R's

stack handling needs to be modified in order not to save the signal context. In the R sources, the `SETJMP` macro in the header file `src/include/Defn.h` should by modified to

```
# define SETJMP(x) sigsetjmp(x,0)
```

instead of `sigsetjmp(x,1)`. Since the nodes are running diskless, there is no local swap space available and processes can terminate when it is impossible to allocate memory.

Chapter 12

Summary and Outlook

The medical application which has lead to the development of the combined classifiers presented in this work, namely the early detection of glaucoma based on laser examinations of the eye background, is rather typical for many discriminant analysis problems in biostatistics or other fields. A large set of possible predictors, maybe highly correlated, shall be used to predict the class of a new observation. However, the learning samples for the construction of classification rules are often rather small compared to the number of possible predictors. In our case, the learning sample consists only of 196 observations but 62 predictors.

In this setup, two main questions arise in classical statistics: "Which (small) subset of the predictors is informative?" and "Which model fits the data best?". Consequently, variable selection and model selection strategies are applied. A common problem is that the estimate of the misclassification error after both variable and method selection leads to overly

optimistic estimates of the true misclassification error.

Moreover, the "best" classifier for a discriminant analysis problem may not even exist. For example, clinical subgroups with different characteristics of a disease are likely to occur. This is a problem especially for glaucoma, a disease with many known and unknown variants. Our simulations, using a model of the optic nerve head for different clinical populations, support this assumption. From the investigations in Chapter 8 we can conclude that neither a linear nor tree based classifier always performs best. Even bagging of classification trees is outperformed by LDA in simple setups, whereas the misclassification error of LDA is much higher when clinical subgroups with respect to the size of the optic nerve head are involved. Therefore, the choice of one single classifier strongly depends on the situation under test.

Bundling of different classifiers by bagging trees as suggested in this work is a proposal to deal with these problems. Because arbitrary classifiers are combined ("bundled") and implicitly selected by classification trees in addition to the original predictors, the need for an explicit method selection disappears. Benchmark and simulation experiments show that bundling outperforms any of the single classifiers for the majority of examples. Especially for the data from the case-control study on glaucoma, bundling performs at least comparably to either linear or tree based classifiers regardless of the setup investigated. In this sense, the performance of the combined classifier is "robust" with respect to different situations. Our results were reproduced by an independent and extensive benchmark

study (Meyer et al., 2003) whose results support our conclusions.

The estimated misclassification error for bundling of stabilized LDA, nearest neighbors and the logistic regression model for glaucoma classification using 100 trees is 11%. This estimate does not suffer any method or variable selection bias and is a substantial reduction of the error rate for glaucoma classification based on numerical measurements of the eye morphology only.

Because bundling uses aggregated predictions over multiple trees which were trained on bootstrap replications of the original learning sample, it is stable in the sense that a large number of predictors can be used without variable selection as for example in bagging and random forests (Breiman, 1996a, 1998, 2001a).

Bundling contradicts the classical principles of dimension reduction: we do not restrict the number of predictors but add data based transformations of the predictors to the set of original predictors. Bundling is a successful representative of procedures which incorporate the "... information in various combinations of the predictor variables ..." in order to construct better classifiers (Breiman, 2001b).

Modifications to the bundling procedure may lead to further improvements with respect to misclassification error but were not investigated until now. One may think of using only a small subset of the predictors for training of the additional classifiers. This is an approach to overcome numerical instabilities due to a large number of variables and a small number of observations in the out-of-bag sample in addition to the use of a stabi-

lized LDA and subundling as discussed in Section 4. In combination with random forests, a randomly selected subset of the predictors could be offered to the classification trees only. Currently, we compute the additional classifiers one time for each bootstrap sample. Computing the additional classifiers for each node of a tree may improve the performance especially when subgroups are inherent in the data. This could be done as follows. For each node, compute the additional classifiers based on the out-of-bag observations identified by the current node and use their predictions for the observations in the current node as additional predictor for building the daughter nodes. However, the computational effort is much larger compared with bundling.

The strong risk consistency of partition based classifiers (Lugosi and Nobel, 1996) is used as a tool for the proof of the strong risk consistency of bundling. Unfortunately, the proof is rather technical and does not lead to any insight why or how the combination of classifiers work. An interesting result is the strong risk consistency of bagging of any strongly risk consistent partition based classifier. The proof is based on the fact that the averages of the conditional class probability estimators derived from classifiers trained on bootstrap samples are L_1 consistent estimators of the conditional class probability.

Since the aim of our investigations was the development of decision systems for glaucoma screening, the performance of any classifier was measured by its misclassification error only. The exploration and interpretation of a classifier as a model for the unknown functional relation-

ship between predictors and classes were therefore not in the focus of our work.

However, classification trees are popular in medical statistics because a graphical representation of the tree seems to be interpretable. The use of trees as "models" has been criticized for various reasons (for example Marshall, 2001). The most serious one, from our point of view, is that classification and regression trees as suggested by Breiman et al. (1984) are not statistical models in the sense that they do not control the error of selecting any non-informative predictor. Trees tend to select predictors with many possible cutpoints and the "right sized" tree is usually determined by resampling procedures like pruning. Recently, Nobel (2003) showed the risk consistency of a complexity based pruning scheme (see Scheffer, 1999, for a more general theoretical framework for the analysis of the bias induced by optimization within a special classifier). One alternative to pruning are P-value adjusted classification and regression trees (Lausen et al., 1994). This approach solves the variable selection problem and provides a statistical stopping criterion by using adjustments based on the P-value of maximally selected rank statistics (Lausen and Schumacher, 1992). A test for the null hypothesis of independence between the response and any of the predictors under a simple cutpoint model is performed and the tree growing is stopped if the P-value of the test exceeds a predefined level. Because the node size shrinks, the use of an exact bound of the null distribution of maximally selected rank statistics (Hothorn and Lausen, 2003c) may be useful. In addition to the use of maximally selected rank statistics,

an adjustment for the number of possible predictors tested in each node of a tree is desirable. A test procedure based on efficient numerical evaluations of the multivariate normal probability (Genz, 1992; Hothorn et al., 2001a) is given in Lausen et al. (2002).

One approach to the comparison and evaluation of different classifiers in a standardized partition space is suggested by Weihs and Sondhauß (2003). Since this approach is applicable to arbitrary classifiers, it promises a possibility to look at the way how bundling for glaucoma classification works.

Although this work is devoted to discriminant analysis problems only, the basic idea of bundling can be extended to regression problems as well. For example, the coefficients of a linear model can be estimated using the out-of-bag sample and the predictions on the bootstrap sample can be used as an additional predictor for regression trees. For censored responses, the linear predictor of a Cox model can be incorporated by the same procedure into bagging of survival trees (Hothorn et al., 2002). In contrast to bagging of classification trees, where the trees are grown until the nodes are pure, it is not obvious when to stop the tree growing for bagging of regression or survival trees. Breiman (1996a) used the out-of-bag observations to prune the regression trees for bagging. The use of P-value adjusted regression trees may be a way to control the size of the multiple trees. However, this requires further investigations.

Appendix A

Manual Page

`bagging` *Bagging Classification, Regression and Survival Trees*

Description

Bagging for classification, regression and survival trees.

Usage

```
ipredbagg.factor(y, X=NULL, nbagg=25, control=
                 rpart.control(minsplit=2, cp=0, xval=0),
                 comb=NULL, coob=FALSE, ns=length(y),
                 keepX = TRUE, ...)
ipredbagg.numeric(y, X=NULL, nbagg=25, control=
                  rpart.control(xval=0),
                  comb=NULL, coob=FALSE, ns=length(y),
                  keepX = TRUE, ...)
ipredbagg.Surv(y, X=NULL, nbagg=25,
               control=rpart.control(xval=0),
               comb=NULL, coob=FALSE, ns=dim(y)[1],
               keepX = TRUE, ...)
bagging(formula, data, subset, na.action=na.rpart, ...)
```

Arguments

`y`	the response variable: either a factor vector of class labels (bagging classification trees), a vector of numerical values (bagging regression trees) or an object of class `Surv` (bagging survival trees).
`X`	a data frame of predictor variables.
`nbagg`	an integer giving the number of bootstrap replications.
`coob`	a logical indicating whether an out-of-bag estimate of the error rate (misclassification error, root mean squared error or Brier score) should be computed. See `predict.classbagg` for details.
`control`	options that control details of the `rpart` algorithm, see `rpart.control`. It is wise to set `xval = 0` in order to save computing time. Note that the default values depend on the class of `y`.
`comb`	a list of additional models for model combination, see below for some examples. Note that argument `method` for double-bagging is no longer there, `comb` is much more flexible.
`ns`	number of sample to draw from the learning sample. By default, the usual bootstrap n out of n with replacement is performed. If `ns` is smaller than `length(y)`, subagging (Bühlmann and Yu, 2002), i.e. sampling `ns` out of `length(y)` without replacement, is performed.
`keepX`	a logical indicating whether the data frame of predictors should be returned. Note that the computation of the out-of-bag estimator requires `keepX=TRUE`.
`formula`	a formula of the form `lhs ~ rhs` where `lhs` is the response variable and `rhs` a set of predictors.
`data`	optional data frame containing the variables in the model formula.
`subset`	optional vector specifying a subset of observations to be used.
`na.action`	function which indicates what should happen when the data contain `NA`s. Defaults to `na.rpart`.
`...`	additional parameters passed to `ipredbagg` or `rpart`, respectively.

Details

Bagging for classification and regression trees were suggested by Breiman (1996a, 1998) in order to stabilize trees.

The trees in this function are computed using the implementation in the rpart package. The generic function ipredbagg implements methods for different responses. If y is a factor, classification trees are constructed. For numerical vectors y, regression trees are aggregated and if y is a survival object, bagging survival trees (Hothorn et al, 2002) is performed. The function bagging offers a formula based interface to ipredbagg.

nbagg bootstrap samples are drawn and a tree is constructed for each of them. There is no general rule when to stop the tree growing. The size of the trees can be controlled by control argument or prune.classbagg. By default, classification trees are as large as possible whereas regression trees and survival trees are build with the standard options of rpart.control. If nbagg=1, one single tree is computed for the whole learning sample without bootstrapping.

If coob is TRUE, the out-of-bag sample (Breiman, 1996b) is used to estimate the prediction error corresponding to class(y). Alternatively, the out-of-bag sample can be used for model combination, an out-of-bag error rate estimator is not available in this case. Double-bagging (Hothorn and Lausen, 2003) computes a LDA on the out-of-bag sample and uses the discriminant variables as additional predictors for the classification trees. comb is an optional list of lists with two elements model and predict. model is a function with arguments formula and data. predict is a function with arguments object, newdata only. If the estimation of the covariance matrix in lda fails due to a limited out-of-bag sample size, one can use slda instead. See the example section for an example of double-bagging. The methodology is not limited to a combination with LDA: bundling (Hothorn and Lausen, 2002) can be used with arbitrary classifiers.

Value

The class of the object returned depends on class(y): classbagg, regbagg and survbagg. Each is a list with elements

y	the vector of responses.
X	the data frame of predictors.
mtrees	multiple trees: a list of length nbagg containing the trees (and possibly additional objects) for each bootstrap sample.
OOB	logical whether the out-of-bag estimate should be computed.
err	if OOB=TRUE, the out-of-bag estimate of misclassification or root mean squared error or the Brier score for censored data.
comb	logical whether a combination of models was requested.

For each class methods for the generics prune, print, summary and predict are available for inspection of the results and prediction, for example:

`print.classbagg`, `summary.classbagg`, `predict.classbagg` and `prune.classbagg` for classification problems.

Author(s)

Torsten.Hothorn <Torsten.Hothorn@rzmail.uni-erlangen.de>

References

Leo Breiman (1996a), Bagging Predictors. *Machine Learning* 24(2), 123–140.

Leo Breiman (1996b), Out-Of-Bag Estimation. *Technical Report* ftp://ftp.stat.berkeley.edu/pub/users/breiman/OOBestimation.ps.Z.

Leo Breiman (1998), Arcing Classifiers. *The Annals of Statistics* 26(3), 801–824.

Peter Bühlmann and Bin Yu (2002), Analyzing Bagging. *The Annals of Statistics* 30(4), 927–961.

Torsten Hothorn, Berthold Lausen, Axel Benner and Martin Radespiel-Troeger (2002), Bagging Survival Trees. *Preprint, Friedrich-Alexander University Erlangen-Nuremberg*, http://www.mathpreprints.com/.

Torsten Hothorn and Berthold Lausen (2002), Bundling Classifiers by Bagging Trees. *Preprint, Friedrich-Alexander University Erlangen-Nuremberg*, http://www.mathpreprints.com/.

Torsten Hothorn and Berthold Lausen (2003), Double-Bagging: Combining classifiers by bootstrap aggregation. *Pattern Recognition* 36(6), 1303–1309.

Examples

```
# Classification: Breast Cancer data
data(BreastCancer)
# Test set error bagging (nbagg = 50): 3.7% (Breiman, 1998,
# Table 5)
mod <- bagging(Class ~ Cl.thickness + Cell.size
                + Cell.shape + Marg.adhesion
                + Epith.c.size + Bare.nuclei
                + Bl.cromatin + Normal.nucleoli
                + Mitoses, data=BreastCancer, coob=TRUE)
print(mod)
# Test set error bagging (nbagg=50): 7.9% (Breiman, 1996a,
# Table 2)
data(Ionosphere)
Ionosphere$V2 <- NULL # constant within groups
bagging(Class ~ ., data=Ionosphere, coob=TRUE)

# Double-Bagging: combine LDA and classification trees
# predict returns the linear discriminant values, i.e.
```

```
# linear combinations of the original predictors
comb.lda <- list(list(
              model=lda, predict=function(obj, newdata)
              predict.lda(obj, newdata)$x))

# Note: out-of-bag estimator is not available in this
# situation, use errorest
mod <- bagging(Class ~ ., data=Ionosphere, comb=comb.lda)
predict(mod, Ionosphere[1:10,])

# Regression:
data(BostonHousing)
# Test set error (nbagg=25, trees pruned): 3.41 (Breiman,
# 1996a, Table 8)
mod <- bagging(medv ~ ., data=BostonHousing, coob=TRUE)
print(mod)

learn <- as.data.frame(mlbench.friedman1(200))
# Test set error (nbagg=25, trees pruned): 2.47 (Breiman,
# 1996a, Table 8)
mod <- bagging(y ~ ., data=learn, coob=TRUE)
print(mod)

# Survival data
# Brier score for censored data estimated by
# 10 times 10-fold cross-validation: 0.2 (Hothorn et al,
# 2002)

data(DLBCL)
mod <- bagging(Surv(time,cens) ~ MGEc.1 + MGEc.2 + MGEc.3 +
                                 MGEc.4 + MGEc.5 + MGEc.6 +
                                 MGEc.7 + MGEc.8 + MGEc.9 +
                                 MGEc.10 + IPI, data=DLBCL,
                                 coob=TRUE)

print(mod)
```

Appendix B

Variable Description

The following table gives a short description of the 62 variables which are derived from a HRT image and were used as the basis for classifier construction in this work. The definition of a subset of them used for the simulation model is described in Chapter 9. For the details we refer to Heidelberg Engineering (1997).

No.	Variable	Description	No.	Variable	Description
1	ag	area global	32	vbsi	volume below surface inferior
2	at	area temporal	33	vasg	volume above surface global
3	as	area superior	34	vast	volume above surface temporal
4	an	area nasal	35	vass	volume above surface superior
5	ai	area inferior	36	vasn	volume above surface nasal
6	eag	effective area global	37	vasi	volume above surface inferior
7	eat	effective area temporal	38	vbrg	volume below reference global
8	eas	effective area superior	39	vbrt	volume below reference temporal
9	ean	effective area nasal	40	vbrs	volume below reference superior
10	eai	effective area inferior	41	vbrn	volume below reference nasal
11	abrg	area below reference global	42	vbri	volume below reference inferior
12	abrt	area below reference temporal	43	varg	volume above reference global
13	abrs	area below reference superior	44	vart	volume above reference temporal
14	abrn	area below reference nasal	45	vars	volume above reference superior
15	abri	area below reference inferior	46	varn	volume above reference nasal
16	hic	height in contour	47	vari	volume above reference inferior
17	mhcg	mean height contour global	48	mdg	mean depth global
18	mhct	mean height contour temporal	49	mdt	mean depth temporal
19	mhcs	mean height contour superior	50	mds	mean depth superior
20	mhcn	mean height contour nasal	51	mdn	mean depth nasal
21	mhci	mean height contour inferior	52	mdi	mean depth inferior
22	phcg	peak height contour	53	tmg	third moment global
23	phct	peak height contour temporal	54	tmt	third moment temporal
24	phcs	peak height contour superior	55	tms	third moment superior
25	phcn	peak height contour nasal	56	tmn	third moment nasal
26	phci	peak height contour inferior	57	tmi	third moment inferior
27	hvc	height variation contour	58	mr	mean radius
28	vbsg	volume below surface global	59	rnf	retinal nerve fiber thickness
29	vbst	volume below surface temporal	60	mdic	mean depth in contour
30	vbss	volume below surface superior	61	emd	effective mean depth
31	vbsn	volume below surface nasal	62	mv	mean variability

Bibliography

Abelson, H., Sussman, G., and Sussman., J. (1996), *Structure and interpretation of computer programs*, Cambridge MA: MIT Press, 2nd edition.

Barak, A. and La'adan, O. (1998), "The Mosix multicomputer operating system for high performance cluster computing," *Journal of Future Generation Computer Systems*, 13, 361–372.

Becker, R. A., Chambers, J. M., and Wilks, A. R. (1988), *The New S Language*, London: Chapman & Hall.

Blake, C. and Merz, C. (1998), "UCI repository of machine learning databases," URL http://www.ics.uci.edu/~mlearn/MLRepository.html.

Breiman, L. (1996a), "Bagging predictors," *Machine Learning*, 24, 123–140.

Breiman, L. (1996b), "Out-of-bag estimation," Technical report, Statistics Department, University of California Berkeley, Berkeley CA 94708, URL ftp://ftp.stat.berkeley.edu/pub/users/breiman/.

Breiman, L. (1996c), "Stacked regressions," *Machine Learning*, 24, 49–64.

Breiman, L. (1998), "Arcing classifiers," *The Annals of Statistics*, 26, 801–824.

Breiman, L. (2001a), "Random forests," *Machine Learning*, 45, 5–32.

Breiman, L. (2001b), "Statistical modeling: The two cultures," *Statistical Science*, 16, 199–231, with discussion.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984), *Classification and regression trees*, California: Wadsworth.

Bühlmann, P. and Yu, B. (2002), "Analyzing bagging," *The Annals of Statistics*, 30, 927–961.

Burk, R. O., Vihanninjoki, K., Bartke, T., Tuulonen, A., Airaksinen, P. J., Völcker, H. E., and König, J. M. (2000), "Development of the standard reference plane for the Heidelberg Retina Tomograph." *Graefes Archive of Clinical and Experimental Ophthalmology*, 238, 375–384.

Bylander, T. (2002), "Estimating generalization error on two-class datasets using out-of-bag estimates," *Machine Learning*, 48, 287–297.

Chambers, J. M. and Hastie, T. J. (1992), *Statistical Models in S*, London: Chapman & Hall.

Chan, K., Lee, T.-W., Sample, P. A., Goldbaum, M. H., Weinreb, R. N., and Sejnowski, T. J. (2002), "Comparison of machine learning and traditional classifiers in glaucoma diagnosis," *IEEE Transactions on Biomedical Engineering*, 49, 963–973.

Coleman, A. L. (1999), "Glaucoma," *The Lancet*, 354, 1803–10.

Cover, T. M. (1965), "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition," *IEEE Transactions on Electronic Computers*, 14, 326–334.

Devroye, L. and Györfi, L. (1985), *Nonparametric Density Estimation: The L_1 View*, New York: Wiley.

Efron, B. and Tibshirani, R. (1997), "Improvements on cross-validation: The .632+ bootstrap method," *Journal of the American Statistical Association*, 92, 548–560.

Fang, K.-T. and Zhang, Y.-T. (1990), *Generalized Multivariate Analysis*, Berlin, Heidelberg: Springer Verlag.

Freund, Y. and Schapire, R. (1996), "Experiments with a new boosting algorithm," in *Machine Learning: Proceedings of the Thirteenth International Conference*, ed. L. Saitta, San Francisco: Morgan Kaufmann, pp. 148–156.

Friedman, J. H. (1997), "On bias, variance, 0/1-loss, and the curse-of-dimensionality," *Data Mining and Knowledge Discovery*, 1, 55–77.

Gentleman, R. and Ihaka, R. (2000), "Lexical scope and statistical computing," *Journal of Computational and Graphical Statistics*, 9, 491–508.

Genz, A. (1992), "Numerical computation of multivariate normal probabilities," *Journal of Computational and Graphical Statistics*, 1, 141–149.

Heidelberg Engineering (1997), *Heidelberg Retina Tomograph: Bedienungsan-*

leitung Software Version 2.01., Heidelberg Engineering GmbH, Heidelberg.

Hilton, S., Katz, J., and Zeger, S. (1996), "Classifying visual field data." *Statistics in Medicine*, 15, 1349–64.

Hothorn, T., Bretz, F., and Genz, A. (2001a), "On multivariate *t* and Gauss probabilities in R," *R News*, 1, 27–29, URL http://CRAN.R-project.org/doc/Rnews/.

Hothorn, T., James, D. A., and Ripley, B. D. (2001b), "R/S interfaces to databases," in *Proceedings of the 2nd International Workshop on Distributed Statistical Computing, March 15-17, 2001, Technische Universität Wien, Vienna, Austria*, eds. K. Hornik and F. Leisch, URL http://www.ci.tuwien.ac.at/Conferences/DSC-2001/Proceedings/.

Hothorn, T. and Lausen, B. (2002a), "Bagging combined classifiers," in *Classification, Clustering and Data Analysis: Recent Advances and Applications*, eds. K. Jajuga, A. Sokołowski, and H.-H. Bock, Heidelberg: Springer, pp. 177–184.

Hothorn, T. and Lausen, B. (2002b), "Bagging tree classifiers for glaucoma diagnosis," in *Proceedings in Computational Statistics: COMPSTAT 2002*, eds. W. Härdle and B. Rönz, Heidelberg: Physica-Verlag, pp. 183–188.

Hothorn, T. and Lausen, B. (2003a), "Bagging tree classifiers for laser scanning images: Data- and simulation-based strategy," *Artificial Intelligence in Medicine*, 27, 65–79.

Hothorn, T. and Lausen, B. (2003b), "Double-bagging: Combining classifiers by bootstrap aggregation," *Pattern Recognition*, 36, 1303–1309.

Hothorn, T. and Lausen, B. (2003c), "On the exact distribution of maximally selected rank statistics," *Computational Statistics & Data Analysis*, in press.

Hothorn, T., Lausen, B., Benner, A., and Radespiel-Tröger, M. (2002), "Bagging survival trees," *Preprint, Friedrich-Alexander-University Erlangen-Nuremberg*, URL http://www.mathpreprints.com/.

Hothorn, T., Pal, I., Gefeller, O., Lausen, B., Michelson, G., and Paulus, D. (2003), "Automated classification of optic nerve head topography images for glaucoma screening," in *Studies in Classification, Data Analysis, and Knowledge Organization: Exploratory Data Analysis in Empirical Research*, eds. M. Schwaiger and O. Opitz, Heidelberg: Springer, pp. 346–356.

Iester, M., Mikelberg, F. S., and Drance, S. M. (1997), "The effect of optic disc size on diagnostic precision with the Heidelberg Retina Tomograph." *Ophthalmology*, 104, 545–548.

Ihaka, R. and Gentleman, R. (1996), "R: A language for data analysis and graphics," *Journal of Computational and Graphical Statistics*, 5, 299–314.

Kohavi, R. (1995), "A study of cross-validation and bootstrap for accuracy estimation and model selection," Technical report, Computer Sci-

ence Department, Stanford University, Stanford, CA 94305, URL http://robotics.stanford.edu/~ronnyk/accEst.ps.

Kropf, S. (2000), *Hochdimensionale multivariate Verfahren in der medizinischen Statistik*, Aachen: Shaker Verlag.

Lausen, B., Hothorn, T., Bretz, F., and Schumacher, M. (2002), "Optimally selected prognostic factors," *Preprint, Friedrich-Alexander-University Erlangen-Nuremberg*, URL http://www.mathpreprints.com/.

Lausen, B., Sauerbrei, W., and Schumacher, M. (1994), "Classification and regression trees (CART) used for the exploration of prognostic factors measured on different scales," in *Computational Statistics*, eds. P. Dirschedl and R. Ostermann, Heidelberg: Physica-Verlag, pp. 483–496.

Lausen, B. and Schumacher, M. (1992), "Maximally selected rank statistics," *Biometrics*, 48, 73–85.

Läuter, J. (1992), *Stabile multivariate Verfahren: Diskriminanzanalyse - Regressionsanalyse - Faktoranalyse*, Berlin: Akademie Verlag.

Läuter, J., Glimm, E., and Kropf, S. (1998), "Multivariate tests based on left-spherically distributed linear scores," *The Annals of Statistics*, 26, 1972–1988, correction: 1999, Vol. 27, p. 1441.

LeBlanc, M. and Tibshirani, R. (1996), "Combining estimates in regression and classification," *Journal of the American Statistical Association*, 91, 1641–1650.

Leisch, F. and Dimitriadou, E. (2001), "`mlbench` – A collection for artificial and real-world machine learning benchmarking problems," URL http://CRAN.R-project.org, R package version 0.5-4.

Li, M. N. and Rossini, A. (2001), "RPVM: Cluster statistical computing in R," *R News*, 1, 4–7, URL http://CRAN.R-project.org/doc/Rnews/.

Liaw, A. and Wiener, M. (2002), "Classification and regression by randomForest," *R News*, 2, 18–22, URL http://CRAN.R-project.org/doc/Rnews/.

Lugosi, G. and Nobel, A. (1996), "Consistency of data-driven histogram methods for density estimation and classification," *The Annals of Statistics*, 24, 687–706.

Mardin, C., Hothorn, T., Peters, A., Jünemann, A., Michelson, G., and Lausen, B. (2002), "New glaucoma classification method based on standard HRT parameters by bagging classification trees," *Preprint, Friedrich-Alexander-University Erlangen-Nuremberg*.

Mardin, C. Y., Horn, F. K., Jonas, J. B., and Budde, W. M. (1999), "Preperimetric glaucoma diagnosis by confocal scanning laser tomography of the optic disc." *British Journal of Ophthalmology*, 83, 299–304.

Marshall, R. J. (2001), "The use of classification and regression trees in clinical epidemiology," *Journal of Clinical Epidemiology*, 54, 603–609.

Merz, C. J. (1999), "Using correspondence analysis to combine classifiers," *Machine Learning*, 36, 33–58.

Meyer, D., Leisch, F., and Hornik, K. (2003), "The support vector machine under test," *Neurocomputing*, in press.

Mikelberg, F., Parfitt, C., Swindale, N., Graham, S., Drance, S., and Gosine, R. (1995), "Ability of the Heidelberg Retina Tomograph to detect early glaucomatous visual field loss," *Journal of Glaucoma*, 4, 242–247.

Miller, A. (2002), *Subset selection in regression*, New York: Chapman & Hall, 2nd edition.

Mojirsheibani, M. (1997), "A consistent combined classification rule," *Statistics & Probability Letters*, 36, 43–47.

Mojirsheibani, M. (1999), "Combining classifiers via discretization," *Journal of the American Statistical Association*, 94, 600–609.

Mojirsheibani, M. (2002), "An almost sure optimal combined classification rule," *Journal of Multivariate Analysis*, 81, 28–46.

Nobel, A. (2003), "Analysis of a complexity based pruning scheme for classification trees," *IEEE Transactions on Information Theory*, in press.

Peters, A., Hothorn, T., and Lausen, B. (2002), "ipred: Improved predictors," *R News*, 2, 33–36, URL http://CRAN.R-project.org/doc/Rnews/.

Rao, J. S. and Tibshirani, R. (1997), "The out-of-bootstrap method for model averaging and selection," Technical report, Cleveland Clinic and University of Toronto, URL http://www-stat.stanford.edu/~tibs/ftp/outofbootstrap.ps.

Ripley, B. D. (1996), *Pattern Recognition and Neural Networks*, Cambridge, UK: Cambridge University Press, URL http://www.stats.ox.ac.uk/pub/PRNN/.

Schapire, R. E., Freund, Y., Bartlett, P., and Lee, W. S. (1998), "Boosting the margin: A new explanation for the effectiveness of voting methods," *The Annals of Statistics*, 26, 1651–1686.

Scheffer, T. (1999), *Error estimation and model selection*, Dissertation, Fachbereich Informatik, Technische Universität Berlin.

Schiavo, R. A. and Hand, D. J. (2000), "Ten more years of error rate research," *International Statistical Review*, 68, 295–310.

Swift, S. and Liu, X. (2002), "Predicting glaucomatous visual field deterioration through short multivariate time series modelling," *Artificial Intelligence in Medicine*, 24, 5–24.

Swindale, N. V., Stjepanovic, G., Chin, A., and Mikelberg, F. S. (2000), "Automated analysis of normal and glaucomatous optic nerve head topography images." *Investigative Ophthalmology and Visual Science*, 41, 1730–1742.

Therneau, T. M. and Atkinson, E. J. (1997), "An introduction to recursive partitioning using the rpart routine," Technical Report 61, Section of Biostatistics, Mayo Clinic, Rochester, URL http://www.mayo.edu/hsr/techrpt/61.pdf.

Venables, W. N. and Ripley, B. D. (1999), *Modern Applied Statistics with S-Plus*, Springer, 3rd edition, URL http://www.stats.ox.ac.uk/pub/MASS3/.

Weih, L. M., Nanjan, M., McCarty, C. A., and Taylor, H. R. (2001), "Prevalence and predictors of open-angle glaucoma: Results from the visual impairment project." *Ophthalmology*, 108, 1966–1972.

Weihs, C. and Sondhauß, U. (2003), "Combining mental fit and data fit for classification rule selection," in *Studies in Classification, Data Analysis, and Knowledge Organization: Exploratory Data Analysis in Empirical Research*, eds. M. Schwaiger and O. Opitz, Heidelberg: Springer, pp. 188–203.

List of Tables

List of Figures