LEARNING DIAGNOSTIC RULES WITH MULTIVARIATE CLASSIFICATION ALGORITHMS

SPECIFIC NEEDS AND CHALLENGES

DISSERTATION

Zur Erlangung des Grades
eines Doktors der Naturwissenschaften
der Universität Dortmund

Der Fakultät Statistik
der Technischen Universität Dortmund
vorgelegt von

Raluca Ilinca Schmitt

Dortmund, June, 2009

Abstract

Considerable efforts are spent in the diagnostic research on finding biomarker panels that have a high potential to accurately identify a complex disease at an early stage.

This thesis addresses the realisability of specific requirements which a diagnostic rule should comply with in order to be accepted and useful within diagnostic workflows. Major aims in the process of rule building for diagnostic purposes are beside the high accuracy also the simplicity and interpretability of diagnostic rules. They have to provide accurate and reproducible results in order to be reliable. They have to be simple for an easy assessment in the diagnostic practice and good interpretable for a high acceptance by medical practitioners.

A simultaneous accomplishment of these quality standards is difficult due to the trade-off between accuracy and model complexity.

For instance *Logic Regression* might be a suitable method for diagnostic classification problems as it provides very simple and interpretable discriminant rules. These are defined as *and-or* combinations of binary predictors. However a performance loss is expected due to the necessity to dichotomize continuous predictors.

Advantages and disadvantages of simple and easy interpretable classification models (e.g. Logic Regression) when compared to established but more complex and powerful ones (e.g. Regularized Discriminant Analysis, Random Forests) are highlighted and discussed.

Another major challenge is to ensure the fair comparison of classification algorithms and diagnostic rules in order to select the most promising candidates. Regarding a general diagnostic task the algorithm should be selected that leads to the most stable and unbiased results. Regarding some special diagnostic question the most accurate discriminant rule should be selected. Adequate designs to evaluate and optimize classification algorithms and rules are presented.

This thesis deals also with the problem of an accurate estimation of rules and of their performance in the context of a heterogeneous target population but non-representative training data. Learning the diagnostic rule on some excerpt of the target population with different observed subclass prevalences than the true ones might be a source of severe bias regarding both the selected rule and its estimated accuracy.

Four weighting classification algorithms that account for the subclass prevalence structure of the target population during the processes of rule building and rule validation are presented. Their feasibility over various practical settings is assessed both empirically and theoretically.

All investigated methods are applied on some real data sets of rheumatoid arthritis cases and controls provided by Roche Diagnostics GmbH, Penzberg. Supplementary information is gained with simulated data.

Acknowledgement

I would like to thank all people supporting me during my dissertation work.

First of all many special thanks to my supervisor Ursula Garczarek and to Andrea Geistanger, for keeping me motivated, for precious mentoring and fruitful discussions. I appreciate not only their professional, but also their human support and friendship.

Thanks to my professor Claus Weihs for useful discussions, cooperativeness and interest with respect to the topic of my dissertation.

Thanks to Veit Peter Grunert, Friedemann Krause and Christoph Berding for giving me the challenging opportunity to collaborate with Roche Diagnostics GmbH for almost five years. To work on interesting projects which have inspired also the subject of my dissertation.

Thanks to my colleagues Geraldine Rauch and Insa Winzenborg for some useful proof reading to the text. A special thank to my friend Geraldine for being always prepared to lend a helping hand.

Thanks to the Roche Ausländer Community for cheering up my last year of self-chosen Penzberg exile. *Muchas gracias por todo* to Micaela Molina Navarro, *vielen Dank* to Daniela Behling, *bolshoe spasibo* to Kirill Bessoonov, and *thank you very much* to Hillary Workman for reviewing my English in this work. Thanks for your friendship and patience!

Finally, I want to thank to the really special persons in my life, which have all contributed more or less to my successful way through the last years. Vielen Dank an Niels Schmitt für seine Liebe und Hilfsbereitschaft. Un grand merci a Yasser Gaou pour son amour précieux et le support dans les bons et mauvais moments.
Last, but not least, thanks to my greatest fans, my parents Elena and Radu-Graţian Pepene, and above all to God. I owe them everything. I dedicate this work to them as well as to the memory of my grandmother, Dr. Gabriela Balea-Pepene. Mulţumesc!

Contents

List of figures

List of tables

Chapter 1

Overview

One of the major interests of the medical research in the field of early diagnosis is the discovery of protein biomarkers and their assessment for clinical needs with novel chip technologies. While medicine still relies on the knowledge and intuition of individual doctors, gradual advances in the biomarker research over the past decades have contributed to a broadening of the diagnostic methodology. Using new genomic and proteomic tools scientists are engaged in an unprecedented large-scale hunt for new biomarkers.
Several biomarkers are available nowadays but none of them is sensitive and specific enough for the early diagnosis of a disease with complex biological profile. So, it is more likely that a combination could increase the diagnostic accuracy in many settings.

The biomarker identification is supported in the statistical field through development, improvement and assessment of multivariate classification algorithms. These are designed to filter out from an initial biomarker pool a powerful combination able to spot the target condition at an early stage. They should provide meaningful rules based on the selected biomarker panels for a reliable identification of the disease.

The quality of the rules for the diagnostic practice is judged not only in terms of their accuracy but also of their simplicity and interpretability. They should be highly accurate to guarantee a low probability of erroneous assignments, thus a reliable diagnosis. They should be simple to enable a fast and easy assessment of the disease status and interpretable to get a high acceptance by the medical professionals.
However, there is usually a trade-off between performance and complexity of the classification models. Therefore, some methods might provide more simple and interpretable rules than other ones in change of some performance loss. An interesting question is here, how much accuracy loss is tolerable for the sake of better handling and understandability of the diagnostic rules.

Chapter 3 highlights dangers and benefits associated to the accomplishment of these quality standards of diagnostic rules. They are exemplified on Logic Regression (LogicR) (Ruczinski *et al.*, 2003). This method seems to be suitable for diagnostic tasks since it provides very simple and interpretable models, logic rules being *and-or* combinations of binary predictors.
However, most biomarkers are measured on a metric scale, thus they have to be dichotomized first in order to be used with Logic Regression. This data transformation from a quantitative to qualitative level affects the accuracy of the logic rules by loss of valuable information. The method is known to suffer not only from its restriction to binary data, but also from its particularly long run times.

In this thesis LogicR is applied to multivariate classification problems with continuous predictors. Keeping track on all the pro- and contra-arguments for LogicR, its performance is compared with that of *Regularized Discriminant Analysis* (RDA) (Friedman, 1989), as a well established procedure. Since LogicR belongs to the family of tree algorithms, its performance is compared additionally to that of *Random Forests* (RF)(Breiman, 1997). The methods are applied on a real data set of rheumatoid arthritis cases and controls that comprises measurements of four biomarkers. Supplementary information is gained with results from three simulation designs.

The diagnostic potential of LogicR is shown to strongly depend on the underlying data structure. In the best case LogicR is as good as RF and RDA, in the worst case its misclassification error rate is 14% while RF and RDA achieve about 9%. However, the real burden of LogicR is the computational time, which impedes from the use of LogicR with settings that would have the potential to enhance its performance and stability.

Another critical issue in the development of new diagnostic rules based on biomarker combinations, which this thesis deals with, is the fair choice of the best performing algorithms and/or rules. Examples of adequate designs to optimize and evaluate classification algorithms and classifiers (i.e. rules) are provided and discussed in Chapter 2. They find application several times throughout this thesis.

From the field of machine learning comes the concept of *benchmark study*, which is a parallel study of some competing algorithms or classifiers. It aims at a comparison of them with respect to a certain performance measure. The taxonomy of statistical questions in machine learning illustrated by Dietterich (1998) distinguishes between two types of targets: finding the best classifier and finding the best algorithm. The first target is related rather to a specific application,

the second to a more general classification task.
An algorithm is better than its competitors if it provides *superior* classifiers over a relevant fraction of experimental tasks. Here *superior* refers to the highest accuracy and a competing consistency of the classification model, thus a similar performance on the training and test data sets.

Given a particular classification task, a fair benchmark of algorithms should take into account different sources of variability which affect the final rule choice. This is possible by embedding all competing algorithms within a so-called Monte Carlo Cross-Validation design (Plutowsky, 1995). By the Monte Carlo procedure one takes into account the variability associated to a particular choice of the training and test data sets.The Cross-Validation (CV) procedure enables a realistic estimation of the optimal rule parameters and features.

A generalization of results is possible by simulation studies. They provide a basis for the comparison of algorithms over a large variety of data situations. Several data conditions are generated and the benchmark is effectuated for every generated condition. In the end a balance is drawn over all particular benchmark results.

For instance, in this thesis a benchmark of the algorithms LogicR, RDA and RF is performed. Initially, a data set of rheumatoid arthritis cases and controls is used. The Monte Carlo Cross Validation design for a realistic assessment of the performance of each algorithm is applied. However, the results of this benchmark study are valid only in the context of this particular data set. To assess the suitability of LogicR within diagnostic workflows in general, supplementary information is needed. This is gained from three simulations designs.

If, additionally, a set of factors is suspected to affect the performance of the algorithms and therefore also the outcome of the algorithm benchmark, a suitable design of experiments is needed to extract the relevant factors. For each considered combination of factor levels according to this design a simulation study is carried out and the benchmark of algorithms is performed. Once the relevant factors are extracted and some useful model information is retained, the design might be augmented to consolidate this information and guarantee a precise estimation of the predicted performance.

Another critical issue in the process of developing diagnostic rules is the representativeness of the data used for learning. This should mirror the true prevalence structure of the target population which the diagnostic rules are actually designed for.
A considerable part of this thesis concerns the problem of a realistic estimation of classification

errors in the context of non-representativeness of the data. In Chapter 4 this issue is studied empirically. A theoretical investigation on this topic is shown in Chapter 5.
The target population is assumed to be heterogeneous with known subclass prevalences.
This situation is especially encountered in the screening practice: asymptomatic patients are recruited in a prospective frame until some reasonable number of patients carrying the condition of interest is accrued. The control collective of an ideal screening population consists of several other diseases for which usually the true prevalences are known.

When diagnostic rules are used as screening tools, it is critical that their accuracy is measured on the relevant clinical population. However, especially if the condition of interest has a low prevalence in the clinical population, such a prospective study can require an enormous sample of patients. Besides, the development of a new diagnostic rule demands the verification of the true disease status (gold standard procedure) for all study patients. The resulting costs of such studies might be prohibitive. Therefore, it is more practical to use only some small excerpt of the screening population, which we call the data *at hand*. However, this is usually non-representative, which means that the true (target population) and observed subclass prevalences (data at hand) are very different. In this case, learning the diagnostic rule without taking into account the true composition of the target population may cause not only a severe bias in the performance estimates, but also the selection of an erroneous diagnostic rule.

In Chapter 4, four weighting algorithms are proposed which should help to overcome the disadvantages related to a suboptimal data at hand. They apply at least one of two modalities to account for the true subclass prevalence structure:

(a) in the phase of rule validation by computation of weighted error estimates as sums of the subclass errors with weights given by the true subclass prevalences;

(b) in the phase of rule building by computation of weighted class distribution parameters: weighted class means and covariance matrices according to the true subclass prevalences.

First, the suitability of these weighting algorithms in the context of a heterogeneous control collectives is surveyed over a large variety of configurations of the target population. The theory of experimental designs is used to generate interesting target situations for the simulation study. For each given configuration of the target population a Monte Carlo Cross Validation design is applied to fairly establish the most adequate weighting procedure.
The algorithms which account for the true subclass prevalences both in the phase of rule building and validation are proved to be superior to algorithms which weight only in the validation step. Weighted rules approach well the expected classification errors in the target population especially when the data at hand is highly suboptimal. In the context of a pronounced non-

representativeness of the data set at hand weighting seems to be the right alternative.

However, the question arises, if the benefits associated to a bias reduction by weighting are not endangered by some simultaneous variance inflation. The variance of weighted estimates might be enhanced by contribution of the up-weighted subclasses, thus, of those subclasses being under-represented in the data set at hand.

In Chapter 5 theoretical investigations are carried out in order to establish suitable and potentially dangerous situations for the application of weighted estimates. These may especially help users with less knowledge about the target population to make decisions about using weighted or unweighted rules before the rule building process starts. In this theoretical survey the heterogeneous class is composed of two subclasses.

Evidence is provided for the superiority of weighted estimates especially in the context of medium to high degrees of mismatch between the data at hand and the target population. Extremely high or extremely low degrees of mismatch have an increased danger potential. They might result in counter-productive weighted rules.

Also, the size of the heterogeneous class and the difference between the true subclass error probabilities influence considerably the benefit expected from weighted classification errors. The larger they are, the higher the chances for an efficient weighting.

The size of the heterogeneous class and the distance between the true subclass means are of great importance regarding the benefit expected by weighting class means, too. The larger they are, the higher also the chances for more efficient weighted than unweighted class mean estimates. Here, it should be noted also that a higher distance between the true subclass means indicates a pronounced subclass structure in the target population.

Chapter 2

Adequate designs to optimize and evaluate classification algorithms and rules

We highlighted in Chapter 1 that multivariate classification algorithms find a strong application in the field of diagnostic classifications by means of biomarker panels. They should be able to provide meaningful (interpretable and easily assessable) but also highly accurate diagnostic rules (diagnostic tests).

Important issues in the process of developing new classification algorithms and rules for diagnostic purposes are the reliable assessment and comparison of their performance. In statistical learning an empirical experiment with the aim of comparing algorithms or rules is called a *benchmark study* (Hothorn *et al.*, 2005). The taxonomy of inference problems in the special case of supervised learning algorithms developed by Dietterich (1998) is helpful to distinguish between finding best classifiers and finding best algorithms. The first target is related rather to a specific classification problem, the second to a more general classification frame.

For instance, given a particular diagnostic task the primary goal is usually to find the best diagnostic rule and estimate its accuracy for future samples.
However, new diagnostic rules that are more accurate but also cheaper, simpler and more interpretable are sought for diagnosis of many conditions. For this reason, the development of new classification algorithms and the general assessment of their suitability for diagnostic questions relatively to old ones is an active research area.
Also, for instance, an algorithm included in a medical device should analyze and update periodically the diagnostic test based on accumulated examples of diseased and non-diseased subjects. In this case it is again critical to implement the most promising algorithm from a pool of candidates.
Depending on the intended comparison, of algorithms or rules, an adequate design should be used to assess a feasible measure of performance. This should take into account, as far as pos-

sible, sources of variation associated to the way in which the learning was performed.

The first section introduces some formal preliminaries to the problem of comparing algorithms and/or rules.

The second section illustrates adequate designs for the comparison of algorithms in the diagnostic context. They find several times application throughout this thesis: in Chapter 3, when we assess the suitability of Logic Regression for diagnostic classifications with respect to established classification algorithms like RDA and Random Forests; in Chapter 4, when we compare four new weighting algorithms in the context of heterogeneous data on real and simulated data sets. Also, these designs can be easily adapted to a comparison of diagnostic rules on a specific diagnostic task.

The last section addresses the benchmark of algorithms from another perspective. Interesting is here a comparison of algorithms over a range of different data sets. For instance, each data set corresponds to some special distributional situation, like in our simulation study in Chapter 4. For the sake of clearness such a benchmark study with more than one data set was called *benchmark survey* by Eugster *et al.* (2008).
Usually none of the algorithms is superior to the others over all data situations. Its final performance depends on the distribution of the data generating process (Hothorn *et al.*, 2005). Therefore, a challenging aspect is to identify appropriate factors for describing the target distributional context in which an algorithm outperforms the others.
A suitable way to proceed in this regard using the theory of experimental designs is presented.

2.1 Formal preliminaries

In supervised learning problems a learning data set $\mathcal{L}$ contains the empirical information about the data generating process DGP (Hothorn *et al.*, 2005). This data reflects our knowledge about the world. It comprises N independent and identically distributed samples drawn from the DGP, $\mathcal{L} = \{z_1, z_2, \ldots, z_N\}$. The learning samples have the form of $(p+1)$-tuples $z_i = (y_i, \boldsymbol{x}_i')'$, where y_i represents the true class (assessed by the best available reference method, the *gold standard*) and $\boldsymbol{x}_i$ is the vector of p observed features (characteristics) of sample i, $i = 1, 2, \ldots, N$.
The aim of the learning task is to construct rules (classifiers) which, based on a meaningful set of object characteristics, map future objects to classes. The classification algorithm, also called inducer, is the learning tool which builds a rule (classifier, test) from the available data.
For instance, an example of diagnostic task frequently addressed in this thesis is the identification of rheumatoid arthritis using biomarker concentrations in serum. Elevated concentration levels of some candidate biomarkers could be good indicators for the disease presence. Thus, a diagnostic rule for predicting the true disease status is induced by means of some suitable

algorithm from the biomarker measurements on the available study collective.

Throughout this chapter the availability of several candidate algorithms as potential problem solvers is assumed. These competitors are denoted as $a_k,\ k = 1, 2, \ldots, K$. Each of them represents a two-step procedure: in a first step, a rule is fitted on the learning sample $\mathcal{L}$, yielding a function $a_k(\cdot, \mathcal{L})$; in a second step, this is used to make class predictions for new objects of interest.

When searching for the best solution, candidates need to be compared with respect to some problem specific performance measure $p(a_k, \mathcal{L})$. This depends on the algorithm used and on the DGP. The right strategy, independently on the available amount of data, is to assess this measure of performance on some test (validation) data set drawn independently from the same DGP as the training data, which is used for learning the rule. Since $\mathcal{L}$ is a random learning sample also $p(a_k, \mathcal{L})$ is a random variable whose variability is induced by the variability of the algorithms and of the training and test samples which follow the same DGP as $\mathcal{L}$.

This measure of performance is defined by some functional μ of the distribution of a loss function $\mathbf{L}$, which is usually the expected, quadratic, median or absolute loss, i.e. $p(a_k, \mathcal{L}) = \mu[\mathbf{L}(a_k(\boldsymbol{x}, \mathcal{L}), y)]$.
Throughout this thesis the expectation as functional and the absolute loss are used, leading to the misclassification rate as empirical measure of performance. It is obvious that the absolute and quadratic loss functions are essentially the same in the context of two class classification problems with class labels 1 (disease) and 2 (controls) (or 1 and 0 respectively) like those addressed in this work.

In a similar way to Hothorn *et al.* (2005) we account for the variability associated to the performance of each algorithm and for the variability of the rules produced by each algorithm using the so-called Monte Carlo Cross-Validation design (Plutowsky, 1995) with inner cross-validation loops for rule optimization. This design and its advantages are detailed in the next section.

2.2 Designs for the comparison of algorithms

A good benchmark of algorithms and rules should take into account essentially four sources of variation, which are listed by Dietterich (1998):

1. the selection of the test set: a classifier outperforms another on a randomly drawn test data set, although they are asymptotically (i.e. on the whole population) equivalent;

2. the selection of the training set: algorithms show usually an ”instability” behavior (Breiman (1996)) as small changes in the training set can result in large modifications of the fitted model;

3. internal randomness of the learning algorithm: the final model depends on a random initialization of the outgoing state of the algorithm;

4. random classification error: if $\eta\%$ supervising errors are available in the test set, no algorithm will achieve a better error rate than $\eta\%$ (gold standard problem); the presence of supervising errors in practice is almost impossible to overcome, therefore a perfect classification is usually a hardly achievable task.

A usual shortcoming of approaches which compare raw point estimates of the performance measure on a validation data set in finite sample situations, is that they ignore the various sources of variability, leading to uncertain conclusions. Common binomial confidence intervals would just control the variability due to the choice of the test data. Common statistical tests, like McNemar’s test (see Dietterich (1998)) should be applied only if the variability due to the choice of the training set or the internal randomness of the algorithm are believed to be negligible.

In the statistical literature various methods were proposed to control the different sources of variability in the performance estimates.
Dietterich (1998) surveys some ways to account for the first two sources of variation while performing statistical tests for the comparison of algorithms.
Bauer & Kohavi (1999) sample repeatedly from the training set, in order to improve the estimates for the bias and the variance of the error rates on the test set. Consequently, they adjust the test sample estimates for the intrinsic noise and compare the candidate algorithms in terms of average relative error reduction.
Another way to address these problems is suggested by Hothorn *et al.* (2005). They introduce a theoretical framework for the comparison of learning algorithms and propose some methods of sampling from algorithm conditioned distributions of the performance measures, in an independent way, in order to enable standard statistical tests.

In a similar way to Hothorn *et al.* (2005) we account for the first three sources of variability associated to the performance of each competing algorithm by various outer training-test splits of the whole learning data. In this way different random training and test samples are obtained that should theoretically follow the same DGP as the original data.
The difference to Hothorn *et al.* (2005) is that we draw the training and test data sets from the original data using a class stratified random sampling without replacement. This procedure is called Monte Carlo (MC) cross-validation (Plutowsky, 1995).
The rule variability associated to its parameter estimates and features, corresponding to the second source of variation, is controlled by a simple inner cross-validation (CV) design on each of

the Monte Carlo training data sets.

In this thesis the short name $L \times M$-MCCV refers to an iterative estimation-optimization design based on L outer MC and M inner cross-validation (CV) loops. It is also referred in some applications as the double-loop cross-validation, since it combines two different cross-validation strategies (Plutowsky, 1995): (1) L-fold outer *Monte Carlo cross-validation* and (2) M-fold inner *Disjoint Set Cross-Validation* (the regular cross-validation method).

The workflow according to an $L \times M$-MCCV design is illustrated in Figure 5.1.

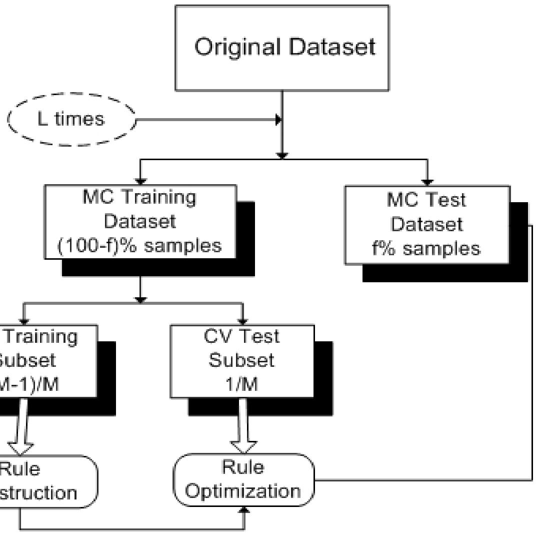

Original Dataset
L times
MC Training Dataset (100-f)% samples
MC Test Dataset f% samples
CV Training Subset (M-1)/M
CV Test Subset 1/M
Rule Construction
Rule Optimization

Figure 2.1: Workflow of an $L \times M$-MCCV design

Strategy (1) consists of L independent iterations of the following steps:

a1. The original learning data $\mathcal{L}$ is subdivided by random sampling without replacement into a so-called MC training and an MC test data set according to a given proportion.

a2. On the MC training data set the *best* rule is fitted. This is defined in terms of minimal CV error rate and its optimization succeeds by means of strategy (2) starting from the current MC training data.

a3. The performance of the rule developed at step (a2) is evaluated by computation of its misclassification rate on the MC test data.

The final misclassification error estimate is an average over the L misclassification rates on the MC test subfractions computed in (a3).

Strategy (2), representing the most common form of cross-validation, consists of the following steps given some starting training data (e.g. an MC training data set):

b1. The data is split into M disjoint and approximately equally-sized subfractions, called also CV test data sets. The split is based on random sampling without replacement.

b2. On each set of $M-1$ subfractions a rule is constructed for a fixed combination of model parameters and features.

b3. The misclassification rate of the rule constructed in (b2) is computed on the remained subfraction.

b4. The CV estimate of the error rate is obtained as average over the misclassification rates obtained on the CV test data sets.

Let the M test subfractions of some M-fold cross-validation strategy be generally denoted as $\mathcal{L}^{(1)}, \mathcal{L}^{(2)}, \ldots, \mathcal{L}^{(M)}$. The corresponding counterparts, i.e. $\mathcal{L} \setminus \mathcal{L}^{(m)}$, $m = 1, \ldots, M$, are used to learn the rule. The CV estimates of the probability to misclassify class i as j by some rule δ constructed on $\mathcal{L}$ are computed as:

$$\begin{aligned}
\hat{\epsilon}_i^{(M-CV)} &= \frac{1}{M} \sum_{m=1}^{M} \hat{\epsilon}_i^{(m)} \\
&= \frac{1}{M} \sum_{m=1}^{M} \frac{1}{N_i^{(m)}} \left[\sum_{(i,\boldsymbol{x}')' \in \mathcal{L}^{(m)}} I_{\{\delta_{-\mathcal{L}^{(m)}}(\boldsymbol{x}) = j\}} \right] \\
&= \frac{1}{N_i} \sum_{m=1}^{M} \left[\sum_{(i,\boldsymbol{x}')' \in \mathcal{L}^{(m)}} I_{\{\delta_{-\mathcal{L}^{(m)}}(\boldsymbol{x}) = j\}} \right], \ (i \neq j,\ i,\ j \in \{1,2\}), \qquad (2.2.1)
\end{aligned}$$

and the overall CV error estimate is:

$$\hat{\epsilon}^{(CV)}(\delta) = \hat{\pi}_1 \hat{\epsilon}_1^{(M-CV)} + \hat{\pi}_2 \hat{\epsilon}_2^{(M-CV)}. \qquad (2.2.2)$$

Here N_i are the class sizes and $N_i^{(m)}$, $i = 1, 2$, are the class sizes in the CV test subfraction $\mathcal{L}^{(m)}$. In the class stratified CV which we practice throughout this work, $N_i^{(m)}$ represent an $\frac{1}{M}$-fraction of the total class size N_i. Further, $\hat{\epsilon}_i^{(m)}$ is the common error estimate within class i on the CV test subfraction $\mathcal{L}^{(m)}$, $\delta_{-\mathcal{L}^{(m)}}$ represents the rule fitted after $\mathcal{L}^{(m)}$ is left out, and $\hat{\pi}_i$ are the observed class prevalences in the whole training data, i.e. $\hat{\pi}_i = \frac{N_i}{N}$, $i = 1, 2$. I_η stands for the indicator function with:

$$I_\eta = \begin{cases} 1 & \text{if } \eta \text{ is true,} \\ 0 & \text{otherwise.} \end{cases}$$

Now, every MC or CV test data set is an example of validation data set. Given some validation data set $\mathcal{T}$ with class sizes represented by $N_i^{(ts)}$, $i = 1, 2$, and δ some rule developed on the learning data $\mathcal{L}$, the test-sample (TS) estimates of the probability to misclassify class i as j by δ are computed as:

$$\hat{\epsilon}_i^{(ts)} = \frac{1}{N_i^{(ts)}} \sum_{(i,x')' \in \mathcal{T}} I_{\{\delta(x)=j\}}, \ (i \neq j, \ i, j \in \{1, 2\}), \tag{2.2.3}$$

and the overall TS-error estimate is:

$$\hat{\epsilon}^{(ts)}(\delta) = \hat{\pi}_1 \hat{\epsilon}_1^{(ts)} + \hat{\pi}_2 \hat{\epsilon}_2^{(ts)}. \tag{2.2.4}$$

For instance, in the inner M-fold CV let the TS-error estimate on the m-th CV test data set be denoted as $\hat{\epsilon}_m^{(ts,CV)}$, $m = 1, 2, \dots, M$. In the L-fold outer MC cross-validation let the TS-error estimate on the l-th MC test data set be denoted as $\hat{\epsilon}_l^{(ts,MC)}$, $l = 1, 2, \dots, L$. Then the final CV error estimate corresponds to:

$$\hat{\epsilon}^{(CV)}(\delta) = \frac{1}{M} \sum_{m=1}^{M} \hat{\epsilon}_m^{(ts,CV)},$$

and the final error estimate obtained by MC cross-validation to:

$$\hat{\epsilon}^{(MC)}(\delta) = \frac{1}{L} \sum_{l=1}^{L} \hat{\epsilon}_l^{(ts,MC)}.$$

Throughout this thesis all splits into MC or CV training and test data sets are performed using the class as stratification variable. This enables to keep track on the quotient between the two classes in the original data. Stratification was proved to be the better scheme, both in terms of bias and variance of the error estimates, when compared to regular cross-validation (Kohavi, 1995). A stratified 10-fold CV was for instance recommended for model selection (this would correspond to a 10-fold inner CV in our case).

Note that the Monte Carlo cross-validation, thus strategy (1), was proven to be asymptotically consistent regarding the model selection. This means that it finds the best prediction model with probability one (Xu *et al.*, 2004) given infinite sample sizes.

Another advantage of the Monte Carlo cross-validation is that it offers also a feasible approximation of the exhaustive cross-validation using much less splits. Given a size n_V of the test data sets, the latter method considers all different training-test splits of the original data with test sets of this size. Though it performs the most complete assessment of the first three sources of variation and is asymptotically consistent, the exhaustive cross-validation is hardly practicable since its computational complexity grows exponentially with respect to n_V. By Monte Carlo

cross-validation the computational complexity of the exhaustive CV is reduced substantially. Theoretically, the more samples are used for learning the less MC loops are needed.
For these reasons, an even better design than ours would be one based both on outer and inner Monte Carlo cross-validation loops. However, this alternative was not considered since it is computationally expensive and time-consuming. The combination of outer MC and inner 10-fold cross-validation loops that we always apply given enough samples is more feasible from a time and complexity point of view. Over simulations it also provides mostly satisfying results in terms of model consistency and final error estimation.

2.3 Simulation designs

Usually none of the algorithms in a benchmark outperforms the others over all data situations. Its performance depends on the strategy used and on the DGP. Therefore, sometimes it is interesting to perform the benchmark of algorithms over multiple domains (data sets from different DGPs) in order to identify the reunion of conditions that favors the application of an algorithm or another. For this, simulations over different data configurations are needed. Especially in the context of time-consuming algorithms, an adequate simulation design helps to gain information from a feasible number of runs.

Ideally, the data simulation for the comparison of algorithms over multiple domains is a sequential process. This enables to take advantage on the information gained from a previous small set of runs in a next well-grounded simulation study. Some possible steps of such a sequential design are:

Step 1 : Identification of a set of factors that describe the multiple domains and potentially impact on the algorithms performance;

Step 2 : Generation of a screening design to separate a subset of factors with significant influence on the performance of all algorithms from the rest;

Step 3 : Augmentation of the original design to enable a reliable prediction of the algorithms performance by means of a quadratic model in the selected factors.

The **first step** towards the generation of an adequate design is a conceptual one: various and relevant data distributions for the regarded diagnostic question are meaningfully described. A set of factors, which enable the specification of the distributional configuration of the data and are assumed to impact on the diagnostic accuracy of the algorithms, is assessed.

For instance, in Chapter 4 the problem of classification when the target population has a subclass structure with known true subclass prevalences is addressed. Four weighting algorithms that account for the true subclass prevalence structure in different phases of the rule development process are compared in order to assess the best way to proceed in this diagnostic situation. However, the position of class and subclasses to each other as well as the degree of mismatch between the study and the target population with respect to the subclass prevalence structure are expected to impact on the outcome of this benchmark. Therefore, in the context of one homogenous and one heterogeneous class the following factors are considered to describe the distributional configuration of the target population: the euclidian distances from the subclass centers to the center of the homogeneous class, the angle between the subclass centers with vertex in the center of the homogenous class and the absolute difference of the true subclass prevalences.

Assume that an initial pool of design factors is available. A **second step** in the development of an adequate simulation design is to rule out eventual noise, thus to perform some reduction of the factor space. This can be done by simulating the data within the frame of a screening design. Each run corresponds to a selected combination of the factor levels for the experimental purpose. Screening designs are useful as a prelude of further experiments. They are also a cheap and efficient way to begin an improvement process since they reduce the number of runs by restricting continuous and multi-levels factors to two (or three) levels and use only a fraction of a full factorial design.

In screening for informative factors a linear model including only the main factor effects is assumed to underlie the data before the design is produced. The columns of the final design matrix, each corresponding to the generated levels of one factor, are empirically uncorrelated (orthogonal design). Besides, each column has the mean 0, thus the design contains an equal number of runs on the high and on the low levels of each factor.
Some center points, i.e. runs in the mid-range of each factor, might be used to check if the assumed model is adequate. They also provide with information about the variance of the prediction error in the center of the factor region.

After the design was generated, the simulation data sets are drawn from each DGP described by the given factors and the algorithms are run on each data set. In the end, performance estimates of each algorithm for each run are provided as responses in the model equation. For decoding of continuous encoded factor models, data analysis, factor selection tools as well as estimation procedure of the main factor effects in the screening case the interested reader is referred to Weihs & Jessenberger (1999).

Assume that the relevant factors are now selected. In a **third step** the existing design can be augmented adding new runs and starting from a model that involves besides main effects also interaction and quadratic effects. Thus, this design is more suitable for prediction. Treating the

simulation from multiple data domains as an iterative process, one can master the temptation to assume that one successful screening experiment has optimized his process. One can also avoid disappointment if a screening experiment leaves behind some ambiguities.
Also, design augmentation is powerful for sequential analysis because the objectives of a response surface methodology can be achieved changing one linear model into a full quadratic one and adding the necessary number of runs. Besides, the information gained from previous runs can be valued and empowered by new ones.

The performance estimates of the algorithms are then computed for every new run. In the end, the results obtained with the whole design are submitted to analysis targeting the selection of a global (i.e. suitable for all algorithms), meaningful (i.e. interpretable) and competitive (i.e. as accurate as possible) model for prediction. This can be used to describe the DGPs under which one algorithm proves superiority upon the others.

The generation of adequate simulation designs can be carried out in a comfortable manner using the Custom Designer provided with the JMP Software (SAS-Institute, 2005), which is a general purpose design environment. As such, it provides also screening designs. The Custom Designer presents some advantages upon classical ones:

- It offers an easily manageable interface to any type of classical design.
- It has more flexibility accommodating any number of factors of any type.
- It requires less experience and expertise than previous tools supporting the statistical design of experiments.

These qualities make the Custom Designer to the recommended way to create a design which is tailored for one's specific situation.
The Custom Designer generates designs using a mathematical optimality criterion. This can be of two types:

- in a so-called **D-optimal** design the optimality criterion focuses on precise estimates of the coefficients;
- in a so-called **I-optimal** design the optimality criterion focuses on the minimization of the average prediction variance inside the region of the factors.

D-optimal designs are the most efficient for designing experiments where the primary goal is the inference, like in case of screening for relevant factors. They are also the default design type in the Custom Designer of the JMP Software. A D-optimal design suits in particular the screening step in the process of generating a sequential simulation design.

I-optimal designs are useful for prediction, therefore they suit in particular the augmentation step in the process of generating a sequential simulation design. They tend to place fewer runs at the extremes of the factor space than do D-optimal designs. Consequently, D-optimal designs often predict better at the extreme values of the factors.

The augmentation step with changeover from a D-optimal to an I-optimal design and involvement of quadratic model terms can be comfortably performed using the Augment Designer provided with the JMP Software (SAS-Institute, 2005). In this phase for instance, if the response variance at the center points during screening was large, one might supply some replicates per design point to get a better estimation of the algorithm performance. In this way, it can be accounted for some possible instability behavior of the algorithms.

Chapter 3

Interpretability of diagnostic rules

The early identification of diseases with a complex profile by means of minimally invasive, cost-effective and highly accurate diagnostic tests is one of the big challenges of modern medicine. A promising diagnostic approach is the combination of easily accessible biomarkers to achieve the desired accuracy where individual biomarkers failed. However, the final diagnostic rule based on a biomarker combination has not only to powerfully discriminate between the diseased and non-diseased status but also to be simple and interpretable. The rule has to be simple in order to be easy to assess and interpretable in order to get high acceptance by medical practitioners.

Effective statistical algorithms are needed to provide cheap and meaningful classification rules in the context of diagnostic problems with many markers. They should comply both with the request of high discriminatory power and that of simplicity, interpretability and efficient implementation.
However, such ideal rules are not easy to design. There is usually a trade-off between performance and complexity of the classification models. Therefore, some rules gain their parsimony and understandability at the price of some performance loss.

This chapter is concerned with the suitability of simple and interpretable classification rules in the diagnostic context. The marginal question is how much performance loss is actually tolerable for sake of easier manageable and understandable rules.
These aspects are ascertained under consideration of Logic Regression (LogicR), a new tree-based method designed both for regression and classification, introduced by Ruczinski *et al.* (2003). This method is particularly interesting for the diagnostic research. It provides very simple and interpretable discriminant rules, defined as *and-or* combinations of binary predictors.
As many biomarkers measure concentration levels they are usually continuous. Consequently, they should be first dichotomized in order to be used with LogicR. On one hand, the information loss induced by this transformation from a high to a low measurement level will cause accuracy loss. On the other hand, dichotomizing for instance by means of quantiles from the control col-

lective, the resulting diagnostic rules are close to well-established diagnostic workflows. This aspect would enhance their acceptance on the diagnostic market.

Two dichotomization approaches presented by Schmitt (2005) are considered for the successful application of LogicR: the first one is based on a set of empirical quantiles (qLogicR) and the second on an optimized (best) threshold for each feature (btLogicR).
The suitability of LogicR for diagnostic classifications is assessed through a comparison with two other classification algorithms known to provide reliable results over many applications. These are Regularized Discriminant Analysis (RDA) (Friedman, 1989) and Random Forests (RF) (Breiman, 2001). RDA is considered as an established classification procedure. RF is considered as a powerful representant of the family of tree-based methods, which includes LogicR, too.

Other reasons for selecting these methods are that:

- RDA proves good results in diagnostic settings over a variety of marker distributions, though it relies on the assumption of multivariate class Gaussians; see Hand (1992) and Vaid *et al.* (2001).
- RDA includes the well known quadratic (QDA) (McLachlan, 1992) and linear (LDA) (Fisher, 1936) discriminant analysis methods as special cases.
- RF is supposed to improve the performance of Decision Trees (Breiman *et al.*, 1984) at the loss of their simplicity and interpretability. Therefore it suggests itself for the evaluation of the simplicity/accuracy trade-off.

Section 3.1 comprises a short description of the methods with focus on LogicR. Section 3.2 introduces the real and simulated data sets. Section 3.3 illustrates the application of these techniques on the available data. In Section 3.4 the final conclusions are drawn, highlighting both benefits and drawbacks of LogicR in comparison with RDA and RF.

3.1 Methods

This section introduces briefly the theoretical background of the three benchmark algorithms, LogicR, RDA and RF, in the context of two-class classification problems.
Data is given by N tuples of the form $(y, \boldsymbol{x}')'$, with $y \in \{0, 1\}$ a binary encoded dichotomous class variable and $\boldsymbol{x} = (x_1, \ldots, x_p)'$ the p-dimensional vector of observed object characteristics (features). In diagnostic problems, 1 denotes typically the disease presence and 0 its absence.
Based on this data, a rule is searched to reliably predict the class membership of future objects. This rule describes the functional connection between the class and some relevant features. The

rule should optimize a preset objective function which compares fitted values with the response and is defined in concordance with the classification scope. The objective function throughout this chapter is the cross-validated error estimate.

Cross-validation (CV) (see Chapter 2) allows for a reliable estimation of the misclassification rate and therefore, for a more adequate choice of parameters for the optimal rule. This is done by randomly splitting the available data into disjoint test subfractions, using all but one for building the rule and the remained one for evaluating the performance of the rule. Then the average over the misclassification rates computed on the test subfractions represents the CV error estimate.

3.1.1 Logic Regression (LogicR)

Logic Regression (LogicR) is an adaptive regression methodology developed for binary covariates by Ruczinski *et al.* (2003). It also applies to classification problems with binary predictors and response, i.e. $0-1$ encoded variables.

3.1.1.1 Logic model and its size

A logic model for classification depends on the outcome of a Boolean expression L. It assigns an object to class 1 if L is true and to class 0 otherwise. Given the observed feature vector of an object $\boldsymbol{x} \in \mathbb{R}^p$, the predicted class is defined as:

$$\hat{y}(\boldsymbol{x}) = I_{\{L(x)\ is\ true\}},$$

where I denotes the indicator function.

The logic model can be viewed as a logic tree, which is structured as follows: Boolean operators *and*, *or* are placed in the root and the inner knots; operands, thus binary predictors, or threshold conditions on the continuous predictors, or their logic complements, are placed in the terminal knots, which are called also leaves. The number of leaves represents the model size and quantifies the complexity of the logic model. A logic model is evaluated by visiting the associated tree for a particular realization of the feature vector, $\boldsymbol{x}$.

Example 1 (Logic rule I)**.** *In Schmitt (2005) eleven biomarkers encoded as* $M_1, \ldots, M_{11}$ *are given. For each biomarker* M_i *some optimal threshold* t_i *is provided,* $i = 1, \ldots, 11$. *The resulting binary predictors are defined as* $D_i = I_{\{M_i \geq t_i\}}$, $i = 1, \ldots, 11$. *Based on them, a logic model of size* 6 *is selected for the diagnosis of rheumatoid arthritis. This is given by the Boolean expression* $L = [(D_6 \wedge D_2) \vee D_4] \vee [(D_5 \wedge D_3) \wedge D_1]$. *If L is true for a new individual, this is classified as case. The tree-visualization of this logic model is illustrated in Figure 3.1.*

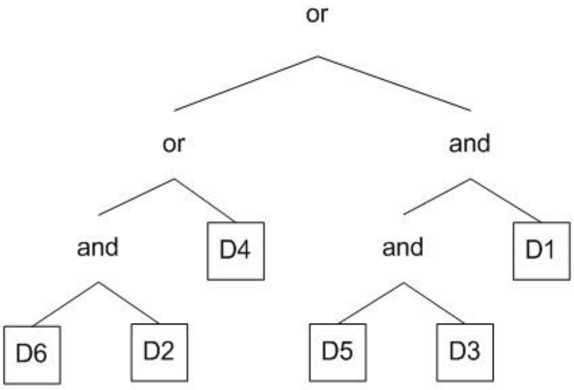


Figure 3.1: Graphical representation of a logic model.

3.1.1.2 Dichotomization

Since LogicR allows only for binary predictors, continuous predictors need to be first dichotomized in order to get the approach working. Within the range of each continuous predictor X_i, $i = 1, \dots, p$, a set of convenient thresholds (i.e. meaningful and easy to assess) should be ascertained. The difficulty to find meaningful thresholds, while exploiting as much information from continuous predictive variables as possible is pointed out by Schmitt (2005).

Two dichotomization approaches proposed by Schmitt (2005) are used: the first one employs empirical quantiles as thresholds (qLogicR); the second is based on the *best* threshold of a feature (btLogicR) in terms of misclassification rate which is obtained by Logistic Regression (Hosmer & Lemeshow, 2000).
Hence, the first approach guarantees a good interpretability of the logic model and the second approach has the potential to achieve very simple classification rules.

1. Quantiles-based Logic Regression (qLogicR)

The *quantiles-based* Logic Regression, short qLogicR, employs a set of empirical quantiles for dichotomization of continuous features, resulting in as many binary predictors (dummies) per feature, as quantiles in the set.

Example 2 (Logic rule II)**.** *Given two biomarkers M_1 and M_2 and c, d the* 0.95 *quantiles of their concentrations on the control population, a logic model for the diagnostic practice may be defined by the Boolean expression $L = (M_1 > c) \wedge (M_2 > d)$. Thus, an individual is classified as* *case if the observed concentrations of both markers exceed the corresponding* 0.95 *quantiles. Such a logic rule, that combines threshold conditions on two continuous features, is graphically represented by a set of rectangles or a step function traversing their scatter plot.*

If $\{q_1, q_2, \ldots, q_s\}$ is the set of empirical quantiles, then a feature X is replaced by a set of s binary predictors $TX_1, TX_2, \ldots, TX_s$ for the use with LogicR. These are given by $TX_i = I_{\{X \geq q_i\}}$, $i = 1, \ldots, s$.

2. Best threshold Logic Regression (btLogicR)

The *best threshold* Logic Regression, short btLogicR, employs for dichotomization of each continuous feature a single threshold, which is optimized with respect to the univariate classification problem based on that feature.
Let Ω be the set of objects to classify. Given $X : \Omega \to \mathcal{A} \subseteq \mathbb{R}$ a continuous predictive variable and $t \in \mathcal{A}$ a possible value of X, the discriminant rule $\delta_t : \mathcal{A} \to \{0, 1\}$ based on the feature threshold t is defined as:

$$\delta_t(X) = I_{\{X \ op \ t\}}, \tag{3.1.1}$$

where op is an operator in $\{\leq, \geq\}$.

The *best* feature threshold $t^* \in \mathcal{A}$ for btLogicR corresponds to the logistic discriminant rule δ_{t^*} that results in the minimal misclassification rate.

In contrast to (3.1.1) a reformulation of the discriminant rule in terms of class posteriors has the advantage of an explicit definition of op. With $P(X) := P(Y = 1|X)$ the conditional probability of class 1 given X, and $p \in [0, 1]$ a probability threshold, the posterior rule $\delta_p(P(X))$ is defined as:

$$\delta_p(P(X)) = I_{\{P(X) \geq p\}}. \tag{3.1.2}$$

Hence, this rule assigns an object to class 1 if the probability of class 1 given the observed feature x exceeds the threshold p.

Logistic discrimination allows for the switch between probability and feature thresholds by a simple transformation. If $(\hat{\beta}_0, \hat{\beta}_1)$ are the ML parameter estimates of the logistic model, then the *best* feature threshold is obtained from the *best* probability threshold p^* as:

$$t^* = \frac{\ln \frac{p^*}{1-p^*} - \hat{\beta}_0}{\hat{\beta}_1}.$$

The value of op in (3.1.1) is also implicitly determined by the sign of $\hat{\beta}_1$.

Feature X is replaced by a logic predictor TX, defined as the dichotomous outcome of the discriminant rule (3.1.1), in which t^* is used. Thus $TX = \delta_{t^*}(X)$.

3.1.1.3 Logic rule optimization

LogicR searches for good model candidates over the huge space of all Boolean expressions (so called logic terms) built as *and-or* combinations of the binary predictors and/or their complements. The classification performance of a logic model is quantified by its score, which is here the number of misclassified objects on the training data.

Using the tree-visualization of logic expressions, the best logic model is adaptively selected by simulated annealing (van Laarhoven & Aarts, 1987), which is a probabilistic search algorithm. At each step a possible operation on the current tree, like adding or removing a knot, is randomly proposed. This operation is always accepted if the new tree results in a smaller score, otherwise is accepted with a probability that depends on the difference between the scores of the old and the new tree and on the stage of the algorithm, indicated by the so-called temperature level. Run as a sequence of Markov chains with decreasing temperatures, the search is guided towards optimal scoring trees, assuming a proper number of iterations in each chain (Ruczinski *et al.*, 2003).

Annealing parameters. The annealing parameters *start*, *end* and *iter* should be set first. They represent the starting (highest) temperature, the finishing (lowest) temperature (both on the decimal logarithmic scale) and the total number of iterations over all annealing chains, respectively. In Kooperberg & Ruczinski (2006) it is recommended that the user should make some previous runs of LogicR on the data in use with the same setting of the annealing parameters and check if the best models have similar scores. If so, then the setting is used for the actual application, otherwise an adjustment of the annealing parameters should be made, in order to increase the stability of the results in the previous runs (see Kooperberg & Ruczinski (2006) for details).
On the one hand, a reasonable number of chains is needed in the so-called *crunch* time, where the number of acceptances is moderate, on the other hand the low and high temperatures should be adjusted to avoid spending time on accepting or rejecting almost every move. The setting is strongly dependent on the data at hand.

Logic model selection. The learning task for some LogicR approach starting from continuous predictors consists of:

(1) determination of the thresholds for feature dichotomizations with one of the methods qLogicR and btLogicR;

(2) determination of the tree structure given a fixed model size by simulated annealing, given the set of binary predictors established at (1);

(3) choice of the right model size.

Choice of the model size. The quality of the logic model is reflected not only by its score but also by its size. In general, the more leaves are allowed the better is the fit in terms of apparent error rates. To avoid over-fitting, the algorithm selects the model size using an approach based on cross-validation(CV).

Simulated annealing is used to determine the *best* logic model of a fixed size k on each of M CV training data sets. This procedure is repeated for each $k = 1, 2, \ldots, K$.
On each CV test and training data set the misclassification rates of the *best* logic trees for all model sizes are determined and then averaged over the CV loops. The model sizes leading to best models with a CV average test estimate within the ±0.01 interval around the minimal CV average test estimate over all models of different sizes are selected as candidates. Finally, the model size corresponding to the minimal gap between the CV average test and training estimates is selected as the *right* one out of the set of candidates.
The motivation for this rather complicated strategy is the following: the CV training error rates drop as the model size increases, while the CV test error rates decrease quickly until some adequate model size is reached; then they rise (with some perturbation) very slowly because of the noise. Consequently, there might be more model sizes resulting in a CV average test estimate very similar to the minimal one. Thus, a common CV optimization could lead to a model of larger size than necessary.

Choice of the feature panel. No additional optimization strategy, like for instance CV, is needed in LogicR for assessment of the best feature panel. The choice of the relevant binary predictors and therefore, of the thresholds corresponding to the original continuous features, is part of LogicR. Given a fixed model size, the feature selection occurs automatically by simulated annealing which chooses the way of moving (growing, pruning, alternating leaves) through the logic tree. The leaves of the final logic tree contain only threshold conditions on the relevant continuous predictors.

3.1.2 Regularized Discriminant Analysis (RDA)

The method Regularized Discriminant Analysis (RDA) is proposed by Friedman (1989) to improve the estimation of the misclassification risk in situations of singular or almost singular class covariance estimates. RDA is based on the assumption of multivariate normal class distributions.

Regularizations in two directions can be subsequently combined, namely shrinking the class estimates towards: (1) the pooled covariance estimate or (2) a multiple of the identity matrix. The degrees of shrinkage in these directions are controlled by the regularization parameters λ and γ, respectively. A more precise description of the shrinkage strategies used in RDA is offered in Chapter 4, Section 4.2.2.
The pair of regularization parameters (λ, γ) and the feature panel should be optimized with respect to the misclassification rate in order to get the best discriminant rule. For a fixed combination of features, the regularization parameters are found by minimizing the cross-validated misclassification rate over a two-dimensional grid on the unity square.
The optimal feature combination is found by an iterative feature selection algorithm that works in a forward manner. This is detailed also in Section 4.2.2.

3.1.3 Random Forests (RF)

In Random Forests (RF) (Breiman, 2001) successive independent trees are grown in a similar way to CART (Breiman *et al.*, 1984), but without pruning. Unlike CART, RF does not evaluate all variables at each knot in order to select the best split but only a randomly chosen subset of predictors. For RF there are two parameters to choose, the number of trees in the forest and the number of predictors randomly sampled at each node. Each combination of this number of predictors has the same probability to be chosen. These parameters are set once at the beginning of the algorithm.

In order to derive a reliable estimate of the misclassification error the initial training data is split for each tree randomly with replacement into a training set and the so-called Out-Of-Bag (OOB) test data. Based on this training set the tree is built and the OOB-samples are used to evaluate its performance in terms of misclassification rate. The OOB misclassification error of RF is the average over the misclassification errors of all trees obtained on the corresponding OOB-samples.

3.2 Data

The comparative study of the classification algorithms is performed on a real data set from the diagnostic practice and on simulated data sets.

3.2.1 Real data set

The real data set comprises rheumatoid arthritis (RA) cases and controls. The practical target is to find an effective biomarker panel for the early identification of RA.

For this study 594 patients are available, among them 272 RA-positive and 322 RA-negative (healthy or with disease conditions of similar symptomatic to RA). We have four markers, M_1 to M_4, whose serum concentrations are transformed on the decimal logarithmic scale to approach normality.

3.2.2 Simulated data sets

We simulate (1) a data structure favorable for the application of RDA but difficult for LogicR, (2) a data structure, on which LogicR is supposed to work well, while RDA to have difficulties, and (3) a data structure, which imitates the real data set, trying to approach its marker distributions. Following the example provided by the real data set, two informative $\{I_1, I_2\}$ and two less or non-informative features $\{N_1, N_2\}$ are considered in each simulation scenario.

For every simulated data structure 10 training data sets with 1.000 observations per class are generated. This is considered a reasonable sample size which allows each classification algorithm to perform smoothly. Thus, our conclusions regarding the performance of the algorithms is not influenced by the sample size.

3.2.2.1 RDA data structure

For the first data structure, 1.000 realizations of the feature vector $\boldsymbol{X} = (I_1, I_2, N_1, N_2)'$ are randomly sampled for each class i from a 4-dimensional normal distribution, $N(\boldsymbol{\mu}_i, \Sigma_i),\ i = 0, 1$.

The normal distributions are given by:

$$\boldsymbol{\mu}_0 = \begin{bmatrix} -2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \Sigma_0 = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{\mu}_1 = \begin{bmatrix} 0 \\ 2 \\ 0 \\ 0 \end{bmatrix} \Sigma_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The class means and unequal covariance matrices are chosen to differ only in the first two components of the feature vectors, corresponding to I_1 and I_2.

Figure 3.2 shows in the leftmost plot the two-dimensional projection of the first data set simulated according to this data structure on the informative coordinates I_1 and I_2. RDA is recommended in this case as the elliptic hulls of the two classes are perpendicular and the class means are well separated. This situation is obviously unfavorable for LogicR as the most natural separation would be done by a straight diagonal line rather than by a step function or a set of rectangular regions.

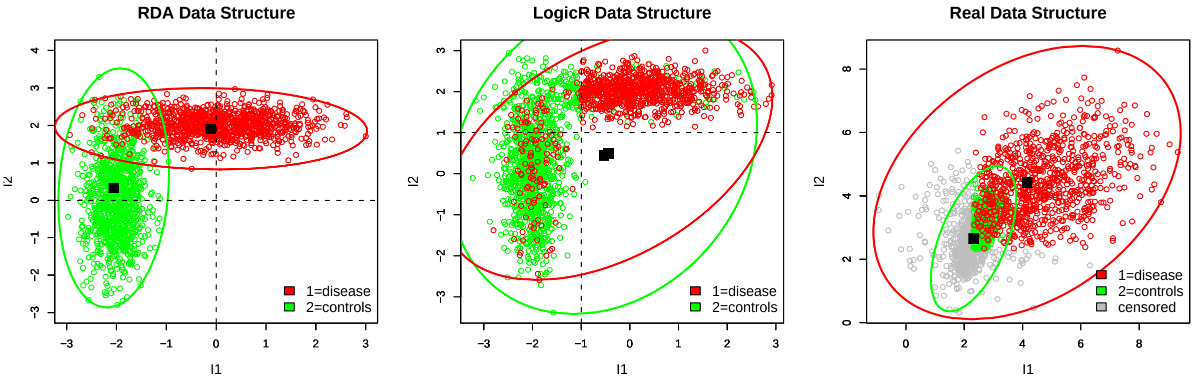


Figure 3.2: Informative coordinates for all simulation designs. *Green depicts the control class and red the disease class; in the simulation after real data the censored observations are distinguished by gray.*

3.2.2.2 LogicR data structure

A data situation is designed, on which LogicR is expected to work properly, while RDA to encounter difficulties in separating the classes. Initially 2.000 random realizations $\boldsymbol{x}$ of the 4-dimensional feature vector $\boldsymbol{X} = (I_1, I_2, N_1, N_2)'$ are drawn from the mixture of normals $0.5N(\boldsymbol{\mu}_0, \Sigma_0)$ + $0.5N(\boldsymbol{\mu}_1, \Sigma_1)$, where $N(\boldsymbol{\mu}_i, \Sigma_i),\ i = 0, 1$ are defined like in (3.2.2.1). Subsequently, the class membership y is assigned using the logic rule based on the theoretical medians of I_1 and I_2: if $L = (I_1 > -1) \wedge (I_2 > 1)$ is true, then the response is drawn from a Bernoulli(0.9)-distribution and otherwise from a Bernoulli(0.1)-distribution. Here it should be noted that the true error rate of the optimal Bayes rule (Hastie *et al.*, 2001) given this data setting is 10%.

Figure 3.2 shows in the middle plot the two-dimensional projection of the first data set simulated according to this data structure on the informative coordinates I_1 and I_2. Cases can be easily separated from controls by means of the right upper rectangle described by a logic rule using threshold conditions based on the medians of I_1 and I_2. This indicates that this distributional setting is favorable for the classification with LogicR. RDA seems to have weaker chances in this context, since the class means and the class elliptic hulls are strongly overlapped.

An interesting aspect for the comparative study is to check the classification potential of RDA and RF on such a data structure that is specially created to fit LogicR.

3.2.2.3 Real data structure

The data structure simulated at last is an attempt to recover a typical data structure for the diagnostic setting. The real data set for rheumatoid arthritis serves as model. The features I_1 and I_2 imitate the markers M_4 and M_2, while N_1 and N_2 are shaped after M_3 and M_1, respectively.

The multivariate distributions within the two classes in the real data set are approximated using the R package VGAM (Yee, 2007). The most plausible univariate distribution is selected for each marker after fitting several univariate distributions (intercept models with marker as response) and comparing their likelihood.
In the case of M_4, M_2, and M_3 the most evidence is provided for a left censored normal distribution, while in case of M_1 for the common normal distribution. The ML-estimates for the means of these normal distributions are computed. M_2, M_3 and M_4 present a lot of ties in the lower range. These are first removed, and then the class covariance matrix estimates are computed in order to prevent from an overestimation of the pairwise correlations of the markers.

Then 1.000 realizations of the feature vector $\boldsymbol{X} = (I_1, I_2, N_1, N_2)'$, aliasing $(M_4, M_2, M_3, M_1)'$, are randomly drawn per class from the multivariate normal distributions defined by these means and covariance matrix estimates of the classes:

$$\text{MVN}\ (\boldsymbol{\mu}_0, \Sigma_0): \quad \boldsymbol{\mu}_0 = \begin{bmatrix} 2.4 \\ 2.8 \\ 2.7 \\ 3.9 \end{bmatrix} \quad \Sigma_0 = \begin{bmatrix} 0.16 & 0.16 & -0.02 & -0.01 \\ 0.16 & 0.43 & -0.02 & -0.01 \\ -0.02 & -0.02 & 0.12 & 0.01 \\ -0.01 & -0.01 & 0.01 & 0.30 \end{bmatrix}$$

$$\text{MVN}\ (\boldsymbol{\mu}_1, \Sigma_1): \quad \boldsymbol{\mu}_1 = \begin{bmatrix} 4.3 \\ 4.2 \\ 3.2 \\ 4.3 \end{bmatrix} \quad \Sigma_1 = \begin{bmatrix} 3.02 & 0.97 & 0.26 & 0.06 \\ 0.97 & 1.40 & 0.24 & 0.18 \\ 0.26 & 0.24 & 0.58 & 0.23 \\ 0.06 & 0.18 & 0.23 & 0.64 \end{bmatrix}$$

Figure 3.2 shows in the rightmost plot the two-dimensional projection of the first data set simulated according to this data structure on the coordinates I_1 and I_2. Finally, I_1, I_2, and N_1 are censored to the left by 2.32, 2.31 and 2.31, respectively. These are the tied values in the lower range of M_4, M_2 and M_3 within the real data set. The censored objects are displayed by gray points in the scatter plot of I_1 and I_2, giving rise to the lower left rectangular corner of the data.

3.3 Comparative study design and results

For a fair comparative study of the classification algorithms the double-loop Monte Carlo cross-validation (MCCV) technique, which we highlighted in Chapter 2, is used. In case of Random

Forests, this technique needs an adjustment. The outer MC loops are combined with the OOB-technique instead of the inner CV.
The exact description of the used comparative designs is provided in the following.

3.3.1 Comparative study design on real data

The original data set is embedded in an 50×10-MCCV design for the comparative study of the algorithms. The proportion of $3 : 1$ is used to split the original data set into MC training and test data sets.

qLogicR. The approach qLogicR is used with the set of quantiles {0.5, 0.75, 0.90, 0.95} computed on the control population. The 0.90 and 0.95-quantiles are commonly used to define normal ranges of the biomarkers. The median and the upper quartile are chosen to allow for less informative splits in the tree, if necessary. Thus, four binary predictors are built per biomarker using this set of quantiles for dichotomizations.

Within every MC training data a 10-fold inner CV is used to estimate the *right* model size. On each CV training set the quantiles are determined and used to dichotomize the biomarkers on the CV training and test data sets.

Simulated annealing is applied to get the best logic trees of model sizes $1, 2, \ldots, 8$. The annealing parameters are determined in some previous runs of LogicR as it was explained in Section 3.1.1.3. They are $start = 3$, $end = -2$ and $iter = 150.000$, which means that the starting temperature is 1.000, the end temperature is 0.01 and the number of iterations per unit of temperature on the $log10$-scale is 25.000.

The best logic trees obtained for sizes $1, 2, \ldots, 8$ are applied on the CV training and test data sets and the average misclassification rates over the 10 CV loops are computed. The *right* model size between 1 and 8 corresponds to the minimal gap between the CV training and CV test average error rates, among those near the minimal CV test average.
Subsequently, the *best* logic tree of *right* size is grown after the same protocol (dichotomizations, simulated annealing, annealing parameters) on the MC training data.
Its performance is finally evaluated using the misclassification rate on the corresponding MC test data.

btLogicR. The approach btLogicR is used with one threshold per feature. This is computed in each CV loop on the CV training data set and in each MC loop on the MC training data set by logistic discrimination. Subsequently, the optimal thresholds are used to dichotomize the

continuous features on the training and on the corresponding test data set.
The best logic models of sizes $1, 2, \ldots, 8$ as well as the *right* model size are determined using the same workflow as described in case of qLogicR. The annealing parameters are *start* = 3, *end* = −2 and *iter* = 600.000.

Since the selection of the best feature panel is part of LogicR itself, we do not need any additional feature selection strategy for the qLogic and btLogicR approaches.

RDA. In RDA the regularization parameters (λ, γ) and the biomarker panel are optimized by a 10-fold CV on each MC training data set. The optimization grid for (λ, γ) is determined by 5 equidistant points over the interval [0, 1] on the x- and y-axis. The best biomarker combination is selected in a forward manner like detailed in Section 3.1.2.

RF. A tree is grown on a random stratified subfraction representing 90% of each MC training data set; two features out of the four available are sampled randomly at each knot. The terminal knots, of size 15, are used to assign the class by *majority voting*. The remained 10% of the samples build the OOB data set used to asses the performance of the forest.

This procedure is repeated 500 times for each MC training data set. The OOB estimate is used to assess the end performance.
The discriminatory power of the features is evaluated by four standard importance measures (Liaw & Wiener, 2002) available in R.

3.3.2 Comparative study design on simulated data

For each simulation scenario 10 data sets are generated: they are used both for training and assessment of the performance of the algorithms after the MCCV workflow. The comparative design consists of 20 outer MC and 5 inner CV loops for each simulated data set. Like in the real data application, the proportion of MC training samples to MC test samples is set at 3 : 1.

The algorithm qLogicR is applied with the set of quantiles $\{0.50, 0.75, 0.90, 0.95\}$ on the RDA data structure and $\{0.10, 0.25, 0.50, 0.75, 0.90\}$ on the other two data structures. The quantiles are computed from the samples of class 0.
The simulation annealing parameters are *start* = 3.5, *end* = −2, and *iter* = 360.000 for the RDA data structure, *start* = 3, *end* = −1, and *iter* = 600.000 for the LogicR data structure, and *start* = 2.5, *end* = −1.5, and *iter* = 500.000 for the real data structure.

For btLogicR the same setting of the annealing parameters, with *start* = 3, *end* = −2, and *iter* = 600.000, is used for all data structures.
Except for these differences, all algorithms are carried out according to the same workflow as described for the real data application.

3.3.3 Results

3.3.3.1 Real data results

On the real data, qLogicR and RF tend to outperform btLogicR (14.2% average misclassification rate) and RDA (14.5% average misclassification rate). RF is even slightly better than qLogicR (13.3% vs. 13.8% average misclassification rates).

The approach qLogicR provides in 28 out of 50 MC loops a one-leaf model. This assigns a patient to the disease class if the concentration of M_4 exceeds the 0.90 quantile of M_4 estimated on the control population of the corresponding MC training data set. The final logic rule provided by qLogicR with the whole training data is based on the 0.90 quantile of M_4 too. This has the value 2.39 on the $log10$-scale.

The approach btLogicR provides in 35 out of 50 MC loops a one-leaf model. This assigns a patient to the disease class if the concentration of M_4 exceeds the best threshold established by logistic discrimination on the corresponding MC training data set. The global value of the *best* threshold for M_4, established on the original training data set for the final logic rule, is 2.34 on the $log10$-scale.

Unlike the LogicR-approaches, RDA shows strong evidence for a combination of M_4 and M_2 over the 50 MC loops. The four standard importance measures of RF computed over the OOB samples indicate also clearly that M_4 and M_2 have a much higher discrimination potential than the other two markers.

Taking into account the simplicity of their rules and the average performance over the MC loops, both LogicR-approaches seem to be competitive to RDA and RF with respect to this particular diagnostic task.

3.3.3.2 Simulation results

Means, medians, standard deviations, and interquartile ranges for the misclassification rates are computed over the 10 simulated data sets and 20 MC loops of each simulation scenario. They are listed in Table 3.1 for each method and data structure.

Table 3.1: Summary of simulation results

Simulation design	**Method**	**Mean**(MCR)*	**SD**(MCR)	**Median**(MCR)	**IQR**(MCR)
RDA data structure	qLogicR	0.039	0.012	0.036	0.021
	btLogicR	0.045	0.006	0.046	0.007
	RDA	**0.025**	0.007	0.024	0.010
	RF	0.026	0.007	0.026	0.010
LogicR data structure	qLogicR	0.115	0.014	0.114	0.018
	btLogicR	0.104	0.007	0.103	0.011
	RDA	0.108	0.015	0.106	0.022
	RF	**0.102**	0.013	0.102	0.018
Real data structure	qLogicR	0.144	0.014	0.145	0.020
	btLogicR	0.132	0.015	0.135	0.023
	RDA	0.091	0.014	0.090	0.018
	RF	**0.090**	0.014	0.090	0.018

* MCR=misclassification rate.

On the RDA and real data structures both LogicR-methods lead to clearly poorer results than RDA and RF. RDA performs best on the RDA typical data structure, as it was expected.

The method btLogicR approaches well the true rate of 10% on the LogicR data structure. Both RDA and RF provide very good results for this data structure also, but btLogicR is still superior since it shows a higher stability, too.

On the real data structure both qLogicR and btLogicR show a performance loss from 9% to 14.4% and 13.2%, respectively, and a similar poor stability in comparison to RDA and RF.

On all data structures the results achieved with RDA and RF look very similar and indicate that both approaches are at least as good as LogicR with respect to the classification performance.
Under the RDA and LogicR data structures the method btLogicR outperforms all other methods regarding the classification stability. In contrast, qLogicR shows a high variability over all data structures.
The larger instability of qLogicR in comparison to btLogicR may be related to the size of the threshold set. Intuitively, it seems plausible that the more splits per feature are available, the more susceptible is LogicR to instability. However, a systematic verification of this hypothesis is impeded by the high computational time of LogicR.
The method qLogicR shows also a stronger bias than btLogicR on the LogicR and on the real data structures. It is possible to enhance the performance of qLogicR by the choice of a finer

threshold set in change of longer run times and some additional instability.
However, the results obtained by our simulations indicate that btLogicR is the better method with respect to the classification performance and stability as well.

The quantiles-based alternative gains its simple and interpretable rules at the price of maximally 5 percentage points performance loss (real data structure) but also of an increased instability in comparison to RDA, RF and btLogicR.

3.4 Conclusions

In this chapter specific requirements for the relevance of classification rules in the diagnostic context are addressed. The necessity of not only highly accurate but also simple and interpretable rules for the diagnostic research is highlighted.
Quality standards like parsimony and understandability and their implications with respect to the classification performance are exemplified by Logic Regression (LogicR). This is a new tree-based classification method with a great theoretical potential to provide simple, interpretable and therefore highly accepted rules in the medical practice.

The suitability of LogicR for the diagnostic context is investigated by a comparison with established classification algorithms. Regularized Discriminant Analysis (RDA) and Random Forests (RF) are used as reference methods as they are competitive algorithms for classification with more complex models.
The former method is able to provide an explicit formula for discrimination rules. However, this presumes computation of empirical class covariance, means and prior estimates. Also the regularized discriminant rule is not easily translatable for the medical practitioners.
The latter method belongs to the same group of tree-based classification methods like LogicR, but leads to a *forest* of simultaneous tree models. Thus, it is actually not suitable in practice, although it can provide very good results.

The benchmark of algorithms is performed on a real data set from the diagnostic practice. Supplementary information about the suitability of Logic Regression for diagnostic classifications is obtained with simulated data sets over three different data structures.
Since LogicR works exclusively with binary predictors, continuous features are first dichotomized in order to get the approach working. Two versions of LogicR are considered, based on different dichotomization strategies. The *quantiles-based* approach, qLogicR, uses a set of empirical quantiles computed on the control collective as splitting thresholds. The *best threshold* approach, btLogicR, uses an optimized splitting threshold per feature. For the latter dichotomization approach the optimal threshold is provided by Logistic Regression.

The method qLogicR leads to satisfactory results on the real data task. It is comparable in performance with a 500-trees RF and outperforms RDA mainly by single one-leaf logic trees.
However, these results of qLogicR are not reproducible on the simulated data sets. The method proves an overall poorer performance than RDA and RF. This happens not only in adversarial situations like those represented by the RDA or the real data structures, but also when the true model underlying the data is given by a simple logic rule except for 10% noise.

The theory of simulated annealing suggests that a high number of iterations within the *crunch* time might prevent from local optima and improve the stability of the results on a particular training data set. However, we make here a careful choice of the number of iterations in order to compromise between the requests of stability and feasible run time at least within the sampled CV training data sets.

The performance of LogicR may be influenced by the **set of thresholds** *used for dichotomizations. Given a meaningful set of thresholds on a training data set, the achievement of an approximately optimal scoring model depends on the number of iterations spent in the crunch time.*
The approach qLogicR is outperformed on the LogicR data structure by btLogicR. The latter is almost as good as RF on the data structure designed for LogicR. The true model underlying the LogicR data structure is defined by a logic expression with two threshold conditions.
However, btLogicR has more difficulties than qLogicR on the more complicated RDA data structure.
The method qLogicR covers by its set of thresholds a larger part of the feature range than does btLogicR with its optimized threshold. Therefore, it has the potential to approximate better more complex models (see here also the results on the real data set and on the real data simulation structure). In spite of this advantage, the performance of qLogicR is limited by the fixed and rough choice of the thresholds. A finer and more flexible grid over the feature range enhances its classification performance. However, this is related also to increased run times and induces additional instability. Therefore, it is not investigated further.

The stability of LogicR may be influenced by the **number of thresholds** *used for dichotomizations. Given a fixed number of thresholds on a training data set, the stability of the final model choice depends further on the number of iterations spent in the crunch time.*
It seems that the method btLogicR owes its better stability to its smaller set of thresholds. Already during the previous runs it can be clearly noticed that the same *optimally* scoring model is reached very fast also for a smaller number of iterations than the finally chosen one. In spite of this remark, the number of iterations is chosen large enough on one hand because the time-saving number of thresholds allows to invest more time in optimization than in case of qLogicR, and on the other hand to ensure a good model choice not only on the randomly selected CV training data sets, but on all training data sets in use. A deeper investigation about

how the size of the threshold set impacts on the stability of LogicR is left for future work due to the long run times required by such an analysis.

The results obtained on the real and simulated data sets in this chapter suggest that btLogicR is superior to qLogicR. It achieves a better stability and a similar or better performance. Its thresholds are also explicitly determined and optimized with regard to the given classification task.

Concerning the actual computation times, each RDA simulation design takes about 10 hours in SAS V8.2 (SAS-Institute, 2000), each RF simulation design about 10 minutes with the R package randomForest (Liaw & Wiener, 2002), and each LogicR simulation design about five days with the R package LogicReg (Kooperberg & Ruczinski, 2006).

LogicR is by far the most time expensive method, in spite of the small threshold list which is used for dichotomizations (of up to five quantiles in the case of qLogicR and of only one threshold in the case of btLogicR). So, the real burden of LogicR is the computational time, which impedes from the use of LogicR with annealing parameters and threshold sets that would have had the potential to enhance its performance and stability.

Now it depends especially on the diagnostic application if the accuracy loss of up to 5 percentage points in change of simpler and more understandable rules is still tolerable or not. We believe that an upgrade of the R implementation of Logic Regression would enhance in future the optimization possibilities and trigger a high acceptance of this method within the diagnostic frame.

Chapter 4

Learning diagnostic rules in case-control studies in the presence of known subclasses

The problem of non-representativeness or how and when to adapt classification algorithms to data with known subclass structure and known prevalences

In this chapter we address the problem of developing meaningful as well as reliable diagnostic rules when the target population is heterogeneous and its subclass prevalence structure is known.

Section 4.1 introduces practical challenges of diagnostic classifications when the collected data is subject to known sources of heterogeneity and the true subclass prevalences are known. Some issues from the statistical literature on this topic are described.

Section 4.2 highlights some weighting methods which are expected to provide more realistic estimates for the parameters of the heterogeneous class distributions as well for the performance of the diagnostic rule. Regularized Discriminant Analysis (RDA) (Friedman, 1989) is used to exemplify how the weighted estimates are computed and applied to account for the prevalence information while estimating parameters, feature panel and performance of a diagnostic rule. A detailed description of the feature selection algorithm based on RDA is also offered in this section.

Commonly a classification algorithm consists of three important phases: rule building, optimization and validation. Four different ways to alter a classification algorithm by using weighted parameter and/or performance estimates at different phases of the algorithm are introduced in Section 4.3. Final applications of the four resulting weighting algorithms based on the RDA feature selection algorithm are illustrated in Section 4.4. These are performed on real and simulated data sets and the results are used in a comparative survey of their classification performance.

Drawbacks and benefits of all weighting alternatives are summarized in Section 4.5.

4.1 Motivation

Any diagnostic rule is designed to predict the disease status of some certain population, which is usually referred as *target population*. For instance, a new screening tool should be developed in order to detect a complex disease (cancer, rheumatic disorders, Alzheimer) at an early stage, where no symptoms are available. In this case the target population refers to an ideal screening population that should comprise only asymptomatic patients.
However, the diagnostic rule is obtained on the *study population*, here also called *data at hand*. This represents a rather small excerpt of the target population for which the disease status was verified (e.g. by the gold-standard procedure). This comprises the whole information available for learning the diagnostic rule. For instance, this data might have been provided by a case-control study.

Characteristics of the target population are here referred as *true* while those of the study population as *observed* ones. Usually a problem in diagnostic classifications is that the study population is non-representative for the target population, which means that there is a large degree of mismatch between some observed and true population characteristics. This might have severe implications on the diagnostic rule conceived for the target population, regarding both its validity as well as the reliability of its performance estimates.

A special form of non-representativeness of the study population is the mismatch between the observed and true subclass prevalences. The discrepancy between study and target population may sometimes be very pronounced in this regard. Adequate modalities to account for the knowledge about the target subclass prevalences in the diagnostic rules are needed.

Throughout this work we assume that subclasses are explicitly known at the begin of the study (e.g. age, gender or tumor stages, progression groups, heterogeneous composition of the control panel, second diagnoses) and at least conjectured probabilities of their occurrence in the target population are available.

Two aspects of this problem should be carefully considered: the validity of the rules and the validity of the accuracy estimates computed to evaluate their classification performance. Ignoring strong sources of data heterogeneity leads almost certainly to learning suboptimal diagnostic rules. Now, assume that there is also some relevant discrepancy between the relative subclass sizes in the data at hand and the true subclass prevalences. Then neglecting the available information about the target subclass prevalence structure can lead both to invalid rules and invalid performance estimates of the rules.

As described by Sukhatme & Beam (1994) three situations of a-priori subclassification of the data may be distinguished:

(i) Both controls and cases are stratified by the same variable, e.g. gender;

(ii) Only cases are stratified having a particular disease condition, which might be of several types e.g. size of the lesion, tumor stage, tumor size, time from disease detection etc.;

(iii) Controls are stratified, but cases not, e.g. in a breast cancer study healthy women may be divided into two strata: women with no history of cancer or chronic diseases and women with benign breast conditions.

A practical example of situation (i) is a real study for screening colorectal cancer (CRC) by means of biomarker panels. The class of cases is further subdivided by the tumor stage; the class of controls consists not only of a group of healthy individuals, but also of some bowel and gastrointestinal disorders, available with known probabilities in the target population. The expected prevalences in the screening context are known for the disease subtypes as well as for disorders gathered by the group of controls.
It is likely that the multivariate distribution of the predictors (here the biomarkers) is subject to heterogeneity due to this clinical configuration (e.g. different pathological patterns associated to tumor stages, unequal fluctuations of marker concentrations in the context of different disorders in the control group).

An example of data with heterogeneous profile like in (iii) is offered by a diagnostic study for the early identification of rheumatoid arthritis (RA). This comprises 794 patients, which were recruited in five European centers at general practitioners' office (GP-data).
A reliable RA-screening rule based on biomarker measurements should be developed starting from the GP-data.
However, the class of non-RA patients has four subclasses, C_1 to C_4, which correspond to different disease conditions (osteoarthritis, crystal-induced synovitis, fibromyalgia and low back pain). They appear in an ideal GP-population of non-RA patients according to known prevalences. Table 4.1 lists the prevalences of these subclasses observed in the GP-data as well as the corresponding true prevalences. The discrepancy between the observed and the true subclass prevalences is negligible in case of the first two subclasses and pronounced in case of C_3 and

C_4. In Section 4.3 this example is used to illustrate our strategies to account for the available prevalence information in classification rules.

Table 4.1: GP-panel of RA-negatives: True and observed subclass prevalences (%)

Prevalence	C_1	C_2	C_3	C_4
observed ($\hat{\pi}_{ik}$)	5.56	4.86	24.30	65.18
true(π_{ik})	3.08	5.46	47.10	44.36

Adequate methods are needed to take into account the true subclass prevalence structure in order to develop reliable diagnostic rules for heterogeneous populations.

From a statistical point of view, the best way to account for the known subclass prevalences in the target population is to use an optimal stratification design in the phase of data collection. Where possible, this reduces the variance of final estimates of accuracy (Obuchowski & Zhou, 2002). But an explicit stratification design is based on the strong assumption that, before the study begins, one disposes of information that allows to stratify the clinical population according to the subclass structure and to sample the subclasses according to their true prevalences. However, sometimes it is impossible to design a study with stratified sampling, for the subclass variable can be assessed only after evaluation of the disease status by the diagnostic test (e.g. tumor size).

Also, this strategy is rather unfeasible in screening studies. The accuracy of the screening tests should be measured on the relevant clinical population, which comprises only asymptomatic subjects. Since any stratification may induce other unknown biases, the strategy of a non-stratified prospective sampling is especially recommended in this case. Although a prospective design would guarantee a good generalization of the diagnostic rule, logistic, economic and ethical reasons usually impede its usage.

For instance, a prospective design for screening a rare disease can require an enormous sample of patients to get even a small sample of cases. Thousands of true negatives may be part of the final sample. They must undergo both the diagnostic test and the verification of the true disease status, the *gold-standard* procedure. This is often invasive, risky (radiography), unpleasant (colonoscopy) or very expensive (mammography). Since true negatives are collected at most, the verification of the disease status is related to unnecessary high costs and psychological distress.

Therefore, one often decides to use a smaller amount of data, provided by a retrospective case-control study. This is a cheaper and faster alternative, since it is based on already available data (cancer registries, hospitals). Beside costs and time savings, its strengths are the informativeness and applicability to rather uncommon diseases. However, the case-control design can cause the final study population to have a sub-optimal clinical configuration, in which known

disease subclasses appear in biased proportions or are completely missing.
The screening situation was used in the practical applications of this chapter as it is the most obvious example of a case, in which there are many good reasons not to carry out a prospective or a representative stratified data collection. Similar reasons can be found in the context of differential diagnosis, with different importance among them depending on the setting.

Referring to such situations of retrospective data collection in the context of known heterogeneity, some publications highlight the benefits of *post-stratification*. For example the size of a tumor (or the disease stage) can be used to stratify the collective of cases in a cancer study about the *prognostic* ability of some marker. This information is known usually after the cases have been selected, so a real stratified sampling is not possible.
Sukhatme & Beam (1994) propose to account for the variability of predictors between different subclasses in a retrospective manner. The data, which was collected without stratification concerns, is treated as if it were a stratified sample, using subclass-specific weights to improve the accuracy estimates. However, the benefits of stratified sampling in the recruitment phase, like smaller variance of the accuracy estimates, is also pointed out.

We propose some ways to use the available information about the true subclass prevalences already in the phase of rule development. A practical challenge is to be able to design a reliable screening tool starting from a case-control population. Some weighting algorithms are designed to perform not only a retrospective correction of the final accuracy estimates, but also a correction of the diagnostic rule itself.

4.2 Learning RDA classifiers with misrepresented subclasses

In Section 4.1 we mentioned some approaches existent in the statistical literature for dealing with the problem of building classification rules when the target population is heterogeneous with known subclass prevalence structure. They use either the principle of stratified sampling at the begin of the study or they account retrospectively for the true subclass prevalences by weighting error estimates in the phase of rule validation (post-weighting).

In this section we introduce some methods to account for the known true subclass prevalences already in the phase of designing the diagnostic rule.

4.2.1 Definitions and notations

First, the more general context of a diagnostic classification problem with two heterogeneous classes is regarded. The classes are encoded as 1 and 2, with the usual interpretation in the diagnostic practice as disease and control class, respectively. The terminology and common ways to tackle general two-class classification problems are shortly reminded to enable the understanding of the weighting approaches.

Two assumptions are made about the structure of the data used for building classification rules:

(1) each class is a mixture of K_i different subclasses; if just one class is subject to heterogeneity, then the number of subclasses in the other class is set at 1;

(2) the true subclass prevalences (probabilities to occur) within the corresponding class are available; they are generally denoted as $(\pi_{ik})_{k=\overline{1,K_i}}$ and sum up to one, i.e. $\sum_{k=1}^{K_i} \pi_{ik} = 1,\ i = 1,2$.

N objects should be assigned to one of two given classes by means of the information provided with the data set at hand. This information consists of N observations of form $z_n = (y_n, \boldsymbol{x}_n')',\ n = 1,\ldots,N$. The vector of p observed object characteristics, also called feature vector, is denoted as $\boldsymbol{x}_n = (x_{n1},\ldots,x_{np})' \in \mathbb{R}^p$. The true class of the n'th object is y_n. The set of i-labeled objects is denoted as C_i and its size as N_i.

The predictive data lies in a p-dimensional subspace Ω of $\mathbb{R}^p$ which is also called *feature space*. A discriminant rule defines a partition of Ω into two regions and assigns every new object to a class according to which region of the partition its observed feature vector $\boldsymbol{x}_0$ belongs to.

The regions determined by a discriminant rule should accurately approximate an ideal partition of Ω which corresponds to some optimal Bayes rule and provides the best separation of the classes in the feature space.

A common discriminant rule is that of the maximal posteriors. This assigns an object with observed feature vector $\boldsymbol{x}_0$ to the class i, whose probability of occurrence given $\boldsymbol{x}_0$, i.e. $P(i|\boldsymbol{x}_0)$, is maximal. If the class density functions are known, then this rule can be reformulated by a simple application of the Bayes theorem in the context of a $\{1,2\}$-valued class outcome as:

"Classify the object with observed characteristics $\boldsymbol{x}_0$ into class 1 if

$$\frac{f_1(\boldsymbol{x}_0)\pi_1}{f_2(\boldsymbol{x}_0)\pi_2} \geq 1 \tag{4.2.1}$$

and to class 2 otherwise",

whereby f_1 and f_2 stand for the known class densities, π_1 and π_2 for the class priors.

Given K_i subclasses within class i, C_{ik} stands for the set of objects belonging to subclass $k,\ k = 1,2,\ldots,K_i$ and N_{ik} for its size.

The position of these subclasses to each other and to the opposite class as well as the degree of mismatch between true and observed subclass prevalences decide about the complexity of the classification problem and also about its limitations (in terms of unavoidable misclassifications). They define the target subclass structure.

Our idea is to weight classification algorithms according to the true subclass prevalences not only in the phase of rule validation, but in the phases of rule building and optimization as well. The aim is to adapt the diagnostic feature panel, the parameter estimates and the discrimination cutoff of the final model, as well as to correct the estimates of diagnostic accuracy with respect to the target population, starting from the suboptimal data at hand.

A training data set $\mathcal{L}$ for rule construction and optimization and an independent validation data set $\mathcal{T}$ for evaluation of the rule performance are assumed to be available. On $\mathcal{L}$ a cross-validation substructure is used to optimize the rule parameters and feature panel. We consider only subclass stratified cross-validation (see Chapter 2, Section 2.2).

Three weighting alternatives are presented which address different levels of a feature selection algorithm. The true subclass prevalences can be applied as weights on

(1) **CV training** data sets, thus in the process of rule building, for correction of the estimates of class distribution parameters;

(2) **CV test** data sets, thus in the process of rule optimization, for correction of the performance estimate of a rule with fixed regularization parameters and features;

(3) **Validation** data set, for correction of the performance estimate of the final rule which was constructed in (1) and optimized in (2).

An established feature selection algorithm based on Regularized Discriminant Analysis (RDA) (Friedman, 1989) is used to exemplify these weighting alternatives.

4.2.2 RDA classifiers in general

The weighting methods used to account for the known subclass prevalence information in the target population are exemplified by means of RDA-rules, which are derived from the rule of maximal posteriors under the assumption of multivariate class Gaussians.

Theoretically, our weighting strategies can be applied to any classification algorithm. However, they were exemplified only on the forward feature selection algorithm based on Regularized Discriminant Analysis (RDA). This was shortly introduced in Chapter 3, Section 3.1.2. In this section a deeper insight into the functionality of RDA is offered to ease understanding of the weighting procedures.

Since RDA generalizes established discriminant analysis methods like Linear Discriminant Analysis (LDA) (Fisher, 1936) and Quadratic Discriminant Analysis (QDA) (McLachlan, 1992), the same weighting procedures apply also to feature selection algorithms based on the latter two methods. We opted for RDA in order to cover a larger family of shapes of the decision boundaries, suspecting that for heterogeneous populations a better separability could be reached with regularized rather than with linear or quadratic surfaces.

The RDA rule is obtained from the rule of maximal posteriors (4.2.1) replacing $f_1(\boldsymbol{x})$ and $f_2(\boldsymbol{x})$ with multivariate normal density functions. Subsequently, a transformation on the natural logarithmic scale, enables the transcription of this rule in terms of so-called *discriminant scores*:

$$d_i(\boldsymbol{x}_0) = \Delta(\boldsymbol{x}_0; \boldsymbol{\mu}_i, \Sigma_i) + \ln|\Sigma_i| - 2\ln\pi_i, \; i = 1, 2. \tag{4.2.2}$$

Here $\Delta(.; \boldsymbol{\mu}_i, \Sigma_i), \; i = 1, 2$ stand for the Mahalonobis-distances from the observed feature vector $\boldsymbol{x}_0$ to the class means μ_i, and Σ_i for the class covariance matrices, $i = 1, 2$. The discriminant rule based on maximal posteriors (4.2.1) is reformulated as:
"Classify the object with observed characteristics $\boldsymbol{x}_0$ into class 1 if

$$d_1(\boldsymbol{x}_0) \leq d_2(\boldsymbol{x}_0)$$

and to class 2 otherwise".
Thus, it requires the assignment of an object to the class with the minimal discriminant score, given its observed vector of characteristics.

Regularization techniques (Friedman, 1989) were developed to handle special situations like weak collinearity (almost linearly dependent features), ill- or poorly-posed classification problems (number of parameters to estimate exceeds or is comparable to the number of samples). These result often into singular or almost singular covariance estimates. Almost singularity gives rise to very small, negatively biased, eigenvalues. Only a negligible variance of them can cause an explosion in the variance of the inverted covariance matrix and result in a high instability of classification. The severeness of this problem is additionally enhanced using plug-in estimates of the eigenvalues. These are usually negatively biased for small eigenvalues and positively biased for large eigenvalues.
Regularization attempts a reduction in the variability of discriminant scores induced by the plug-in estimates of the class covariance matrices. Different techniques can be used to manipulate the class covariance estimates for this target.

RDA combines two regularization steps, based on the parameters $\lambda, \; \gamma \; \in [0, 1]$:

(1)

$$\hat{\Sigma}_i(\lambda) = \frac{(1-\lambda)(N_i - 1)\hat{\Sigma}_i + \lambda(N-2)\hat{\Sigma}_{pooled}}{(1-\lambda)(N_i - 1) + \lambda(N-2)}, \tag{4.2.3}$$

(2)

$$\hat{\Sigma}_i(\lambda,\gamma) = (1-\gamma)\hat{\Sigma}_i(\lambda) + \gamma\frac{tr(\hat{\Sigma}_i(\lambda))}{p}I_p. \tag{4.2.4}$$

where $N_i,\ i = 1,2$, are the class sizes and N the total sample size. Also, $\hat{\Sigma}_{pooled}$ stands for the pooled covariance matrix, a linear combination of the class covariance estimates and $tr(\hat{\Sigma}_i)$ for the trace (sum of diagonal elements) of the covariance estimate $\hat{\Sigma}_i$.
The first equation corresponds to a shrinkage of the class covariance estimates towards the pooled covariance matrix and the second to a shrinkage of the λ-regularized class covariance estimates against a multiple of the identity matrix. The degrees of shrinkage in these two directions are tuned by the regularization parameters λ and γ.

The first regularization step provides more stable class covariance estimates at the price of a small bias injection. It can be easily shown that QDA and LDA are special cases of this regularization, as they can be obtained for $\lambda = 0,\ \gamma = 0$ and $\lambda = 1,\ \gamma = 0$, respectively.

The second regularization step attempts to counteract the inherent bias coupled to the plug-in estimation of the eigenvalues of the class covariance matrices by shifting them towards their mean values, $\frac{tr(\hat{\Sigma}_i)}{p}$.

The pair of regularization parameters (λ,γ) and the feature panel should be optimized with respect to the misclassification error rate in order to get the best discriminant rule. For a fixed combination of features the regularization parameters are found by minimizing the cross-validated misclassification error rate over a two-dimensional grid on the unity square.
The best feature combination is herein obtained by an iterative feature selection algorithm that works in a forward manner (Schmitt, 2005).
Consider that the current model is given by the feature combination $\mathcal{I}$, while the set of not yet selected features is $\mathcal{O}$. Then, at one step of the algorithm, $\mathcal{I}$ is extended by a feature F from $\mathcal{O}$. Conditioned on the combination $\mathcal{I} \bigcup \{F\}$ the regularization parameters are optimized by the grid method. This is repeated for each feature in $\mathcal{O}$. The cross-validated misclassification error rates of the optimal discriminant rules determined with RDA for the extensions of the current model with each of the features from $\mathcal{O}$ are compared. The not yet selected feature resulting in the minimal CV estimate is added to the model.
This algorithm ends when no relevant improvement of the misclassification error rate relatively to the current model can be achieved with any extension based on one of the remained features.

4.2.3 Weighting of parameter estimates for rule building

The first weighting alternative applies to the CV training data sets. Thus, it accounts for the target subclass prevalence structure in the phase of rule building.

Discriminant scores are based on plug-in estimates of class means, covariance matrices and class priors. The known true subclass prevalences are used to adapt class mean and covariance estimates to the target situation. Thus, this weighting approach aims at a correction of the estimates of class distribution parameters for the bias induced when the known subclass structure and true subclass proportions within each class are neglected.

Two ways are proposed for weighting on the CV training data sets:

(1) *Corrected CV Training*: Weighted mean and covariance estimates of the distribution parameters in the heterogeneous class are computed as weighted sums of their subclass analogs. The true subclass prevalences are used as weights.

(2) *Inflated CV Training*: In order to resemble the target population subclasses are sampled at random from each CV training data set according to their true prevalence - if enough items of a subclass are available sampling is done without replacement, otherwise with replacement.

4.2.3.1 Corrected CV Training

The common plug-in estimates of the class means are the empirical class means:

$$\hat{\boldsymbol{\mu}}_i = \bar{\boldsymbol{x}}_i := \frac{1}{N_i} \sum_{v \in C_i} \boldsymbol{x}_v, \tag{4.2.5}$$

with $\boldsymbol{x}_v$ the observed characteristics of object v and N_i the size of class i in $\mathcal{L}$. These are already weighted sums of the subclass means. However, they use the relative subclass sizes in the data set at hand as weights. Thus:

$$\begin{aligned} \hat{\boldsymbol{\mu}}_i &= \frac{1}{N_i} \sum_{k=1}^{K_i} \sum_{v \in C_{ik}} \boldsymbol{x}_v = \sum_{k=1}^{K_i} \frac{N_{ik}}{N_i} \bar{\boldsymbol{x}}_{ik} \\ &= \sum_{k=1}^{K_i} \hat{\pi}_{ik} \hat{\boldsymbol{\mu}}_{ik} \end{aligned} \tag{4.2.6}$$

where N_{ik} is the size of subclass k within class i, $\hat{\pi}_{ik}$ stands for the observed prevalence of subclass k within class i, and $\hat{\boldsymbol{\mu}}_{ik},\ k = 1, 2, \ldots, K_i$, are the common plug-in estimates of the subclass means in $\mathcal{L}$.

The ML-estimates of the class covariance matrices can be also rewritten as a sum between some weighted sum of the ML-estimates of the subclass covariance matrices and an additional quadratic term. See terms T_1 and T_2 in (4.2.7), respectively. Again, the natural weights used in these typical class covariance estimates are the subclass proportions $\hat{\pi}_{ik}$ within the corresponding classes of the training data $\mathcal{L}$. Denoting the ML-estimates of class and respectively subclass covariance matrices as $\hat{\Sigma}_{i,ML}$ and $\hat{\Sigma}_{ik,ML}$, it holds:

$$
\begin{aligned}
\hat{\Sigma}_{i,ML} =& \frac{1}{N_i} \sum_{v \in C_i} (\boldsymbol{x}_v - \hat{\boldsymbol{\mu}}_i)(\boldsymbol{x}_v - \hat{\boldsymbol{\mu}}_i)' \\
=& \frac{1}{N_i} \sum_{k=1}^{K_i} \sum_{v \in C_{ik}} \left[(\boldsymbol{x}_v - \hat{\boldsymbol{\mu}}_{ik}) + (\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_i)\right]\left[(\boldsymbol{x}_v - \hat{\boldsymbol{\mu}}_{ik}) + (\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_i)\right]' \\
=& \frac{1}{N_i} \sum_{k=1}^{K_i} [\sum_{v \in C_{ik}} (\boldsymbol{x}_v - \hat{\boldsymbol{\mu}}_{ik})(\boldsymbol{x}_v - \hat{\boldsymbol{\mu}}_{ik})' + \underbrace{\sum_{v \in C_{ik}} (\boldsymbol{x}_v - \hat{\boldsymbol{\mu}}_{ik})(\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_i)'}_{=0} + \\
&+ \underbrace{\sum_{v \in C_{ik}} (\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_i)(\boldsymbol{x}_v - \hat{\boldsymbol{\mu}}_i)'}_{=0} + \sum_{v \in C_{ik}} (\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_i)(\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_i)'] \\
=& \frac{1}{N_i} \left[\sum_{k=1}^{K_i} N_{ik} \hat{\Sigma}_{ik,ML} + \sum_{k=1}^{K_i} N_{ik} (\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_i)(\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_i)' \right] \\
=& \sum_{k=1}^{K_i} \frac{N_{ik}}{N_i} \hat{\Sigma}_{ik,ML} + \sum_{k=1}^{K_i} \frac{N_{ik}}{N_i} \left(\hat{\boldsymbol{\mu}}_{ik} - \frac{\sum_{l=1}^{K_i} N_{il} \hat{\boldsymbol{\mu}}_{il}}{\sum_{l=1}^{K_i} N_{il}} \right) \left(\hat{\boldsymbol{\mu}}_{ik} - \frac{\sum_{l'=1}^{K_i} N_{il'} \hat{\boldsymbol{\mu}}_{il'}}{\sum_{l'=1}^{K_i} N_{il'}} \right)' \\
=& \sum_{k=1}^{K_i} \frac{N_{ik}}{N_i} \hat{\Sigma}_{ik,ML} + \sum_{k=1}^{K_i} \frac{N_{ik}}{N_i} \left(\frac{\sum_{\substack{l=1 \\ l \neq k}}^{K_i} N_{il} (\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_{il})}{N_i} \right) \left(\frac{\sum_{\substack{l'=1 \\ l' \neq k}}^{K_i} N_{il'} (\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_{il'})}{N_i} \right)' \\
=& \sum_{k=1}^{K_i} \hat{\pi}_{ik} \hat{\Sigma}_{ik,ML} + \sum_{k=1}^{K_i} \hat{\pi}_{ik} \left\{ \left[\sum_{\substack{l=1 \\ l \neq k}}^{K_i} \hat{\pi}_{il} (\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_{il}) \right] \cdot \left[\sum_{\substack{l'=1 \\ l' \neq k}}^{K_i} \hat{\pi}_{il'} (\hat{\boldsymbol{\mu}}_{ik} - \hat{\boldsymbol{\mu}}_{il'})' \right] \right\} \\
=& \sum_{k=1}^{K_i} \hat{\pi}_{ik} \hat{\Sigma}_{ik,ML} + \sum_{k=1}^{K_i} \hat{\pi}_{ik} \left\{ \left[(1, 1, \ldots, 1) \bigotimes \hat{\boldsymbol{\mu}}_{ik} - (\hat{\boldsymbol{\mu}}_{i1}, \hat{\boldsymbol{\mu}}_{i2}, \ldots, \hat{\boldsymbol{\mu}}_{iK_i}) \right] (\hat{\pi}_{i1}, \hat{\pi}_{i2}, \ldots, \hat{\pi}_{iK_i})' \right\} \\
& \left\{ \left[(1, 1, \ldots, 1) \bigotimes \hat{\boldsymbol{\mu}}_{ik} - (\hat{\boldsymbol{\mu}}_{i1}, \hat{\boldsymbol{\mu}}_{i2}, \ldots, \hat{\boldsymbol{\mu}}_{iK_i}) \right] (\hat{\pi}_{i1}, \hat{\pi}_{i2}, \ldots, \hat{\pi}_{iK_i})' \right\}' \\
=& \underbrace{\sum_{k=1}^{K_i} \hat{\pi}_{ik} \hat{\Sigma}_{ik,ML}}_{:=T_1} + \underbrace{\sum_{k=1}^{K_i} \hat{\pi}_{ik} (A_{ik} - M_i) \hat{\boldsymbol{\Pi}}_{1,\ldots,K_i} \hat{\boldsymbol{\Pi}}'_{1,\ldots,K_i} (A_{ik} - M_i)'}_{:=T_2} . \qquad (4.2.7)
\end{aligned}
$$

The following notations are used:

$$\begin{array}{rcl} M_i & := & (\hat{\boldsymbol{\mu}}_{i1}, \hat{\boldsymbol{\mu}}_{i2}, \dots, \hat{\boldsymbol{\mu}}_{iK_i}) \in \mathbb{R}^{p \times K_i} \\ A_{ik} & := & (\hat{\boldsymbol{\mu}}_{ik}, \hat{\boldsymbol{\mu}}_{ik}, \dots, \hat{\boldsymbol{\mu}}_{ik}) \in \mathbb{R}^{p \times K_i} \\ \hat{\boldsymbol{\Pi}}_{1,\dots,K_i} & := & (\hat{\pi}_{i1}, \hat{\pi}_{i2}, \dots, \hat{\pi}_{iK_i})' \in \mathbb{R}^{K_i}. \end{array}$$

M_i is the matrix with the subclass mean vectors within class i as columns, A_{ik} is the matrix obtained from M_i by replacement of each subclass mean with the mean of some particular subclass k of class i and $\hat{\boldsymbol{\Pi}}_{1,\dots,K_i}$ is the vector of observed subclass prevalences within class i.

The so-called weighted class means and covariance estimates are obtained from (4.2.6) and (4.2.7), respectively, using the true instead of the observed subclass prevalences.
Thus, they are defined as:

$$\hat{\mu}_{i,w} = \sum_{k=1}^{K_i} \pi_{ik} \hat{\mu}_{ik}. \qquad (4.2.8)$$

and

$$\hat{\Sigma}_{i,w} = \sum_{k=1}^{K_i} \pi_{ik} \hat{\Sigma}_{ik} + \sum_{k=1}^{K_i} \pi_{ik} (A_{ik} - M_i) \boldsymbol{\Pi}_{1,\dots,K_i} \boldsymbol{\Pi}'_{1,\dots,K_i} (A_{ik} - M_i)'. \qquad (4.2.9)$$

Here $\boldsymbol{\Pi}_{1,\dots,K_i}$ is the vector of true subclass prevalences within class i, thus $\boldsymbol{\Pi}_{1,\dots,K_i} = (\pi_{i1}, \pi_{i2}, \dots, \pi_{iK_i})'$ for $i = 1, 2$.

Since the discriminant scores of RDA are computed by the most software packages with unbiased rather than ML covariance estimates, the weighted covariance estimates $\hat{\Sigma}_{i,w}$ are first transformed into their unbiased analogs by multiplication with the factor $\frac{N_i}{N_i - 1}$ and then used in (4.2.2).

4.2.3.2 Inflated CV Training

With inflated CV training weighting of class covariance and mean estimates is done in an implicit way. The observed subclass proportions within classes are adapted by a random and subclass stratified sampling to approach the subclass prevalence structure in the target population. This procedure is repeated for each CV training data set.

Expected subclass sizes are computed starting from the observed class size and using the true subclass prevalences ($EN_{ik} = \pi_{ik} N_i$). Sampling is performed with replacement (subclass inflation) when the expected subclass size exceeds the observed subclass size, i.e. $EN_{ik} > N_{ik}$. Otherwise sampling succeeds without replacement (subclass deflation). In this artificial way,

a poorly represented subclass may be expanded, while a highly represented one may be compressed to its expected size. Since we usually deal with small observed subclass sizes, both sampling types are here referred in a generic way as *data inflation*.
The weighted class and covariance estimates are those obtained as natural plug-in estimates on the inflated data sets.

The CV training data sets in the frame of an M-fold CV are inflated in the following way: the M CV test data sets are inflated first and each reunion of $M-1$ inflated CV test data sets represents the inflated training counterpart of the remained CV test data set.
Each CV training data set preserves its definition as reunion of $M - 1$ disjoint CV test data sets also after inflation. All M possible reunions of inflated CV training and inflated CV test counterparts lead to the same inflated version of the original training data, which is used in the end to get the final rule.

Some pitfalls of this *weighting-by-inflation* procedure should be, however, noted:

- If a subclass is over-represented in the data at hand, then deflation causes information loss.
- If a subclass has a high known prevalence in the target population, but is under-represented within the data at hand, then inflation causes ties; they can have negative (numerical) effects in the estimation of parameters.
- Ties enhance the feature correlations within the subclass under consideration; then one might overweight a very biased subclass covariance estimate in the computation of the corresponding class covariance estimate.

Also, our inflation strategy is performed basically on the CV test data sets, which are usually small. It should be checked in advance if the expected subclass sizes are adequate for a reliable estimation of the subclass distribution parameters.
If this is not the case, then the size of the CV test data sets can be augmented by a factor f to ensure the minimal necessary expected size of each subclass. Using the previous notations - $N_1^{(m)}, N_2^{(m)}$ class sizes, $N_{ik}^{(m)}$, $i = 1, 2$, $k = 1, \ldots, K_i$, subclass sizes in the CV test data, $m = 1, \ldots, M$ - this means that an integer factor $f > 1$ is needed such that the expected subclass sizes within the CV test data sets, i.e. $EN_{ik}^{(m)} = \pi_{ik}(fN_i^{(m)})$, are adequate.
The CV test data sets are inflated according to the recalculated expected sizes. Then, each combination of $M - 1$ inflated CV test data sets provides an inflated CV training data set. This is consequently augmented by the same factor.

4.2.4 Weighting of performance estimates for optimization and validation

Empirical estimates of statistical measures like misclassification error, sensitivity, specificity, AUC, pAUC etc. (Pepe, 2003) are used to judge the classification performance, thus the ability of a classification rule to identify the right class. They support in this way the process of rule validation.
Used within a feature selection algorithm, they help also to compare rule candidates in order to decide upon the most suitable rule configuration in terms of rule parameters and features. Thus, they support the process of rule optimization, too.
True subclass structure and prevalences can be used to appropriately adjust these estimates to the target situation. This is expected to increase the reliability of optimal rules and of the associated estimates of classification performance.

Most statistical or machine learning algorithms base their rule optimization on the misclassification rate as measure of algorithm's performance. For this reason, and also for the sake of simplicity, our weighting alternatives are exemplified using the misclassification rate. Other performance measures like sensitivity, specificity, AUC, etc. are not reviewed in this work, but the general weighting principles presented here apply to their estimates in a similar way, too.

The **first weighting alternative** introduced in Section 4.2.3 uses the true subclass prevalences to weight the class mean and covariance estimates in the computation of discriminant scores.
The next two weighting alternatives replace the natural weights, thus the observed subclass prevalences, by the true subclass prevalences in the computation of the error estimates.
From now on, all error estimates which are naturally weighted by means of the observed subclass prevalences are called *unweighted*. Under *weighted* error estimates we understand all error estimates in which the true subclass prevalences have replaced the observed ones.

The **second weighting alternative** uses the true subclass prevalences on the CV test data sets, to weight the cross-validation (CV) error estimates. Thus, it accounts for the known target subclass prevalence structure in the process of rule optimization.
Two ways are proposed to weight on the CV test data sets:

(1) *Corrected CV Test*: In each CV test data set the misclassification error estimate is computed as a weighted sum of the subclass misclassification rates using true subclass prevalences as weights.

(2) *Inflated CV Test*: In each CV test data set subclasses are sampled at random with or without replacement according to their true prevalence.

The **third weighting alternative** uses the true subclass prevalences to weight the test-sample (TS) error estimates. Thus, the available prevalence information is exploited in the phase of rule validation.

The final weighted error estimate is a sum of its subclass analogs with coefficients given by the true subclass prevalences. This strategy is called *Corrected Test*.

Considering *Inflated CV Test* as self-explanatory, only the weighting strategies *Corrected CV Test* and *Corrected Test* are formally described in the following.

The principles of common CV and TS estimation of the misclassification error were presented in more detail in Section 2.2, in (2.2.1) and (2.2.3), respectively. Further, the same notations are used.

4.2.4.1 Corrected CV Test

Starting from (2.2.1) it results that in the context of class heterogeneity the common CV estimates can be expressed as:

$$\begin{aligned}
\hat{\epsilon}_i^{(M-CV)} =& \frac{1}{M}\sum_{m=1}^{M}\frac{1}{N_i^{(m)}}\sum_{k=1}^{K_i}\left[\sum_{(i,\boldsymbol{x}')'\in\mathcal{L}^{(m)}\bigcap C_{ik}} I_{\{\delta_{-\mathcal{L}^{(m)}}(\boldsymbol{x})=j\}}\right]\\
=& \frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K_i}\frac{N_{ik}^{(m)}}{N_i^{(m)}}\frac{1}{N_{ik}^{(m)}}\left[\sum_{(i,\boldsymbol{x}')'\in\mathcal{L}^{(m)}\bigcap C_{ik}} I_{\{\delta_{-\mathcal{L}^{(m)}}(\boldsymbol{x})=j\}}\right]\\
=& \frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K_i}\hat{\pi}_{ik}^{(m)}\hat{\epsilon}_{ik}^{(m)}. \qquad (4.2.10)
\end{aligned}$$

Here an M-fold cross-validation is considered. For every $m = 1, \ldots, M$, $N_i^{(m)}$ is the size of class i, $N_{ik}^{(m)}$ the size of subclass k within class i, and $\hat{\epsilon}_{ik}^{(m)}$ the misclassification rate in subclass k of class i in the CV test subfraction $\mathcal{L}^{(m)}$. The weight $\hat{\pi}_{ik}^{(m)}$ is the observed prevalence of subclass k within class i in the CV test subfraction $\mathcal{L}^{(m)}$. This is approximately the same in every CV test subfraction, as long as the cross-validation procedure is stratified by subclass.

The weighted CV error estimate of class i is obtained by replacing each natural subclass weight $\hat{\pi}_{ik}^{(m)}$ with the true weight π_{ik}:

$$\hat{\epsilon}_i^{(M-CV),w} = \frac{1}{M}\sum_{m=1}^{M}\sum_{k=1}^{K_i}\pi_{ik}\hat{\epsilon}_{ik}^{(m)},\ i \in \{1,2\}. \qquad (4.2.11)$$

The theoretical principle of weighting was here introduced in a general frame with K_i known subclasses within class i, $i = 1, 2$. However, the theoretical and practical investigations in this work are carried out for the simplified version of a two-class classification problem with a heterogeneous control (labeled as C or 2) and a homogeneous disease class (D or 1). Another simplification is achieved by assuming that the heterogeneous class possesses just two subclasses, C_1 and C_2. Thus K_1 is 1 and K_2 is 2.

In this particular case, the weighted CV error estimate of the heterogeneous class C is:

$$\hat{\epsilon}_2^{(M-CV),w} = \frac{1}{M}\sum_{m=1}^{M}[\pi_{21}\hat{\epsilon}_{21}^{(m)} + \pi_{22}\hat{\epsilon}_{22}^{(m)}] \tag{4.2.12}$$

For the homogeneous class D the unweighted CV error estimate $\hat{\epsilon}_1$ is used further.
The final weighted CV error estimate is:

$$\hat{\epsilon}^{(M-CV),w} = \hat{\pi}_1\hat{\epsilon}_1^{(M-CV)} + \hat{\pi}_2\hat{\epsilon}_2^{(M-CV),w}.$$

This error estimate is used instead of the unweighted one to optimize the rule in some of the weighting algorithms introduced in this chapter. Weighting the CV error estimates, on which the rule optimization (i.e. the choice of the optimal rule parameters and feature panel) is based, the feature selection algorithm is potentially focused towards a feature panel that fits better the classification task when a subclass structure is given.

4.2.4.2 Corrected Test

Starting from (2.2.3) it can be easily shown that the TS-estimates are naturally weighted by means of the observed subclass prevalences within the test data $\mathcal{T}$, i.e. $\hat{\pi}_{ik}^{(ts)}$, $k = 1, 2, \ldots, K_i$:

$$\begin{aligned}\hat{\epsilon}_i^{(ts)} &= \frac{1}{N_i^{(ts)}}\sum_{k=1}^{K_i}\sum_{(i,\boldsymbol{x}')'\in C_{ik}} I_{\{\delta(\boldsymbol{x})=j\}} \\ &= \sum_{k=1}^{K_i}\frac{N_{ik}^{(ts)}}{N_i^{(ts)}}\hat{\epsilon}_{ik}^{(ts)} \\ &= \sum_{k=1}^{K_i}\hat{\pi}_{ik}^{(ts)}\hat{\epsilon}_{ik}^{(ts)}.\end{aligned} \tag{4.2.13}$$

Here $N_i^{(ts)}$ is the size of class i, $N_{ik}^{(ts)}$ is the size of subclass k within class i, and $\hat{\epsilon}_{ik}^{(ts)}$ is the misclassification rate in subclass k of class i in the validation data $\mathcal{T}$.

The weighted TS-estimate is:

$$\hat{\epsilon}_i^{(ts),w} = \sum_{k=1}^{K_i} \pi_{ik}\hat{\epsilon}_{ik}^{(ts)},\ i \in \{1,2\}. \tag{4.2.14}$$

Now, consider the particular situation with class D homogenous and class C heterogeneous with two subclasses. In this particular situation, the common TS error estimate in the heterogeneous class C, $\hat{\epsilon}_2^{(ts)}$, is given by:

$$\hat{\epsilon}_2^{(ts)} = \hat{\pi}_{21}\hat{\epsilon}_{21}^{(ts)} + \hat{\pi}_{22}\hat{\epsilon}_{22}^{(ts)}. \tag{4.2.15}$$

Therefore, the weighted TS error estimate of the heterogeneous class C is obtained as:

$$\hat{\epsilon}_2^{(ts),w} = \pi_{21}\hat{\epsilon}_{21}^{(ts)} + \pi_{22}\hat{\epsilon}_{22}^{(ts)}. \tag{4.2.16}$$

The final weighted TS error estimate is:

$$\hat{\epsilon}^{(ts),w} = \hat{\pi}_1\hat{\epsilon}_1^{(ts)} + \hat{\pi}_2\hat{\epsilon}_2^{(ts),w}.$$

Weighting the TS error estimates, on which the final assessment of the rule performance is based, one attempts to correct them for the situation in the target population.

4.3 Benchmark of algorithms on simulated data

In the last two sections we proposed three weighting alternatives: (1) on the CV training data sets, (2) on the CV test data sets and (3) on the validation data. In the first weighting alternative the true subclass prevalences are used to weight the estimates of the distribution parameters in the heterogeneous class. In the second weighting alternative the CV estimates of misclassification error are weighted by the true subclass prevalences. In the third weighting alternative the final error estimate is weighted by the true subclass prevalences.

These weighting alternatives are combined starting from the *RDA* feature selection algorithm (Schmitt, 2005). They lead to four weighting algorithms of interest. Their classification performance is investigated and compared by means of a simulation study.

4.3.1 Selected weighting algorithms

We consider also the situation in which no weighting is performed on the CV training and/or on the CV test data sets. This results in two other strategies, which we call *As-It-Is CV Training* and *As-It-Is CV Test*, respectively. They enable a comparison between weighted and unweighted algorithm versions.

Recalling the strategies which were proposed for the realization of the three weighting alternatives, we have the possibilities:

- **CV training data**
  - (a1) *Corrected CV Training*
  - (a2) *Inflated CV Training*
  - (a3) *As-It-Is CV Training*
- **CV test data**
  - (b1) *Corrected CV Test*
  - (b2) *Inflated CV Test*
  - (b3) *As-It-Is CV Test*
- **Validation Data**
  - (c) *Corrected Test*

There are only six meaningful possibilities to define a weighting algorithm using a combination of these strategies. Only four are considered for further investigation. They are listed with an explanation of their acronyms in Table 4.2 and are interpreted below.

Table 4.2: Selected weighting algorithms

Design	CV Training Data	CV Test Data	Validation Data
AAC*	As-it-is	As-it-is	Corrected
I I C	Inflated	Inflated	Corrected
ACC	As-it-is	Corrected	Corrected
CCC	Corrected	Corrected	Corrected

* reference method.

The combination **AAC** corresponds to the usual unweighted approach, which builds the classification rule without keeping track of any potentially existent data structures. It performs only a *post-weighting* (see Section 4.1) of the final classification rule in the validation phase.

The combination **ACC** involves unweighted estimates of class distribution parameters in the phase of rule building. However, a *post-weighting* is performed in this case both for the unweighted rules obtained on CV training data sets and for the final unweighted rule obtained on the whole training data $\mathcal{L}$.

The combination **IIC** uses unweighted estimates from inflated CV training and test data sets in the phase of rule building and optimization, respectively. The final rule is constructed on the inflated version of the original training data $\mathcal{L}$, too. Consequently, this algorithm weights implicitly the class distribution parameters and the CV estimates of performance, enforcing the true subclass prevalence structure by random and subclass stratified sampling.
The combination **CCC** applies the true subclass prevalence explicitly in the computation of class distribution parameters and CV estimates of performance.
Finally both IIC and CCC weight explicitly on the independent validation data set to get reliable estimates of the rule performance under the targeted conditions (*post-weighting*).

The other two other possible ways to combine the weighting strategies on CV training, CV test and validation data, III and ICC, are not considered, since they essentially resemble IIC. Consequently, they are expected to provide very similar results to IIC.
Besides, IIC is sufficient as example from this group of inflation-based methods. The data inflation procedure is actually not desirable in practice because of the earlier mentioned pitfalls. Although the method ICC applies the data inflation only on the CV training data sets, IIC is still preferred providing inflated CV training and test data sets which are perfectly concordant with the classical cross-validation principle.

4.3.2 Simulation design

Four weighting algorithms to account for the true subclass prevalence structure while developing classification rules are considered for further investigation. Using the acronyms defined in Table 4.2 these were AAC, ACC, IIC and CCC. A simulation study is conducted to establish the individual performance of each weighting algorithm and to perform a comparison of algorithms in terms of their classification ability. This should also help to understand how four interesting factors affect the performance of the weighting algorithms and enable a comparison of algorithms over various factor settings.

The simplified scenario with a heterogeneous control class (labeled as C or 2) and a homogenous disease class (labeled as D or 1) is considered. The heterogeneous control class is assumed to consist of only two subclasses, C_1 and C_2. These are expected to come up with certain probabilities π_{21} and π_{22}, respectively, within the control class of the target population.
Regarding the notations, subclass distribution parameters or subclass prevalences are indexed

either by the numerical (21, 22) or by the character labels (C_1, C_2) of the corresponding subclasses. These alternative notations are equivalent.
The relative subclass sizes in the data at hand are assumed to be equal $(\hat{\pi}_{21} = \hat{\pi}_{22})$, this case being often encountered in the diagnostic practice. Consequently, a measure of discrepancy between the true subclass prevalences quantifies the mismatch between target and study population with respect to the subclass prevalence structure, too.

Four continuous features, M_1 to M_4, are available for classification. The first two features are assumed to be informative, while the last two features to be non informative for the class and subclass discrimination.

Some factors describing the subclass structure in the target population are suspected to impact on the classification performance. Situations in which one weighting algorithm outperforms the others may depend on the particular setting of some of these factors.

Our simulation design (see Section 2.3) targets the simultaneous analysis of the relationship between the continuous response of each weighting algorithm, here some relative measure of performance, and the following four factors:

(1) $\Pi_{C_1,C_2} = |\pi_{C_1} - \pi_{C_2}| \in [0, 0.8]$ - the absolute difference between the true prevalences of the control subclasses C_1 and C_2;

(2) dist.DC$_1$ = $||\mu_{C_1} - \mu_D||_2 \in [1, 4]$ - the euclidian distance between the centers of the disease class and the first control subclass, $||.||_2$ denotes the euclidian norm $\mathcal{L}_2$;

(3) dist.DC$_2$ = $||\mu_{C_2} - \mu_D||_2 \in [1, 4]$ - the euclidian distance between the centers of the disease class and the second control subclass;

(4) $\widehat{C_1DC_2} = \sphericalangle(DC_1, DC_2) \in [0^0, 180^0]$ - the angle described by the subclass centers with vertex in the center of the disease class.

These continuous factors are encoded such that their lowest/highest values corresponds to $-1/+1$. This is helpful for generating an appropriate simulation design with JMP (SAS-Institute, 2005).
The first three factors are regarded as categorical two-level factors while the fourth one is analyzed on three levels.
An overview of the actual values behind the final low and high factor levels is given in Table 4.3.

The lowest value of the first factor is 0, being achieved when both subclasses appear equally often in the control target population, thus when $\pi_{C_1} = \pi_{C_2} = 0.5$. The highest value is set to 0.8, when one subclass is rather rare $(\pi_{C_1} = 0.1)$ and the other represents the majority $(\pi_{C_2} = 0.9)$.

The lowest value for the euclidian distances between the center of the disease class and the centers of the control subclasses is set to 1 while the highest value to 4.

The fourth factor, being an angle, is explored not only at the extremities $\{0^0, 180^0\}$, when means are collinear, but also in the middle, when subclass centers form with the center of the disease class a right angle.

Table 4.3: Overview of the factor levels

Factor Level	Π_{C_1,C_2} (π_{21}, π_{22})	dist.DC$_1$	dist.DC$_2$	$\widehat{C_1DC_2}$
Low (-1)	$0\ (0.5, 0.5)$	1	1	0
Center (0)	$0.4\ (0.3, 0.7)$	2.5	2.5	$\frac{\pi}{2}$
High (1)	$0.8\ (0.1, 0.9)$	4	4	π

It should be clearly distinguished between the meaning of the considered design factors. The angle factor and the euclidian distances describe the position of class and subclasses relatively to each other within the feature space. They enable a meaningful description of the degree of heterogeneity in the data. The prevalence factor only provides information about the true dimensions of the heterogeneous class components.

The goal of our weighting algorithms is to adjust classification rules for the effect of data non-representativeness with respect to the true subclass prevalences. Thus, a successful weighting should primarily lead to results that do not anymore depend on the subclass prevalences in the target population.

The first target of this simulation is to check which of the four design factors impact on the bias of the weighted misclassification rates and adjust the design by excluding some eventually non-relevant factors.

The second target is to survey under which design configurations the use of a particular weighting approach provides a real benefit in comparison to the others. On one hand, this is done by a direct comparison of the relative bias of the algorithms at each design point. On the other hand, an interesting aspect is to investigate the expected changes in predicted values due to individual and overall factor variations. A good model for prediction of the relative bias of each algorithm by means of the selected factors is necessary in order to be able to reliably extend the explorative benchmark over other factor settings.

Therefore, a sequential two-step-design is used.

In the **first step** the target conditions with a relevant impact on the performance of the algorithms are searched. For this purpose a D-optimal screening design (see Section 2.3), denoted as DOE1, is generated with the Custom Designer in the JMP Software (Version 7.0) (SAS-Institute, 2005).

DOE1 comprises 16 common points and 4 center points. The latter are used to assess the lack of fit of the final model and establish if replicates are needed for further experiments.
The combinations of factor levels for the 20 experimentation points of DOE1 are presented in Table 4.4. It can be noticed that this design contains the center points 1, 6, 7 and 16 and some other ties, namely two pairs of replicates (9, 13) and (12, 17).

Table 4.4: **DOE1: Screening design (D-optimal)**

Row	**Design Point**	Π_{C_1,C_2}	dist.DC$_1$	dist.DC$_2$	$\widehat{C_1DC_2}$	**true error**(ϵ_B)**
1	**1, 6, 7, 16***	0	0	0	0	0.1334
2	**2**	1	−1	1	−1	0.0707
3	**3**	−1	1	−1	−1	0.2029
4	**4**	−1	1	1	1	0.0341
5	**5**	1	1	1	1	0.0254
6	**8**	1	1	−1	0	0.2925
7	**9, 13**	−1	−1	1	0	0.2115
8	**10**	1	1	−1	1	0.3014
9	**11**	1	1	1	0	0.0281
10	**12, 17**	−1	−1	−1	1	0.3959
11	**14**	−1	1	−1	0	0.2129
12	**15**	1	−1	−1	−1	0.3104
13	**18**	1	−1	−1	0	0.3263
14	**19**	1	−1	1	1	0.0718
15	**20**	−1	1	1	−1	0.0231

* center points.
** true error rate (computation details in Section 4.3.4).

The last column of the table specifies the true error rate corresponding to some optimal Bayes rule on the target population associated to each design point (Hastie *et al.*, 2001). The true error rate is needed to calculate the bias of the misclassification error estimate obtained with each weighting algorithm. Details on its computation are given in Section 4.3.4. The error rates presented at rows 1, 7 and 10 of Table 4.4 represent averages over the true error rates obtained for the replicates.

The **second step** targets a reliable model for the prediction of each weighting algorithm's performance. Here main, quadratic and two-factor interaction effects of the factors selected in the first step are regarded. Therefore, DOE1 has been augmented to an I-optimal design with 16 further experimental points in which all initial factors have been preserved. This was also done with the JMP Software (Version 7.0) using the Augment Designer tool.

The combinations of factor levels corresponding to the additional design points are listed in Table 4.5. No other center points have been added since the response variability obtained at the center points in DOE1 was moderate.

Table 4.5: **DOE2: Design augmentation (→ I-optimal)**

Row	**Design Point**	Π_{C_1,C_2}	dist.DC$_1$	dist.DC$_2$	$\widehat{C_1DC_2}$	**true error**(ϵ_B)*
1	**21**	−1	1	0	1	0.0899
2	**22**	0	−1	0	0	0.2040
3	**23**	0	−1	0	1	0.2238
4	**24**	0	1	0	0	0.0986
5	**25**	0	−1	−1	−1	0.3029
6	**26**	1	0	0	1	0.1253
7	**27**	0	0	1	0	0.0741
8	**28**	−1	0	1	1	0.0883
9	**29**	−1	−1	0	−1	0.2238
10	**30**	1	1	0	−1	0.0975
11	**31**	0	0	1	−1	0.0556
12	**32**	−1	0	0	−1	0.1046
13	**33**	1	0	0	−1	0.1064
14	**34**	0	0	−1	1	0.3083
15	**35**	−1	0	−1	0	0.2532
16	**36**	0	1	−1	−1	0.2532

* true error rate (computation details in Section 4.3.4).

The data generation workflow for both the screening and the augmented design is described in the following section.

4.3.3 Simulated data

The necessary data for this simulation study is generated at each design point in two steps:

(1) A large data set mimicking the target population is randomly drawn from the distributions assumed at each design point according to the true subclass prevalences.

(2) A data set mimicking some eventually non-representative excerpt of the target population is afterwards randomly drawn without replacement from the target population simulated at (1); equal subclass prevalences in the control class are assumed.

The subclass membership is used as stratification variable in both steps.

In simulation step (1) for each of the 36 points in the augmented design (DOE1 $\bigcup$ DOE2) 10.000 four-dimensional feature vectors are generated per class in order to approach well the target population.
The distribution of class D at point s of the simulation design is assumed to be multivariate normal $MVN(\boldsymbol{\mu}_{s,D}, \Sigma_{s,D})$. The distribution of class C at point s is a mixture of the multivariate normal distributions of subclasses C_1 and C_2, denoted as $MVN(\boldsymbol{\mu}_{s,C_1}, \Sigma_{s,C_1})$ and $MVN(\boldsymbol{\mu}_{s,C_2}, \Sigma_{s,C_2})$, respectively. The mixing proportions are given by the subclass prevalences $(\pi_{s,21}, \pi_{s,22})$ associated to the actual level of the prevalence factor Π_{C_1,C_2} at design point s.

Based on the 10.000 observations per class the true error rate is estimated. This is used to judge the ability of the weighting algorithms to provide reliable rules and performance estimates.

In the target population classes C and D are set equally prevalent $\pi_{s,1} = \pi_{s,2} = 0.5$, while subclasses C_1 and C_2 show up with the probabilities $\frac{\pi_{s,2k}}{2}, k = 1, 2$, for all design points $s = 1, 2, \ldots, 36$. Therefore, the original subclass sizes are $\pi_{s,21} \cdot 10.000$ and respectively $\pi_{s,22} \cdot 10.000$. The simple case of identical spherical class and subclass covariance matrices is considered. They are given by the identity matrices $\Sigma_{s,D} = \Sigma_{s,C_1} = \Sigma_{s,C_2} = I_4$, $s = 1, \ldots, 36$.
Starting from the factor settings at each design point s, i.e. the euclidian distances between the center of class D ($\boldsymbol{\mu}_{s,D}$) and the subclass centers of C_1 and C_2 ($\boldsymbol{\mu}_{s,C_1}$, $\boldsymbol{\mu}_{s,C_2}$) as well as the angle $\widehat{C_1 D C_2}$ between DC_1 and DC_2, the class and subclass centers $\boldsymbol{\mu}_{s,D}$, $\boldsymbol{\mu}_{s,C_1}$ and $\boldsymbol{\mu}_{s,C_2}$, are straightforward computed. However, they are not unique.
In order to restrict the set of solutions for the class and subclass centers given a certain factor setting the assumption is made that $\boldsymbol{\mu}_{s,D} = (0, 0, 0, 0)'$. Thus, the center of the disease class is set at the origin.

Only the first two features are informative for class and subclass discrimination. Thus, every subclass center differs from the center of the opposite class at most in the first two coordinates, e.g. $\boldsymbol{\mu}_{s,C_1} = (x, \epsilon, 0, 0)'$ and $\boldsymbol{\mu}_{s,C_2} = (\epsilon', y, 0, 0)'$.

The first two components of the mean vectors are determined by solving the equation system:

$$\begin{aligned} \sqrt{x^2 + \epsilon^2} &= dist.DC_1 \\ \sqrt{\epsilon'^2 + y^2} &= dist.DC_2 \\ < (x, \epsilon), (\epsilon', y) > &= \cos(\widehat{C_1 DC_2}) \cdot dist.DC_1 \cdot dist.DC_2 \end{aligned}$$

Finally, the restrictions $\epsilon = 0$ and $x > 0$ lead to a unique solution at each design point:

$$\begin{aligned} x &= dist.DC_1 \\ y &= dist.DC_2 \cdot \sin(\widehat{C_1 DC_2}) \\ \epsilon' &= dist.DC_2 \cdot \cos(\widehat{C_1 DC_2}) \end{aligned}$$

In step (2), from the 10.000 samples in each class of the target population obtained in step (1) at some design point s, 1.000 observations are drawn at random without replacement. The subclass membership is used to stratify sampling and C_1, C_2 are drawn in equal proportions, receiving 500 samples each.

The 2.000 samples drawn in step (2) build the actual data set used for classification at the design point s.

In general, classification algorithms use proportional priors of the subclasses (i.e. estimated from the data) although these may be incompatible with the reality. The subclass sizes are chosen equal within each data set generated in step (2) as if no information about the real subclass proportions in the target population were available. The degree of suboptimality of the data at hand is tuned by means of different settings of the target subclass prevalences.

In this simulation the main focus is not a comparison of rules with respect to a particular diagnostic task, but a comparison of the weighting algorithms with respect to their classification performance. Therefore the computation of a final misclassification error estimate on an independent validation data is not needed and all samples generated at step (2) are used for training.

4.3.4 Comparative design on simulated data

On the $S = 36$ learning data sets created in simulation step (2) the four weighting algorithms known as AAC, IIC, ACC and CCC are applied and their classification performances are eval-

uated and compared.
Obviously, at design points characterized by equal true subclass prevalences or by completely overlapped subclasses ($\boldsymbol{\mu}_{s,C_1} = \boldsymbol{\mu}_{s,C_2}$) the methods AAC, ACC and CCC are identical. Theoretically, the method IIC should be in this context also identical to the other methods, since at these design points no sampling from subclasses is actually necessary. Yet IIC is still performed by sampling with or without replacement according to the level of the prevalence factor in order to check for peculiar aspects in its application. If available, such aspects are easier to detect in cases where all weighting methods must perform identically.

On every of the 36 data sets of the augmented design, comprising 2.000 samples each, the competing algorithms AAC, IIC, ACC and CCC are run within an estimation-optimization MCCV design with 50 outer MC and 10 inner CV loops. Each data set is split 50 times by subclass stratified random sampling without replacement into an MC training and an MC test data set in a proportion of 3 : 1.

The MC training data sets are used to fit repeatedly the optimal discrimination rule with every algorithm. On each MC training data set a 10-fold CV is applied to find the optimal RDA rule in terms of minimal misclassification rate, thus to get the optimal regularization parameters and feature panel. The weighted misclassification rates for every algorithm are computed as average over the weighted misclassification rates obtained on the 50 MC test data sets. Figure 4.1 presents the MCCV analysis flowchart used at each design point.

The MC training data sets are denoted as $\mathcal{L}^{(l)}$ and the MC test data sets as $\mathcal{T}^{(l)}$, $l = 1, 2, \ldots, 50$. At each design point $s = 1, 2, \ldots, 36$ and for every weighting algorithm, weighted test sample estimates are obtained on the 50 MC test data sets $\mathcal{T}^{(l)}$. They are denoted as $(\hat{\epsilon}^{W}_{\mathcal{T}^{(l)},\ algo}(s))_{l=1,2,\ldots,50}$, where *algo* stands for one of the algorithms AAC, ACC, IIC or CCC.
The final weighted test estimates of misclassification error at each design point for each weighting algorithm are obtained as averages over the 50 weighted MC test results:

$$\hat{\epsilon}^{W}_{algo}(s) = \frac{1}{50} \sum_{l=1}^{50} \hat{\epsilon}^{W}_{\mathcal{T}^{(l)},\ algo}(s) \tag{4.3.1}$$

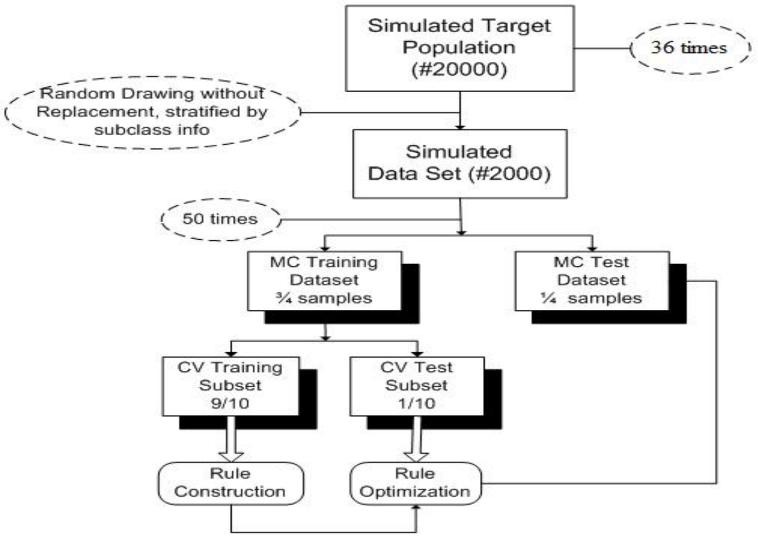

Simulated Target Population (#20000)
36 times
Random Drawing without Replacement, stratified by subclass info
Simulated Data Set (#2000)
50 times
MC Training Dataset ¾ samples
MC Test Dataset ¼ samples
CV Training Subset 9/10
CV Test Subset 1/10
Rule Construction
Rule Optimization

Figure 4.1: MCCV design for the comparison of weighting algorithms on the simulated data.

Based on these final weighted test estimates a response is defined to compare the ability of the weighting algorithms to find classification rules and to provide estimates of their performance that approach well reality. The response is given at each design point s by the relative bias (RB) of the final weighted test estimates of misclassification error with respect to the true error $\epsilon_B(s)$.

$$\hat{\epsilon}^W_{RB,\ algo}(s) = \frac{\hat{\epsilon}^W_{algo}(s) - \epsilon_B(s)}{\epsilon_B(s)}, \ \forall s \in \{1, \dots, 36\}. \tag{4.3.2}$$

The absolute bias of the weighted misclassification error estimate $|\hat{\epsilon}^W_{algo}(s) - \epsilon_B(s)|$ provides a good measure to judge the importance of the benefit achieved with one weighting algorithm, therefore it represents a response alternative that is worth considering.

The true error rate $\epsilon_B(s)$ is computed from the target population simulated in step (1) at point s. The Bayes'Theorem is used to derive it using the distributions of D, C_1 and C_2 in the target population, which are all assumed to be multivariate Gaussians. At iteration point s the density functions in C_1, C_2 and D given an observation vector $\boldsymbol{x}$ from the target population is:

$$f(\boldsymbol{x}|\boldsymbol{\mu}_{s,ik}, \Sigma_{s,ik}) = \frac{1}{(2\pi)^{p/2}\sqrt{\det \Sigma_{s,ik}}} \exp\big(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_{s,ik})' \Sigma^{-1}_{s,ik}(\boldsymbol{x} - \boldsymbol{\mu}_{s,ik})\big) \tag{4.3.3}$$

where $i \in \{1, 2\}$ represents the class, $k = 1, 2, \dots, K_i$ the subclass and $K_i = i$.
According to the Bayes'Theorem the a-posteriori class probabilities given an observed vector of object characteristics $\boldsymbol{x}$ are assessed using the relationship:

$$P(i|\boldsymbol{x}) = \frac{\pi_i f_i(\boldsymbol{x})}{\pi_1 f_1(\boldsymbol{x}) + \pi_2 f_2(\boldsymbol{x})}, \; i = 1, 2, \tag{4.3.4}$$

where f_1 and f_2 represent the class densities and π_1 and π_2 the class priors.
Since equal class priors are chosen at all design points both in the target and in the study population ($\pi_1 = \pi_2 = 0.5$), the a-posteriori class probabilities depend only on the class densities. Replacing in (4.3.4) the class density functions with their expressions as mixtures of subclass density functions we get:

$$P(i|\boldsymbol{x}, s) = \frac{\sum_{k=1}^{K_i} f(\boldsymbol{x}|\boldsymbol{\mu}_{s,ik}, \Sigma_{s,ik})\pi_{s,ik}}{\sum_{j=1}^{2} \sum_{l=1}^{K_j} f(\boldsymbol{x}|\boldsymbol{\mu}_{s,jl}, \Sigma_{s,jl})\pi_{s,jl}}, \; i = 1, 2. \tag{4.3.5}$$

The class posteriors are estimated using the subclass multivariate normal densities from (4.3.3) in (4.3.5).
At each design point s, the optimal Bayes rule used to determine the true error rate on the target population is given by the condition:

$$\delta_{Bayes}^{opt}(\boldsymbol{x}, s) = \begin{cases} 1 & \text{if } P(1|\boldsymbol{x}, s) \geq 0.5, \\ 2 & \text{otherwise.} \end{cases} \tag{4.3.6}$$

A comparison between the true and the class labels assigned by this rule over the 20.000 samples of the target population yields an estimate of the true error rate.
A location near to zero and a small variation of the relative bias $\hat{\epsilon}_{RB,algo}^{W}$ over the design points provides evidence for a generally suitable weighting method. At each design point s, the winner among the weighting algorithms is established. It is defined as the algorithm that results into the smallest absolute value of the relative bias $\hat{\epsilon}_{RB,algo}^{W}(s)$.

4.3.5 Application and results

For the distributional configuration associated to each design point the true error is calculated according to (4.3.6) using the 20.000 observations in the corresponding target population. The true error rates corresponding to the 20 points of $DOE1$ and to the 16 points of $DOE2$ are listed in the last columns of Tables 4.4 and 4.5, respectively.

Then the actual response, i.e. the relative bias of the average weighted misclassification rate with respect to the true error rate, is assessed at each design point. The absolute bias is used to compare the importance of the benefits achieved with these different methods.

4.3.5.1 Graphical approach DOE1

The response values obtained with all algorithms for all design points of DOE1 are listed in Table 4.6 together with the winner method. This method results into the minimal absolute relative bias among all methods.

Table 4.6: DOE1: Relative bias per design point (average over 50 MC loops). *Blue stands for similar results of AAC and ACC, red for similar results of CCC and IIC.*

Design Point(s)	$\hat{\epsilon}^{W}_{RB,AAC}(s)$	$\hat{\epsilon}^{W}_{RB,ACC}(s)$	$\hat{\epsilon}^{W}_{RB,CCC}(s)$	$\hat{\epsilon}^{W}_{RB,IIC}(s)$	Winner
$1^{*} = (1,\ 6,\ 7,\ 16)$	0.053	0.051	**0.035**	0.041	CCC
2	0.102	**0.089**	−0.113	−0.112	ACC
3	0.032	**0.031**	0.040	0.041	ACC
4	0.564	0.564	0.539	**0.480**	IIC
5	0.447	0.448	0.451	**0.436**	IIC
8	0.156	0.069	−0.010	**0.004**	IIC**
9^{*}= $(9,\ 13)$	0.093	0.093	**0.086**	0.090	CCC
10	0.461	0.469	**0.065**	0.079	CCC**
11	0.704	0.708	**0.428**	0.492	CCC**
$12^{*} = (12,\ 17)$	**0.008**	**0.008**	**0.008**	0.019	all but IIC
14	0.035	0.036	0.036	**0.034**	IIC
15	**0.012**	**0.012**	0.017	0.022	ACC
18	0.102	**−0.022**	−0.039	−0.038	ACC**
19	0.336	0.172	**0.043**	0.044	CCC**
20	**0.092**	**0.092**	**0.092**	0.093	all

* design replicates, results have been averaged over replicates.
** relevant benefit with respect to the absolute bias.

The design replicates have been previously grouped and identified by 1^{*}, 9^{*} and 12^{*}. In the rows corresponding to the design replicates Table 4.6 shows the relative bias achieved with each weighting method averaged over the replicates.

Although a single method wins the competition at most of the design points in terms of the absolute relative bias, the advantage of the winner upon the other methods is sometimes rather unimportant in terms of absolute bias. This is the case at the design points 1^{*}, 3, 4, 5, 9^{*}, 12^{*}, 14, 15 and 20.

Table 4.6 reveals two pairs of algorithms that provide similar results over all data configurations: AAC and ACC (blue pairs) as well as CCC and IIC (red pairs).

AAC and ACC work in a similar manner involving weights only on validation data sets. However, some small advantage of ACC upon AAC is also available, since this method influences also the final combination of features by weighting additionally in the phase of rule optimization. In Table 4.6 ACC provides the best results at three design points, namely 2, 3 and 18, while AAC never succeeds in outperforming all other methods. Nevertheless, the advantage of ACC over the methods that perform a thorough weighting during the learning process, CCC and IIC, is rather small at the design points 3 and 18.

The algorithm pair **CCC and IIC** wins in most of the design situations. The importance of their benefit upon the other pair of methods is confirmed also on the absolute bias scale at the design points 8, 10, 11 and 19.

Figure 4.2 illustrates the relative bias of the final weighted estimates of the misclassification error for each of the weighting methods. This is computed at each design point using (4.3.2). Blue lines are used for balanced target subclass structures, red lines for unbalanced subclass structures. At center points and design replicates the individual responses as well as their means over the replicates are depicted.

In Figure 4.2 the shapes of the red lines have an obvious distinctive pattern when compared to the blue lines. This suggests that, when the target population is highly unbalanced, which corresponds to a high degree of mismatch between the target population and the data at hand, both CCC and IIC outperform the other algorithms. Thus, the prevalence factor and therefore, the degree of suboptimality of the data at hand, affects clearly the result of our benchmark.

Figure 4.3 offers a useful overview of all factor settings of DOE1, including the first results of the benchmark of algorithms. This helps to understand also the relationship between the other design factors and the results of our benchmark.
It illustrates the geometrical configurations of the class and subclass means as well as the 95% contours of the class and subclass distributions projected on the first two feature coordinates. The winner method is specified in each plot to enable a visual exploration of possible connections between the performance of an algorithm and the target subclass structure.

Design points 8, 10 and 19, at which a clear benefit is achieved by application of IIC and CCC, are not only characterized by a high discrepancy between the subclass prevalences, but also by well separated subclasses (angles of 90^0 or 180^0 and at least a large distance factor). This corresponds to a pronounced heterogeneity of class C in the target population.

Besides, these design points illustrate a situation which is often encountered in the diagnostic practice. Typically, the control collective comprises a bunch of diseases which are interesting from a *differential diagnosis*-point of view. These diseases have a similar symptomatic with the disease of interest. They should help to identify a highly sensitive marker combination for a reliable diagnosis.
Some of these disease-subclasses of the control class may be easily distinguished from the disease of interest by means of a single marker while the rest of them may not. For instance, in a study for the early diagnosis of rheumatoid arthritis (RA), the control collective comprises beside a group of healthy controls also patients with osteoarthritis (OA). While the former can be successfully distinguished from RA cases by means of a specific marker for rheumatic diseases, the latter group is harder to separate, since both OA and RA are characterized by elevated concentration levels of this marker.

All algorithms result in the same poor performance at the design points 4 and 5. Here, the class and subclass centers are collinear and the subclass centers are positioned on different sides with respect to the center of the opposite class at equal distance from it.
A similar practical case is when a specific inflammatory marker is used in the context of RA diagnosis while the control collective comprises healthy controls and some typical inflammatory disease. Even if a highly unbalanced prevalence structure is given in the heterogeneous class, weighting alone cannot help to a performance improvement. In such cases weighting strategies applied to classification methods that account explicitly for the subclass distribution in the target population, like Mixture Discriminant Analysis (Hastie & Tibshirani, 1996) or local classifiers (Szepannek & Weihs, 2006), are expected to work better.

The set of design points related to similar results of all algorithms or to a slightly better performance of ACC can be described using three not necessarily disjoint attributes, deduced from Figure 4.3. These points are characterized either by equal subclass prevalences or by an angle of 0° (see point 2) or by a big overlap between class and subclass distributions (20, 15, 12^*, 18).

4.3.5.2 Linear Model DOE1

A further goal is the adjustment of DOE1 for a more thorough investigation of the relationship between the target data structure and the performance of the weighting algorithms.
Separate linear models including only main factor effects were fit to check for the linear impact of the design factors on the performance of each weighting algorithm, which is expressed in terms of relative bias. All factors had a significant effect on the performance of at least one algorithm. Thus, none of them was dropped out before switching over to the design augmentation

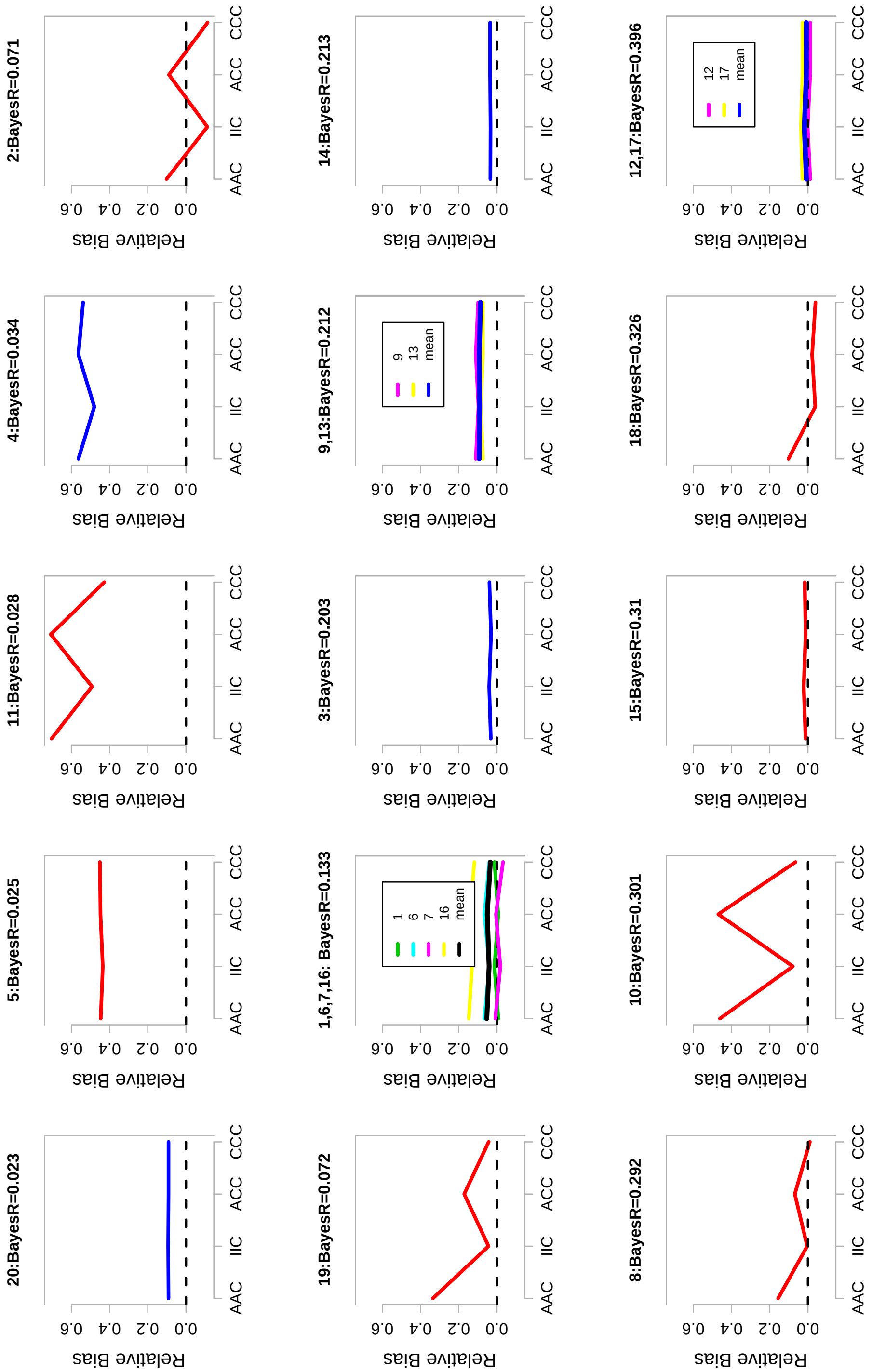


Figure 4.2: DOE1: Relative bias of the misclassification error rates over the weighting methods. *Above each graphic stand the no. of the design point and the corresponding Bayes rate. Blue lines stand for* $\Pi_{C_1,C_2} = (0.5, 0.5)$, *red lines for* $\Pi_{C_1,C_2} = (0.1, 0.9)$, *black lines for* $\Pi_{C_1,C_2} = (0.3, 0.7)$. *Design points have been placed by row in ascending order of the true error rates. The dashed black line stands for no bias.*

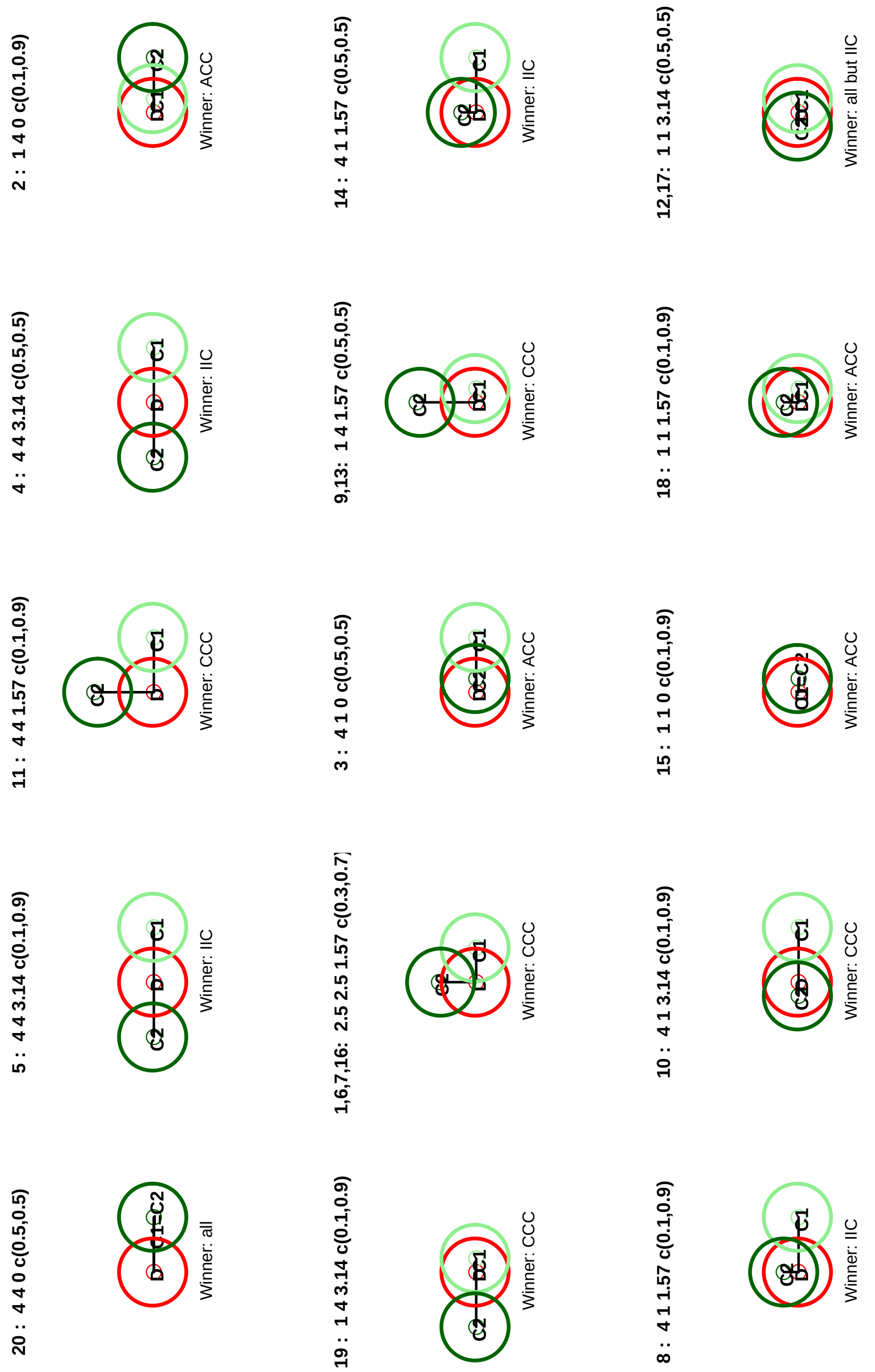


Figure 4.3: Geometrical illustration of the distributional configuration at each design point in DOE1. *Above each graphic the factor settings are specified as* ***no. design point: dist.DC$_1$, dist.DC$_2$, $\widehat{C_1DC_2}$, Π_{C_1,C_2}****. Circles represent the 95% contours for the class and subclass distributions in the target populations, projected on the first two coordinates; C_1 = dark green, C_2 = light green, D = red. Design points are listed by row in ascending order of the true error rates.*

step.

Also, the adjusted R-square measures of the four final models indicated a rather poor linear fit, the maximal values being achieved for the models associated to AAC and ACC (0.67 and 0.58 respectively).

Consequently, DOE1 is augmented in JMP7.0 to an appropriate design for prediction which includes the quadratic and two-factor interaction effects. This extension of the original model is expected to reduce the amount of unexplained response variation and enhance understanding of the model connections.

4.3.5.3 Graphical approach DOE2

The response values obtained with all algorithms for all design points of DOE2 are listed in Table 4.7 together with the winner method.

The same plots are used to explore the results of DOE2 as those used for the analysis of DOE1. Figures 4.4 and 4.5 enable a multidimensional overview of the results and of the factor settings.

Out of the 16 additional design configurations, 8 are characterized by a medium discrepancy between the subclass prevalences in the target population. Only 3 design points correspond to an unbalanced subclass structure and the rest of 5 design points are characterized by a balanced subclass structure in the target population (see Table 4.5).

Like in case of the design points 15 and 20 of DOE1, also at the design points 25, 32 and 33 of DOE2, the algorithms perform identically. All these points are characterized by completely overlapped subclasses. Thus, the problem reduces to a simple classification in the context of no heterogeneity in the data. Only IIC presents some negligible variations due to the practiced data resampling. These confirm however, the potential of IIC to provide reliable results over various data configurations.

The balanced situations, illustrated by the blue lines, are again characterized by no or only negligible variations in the performance of the algorithms.

The red lines, corresponding to a highly unbalanced subclass prevalence structure in the target population, exhibit the same pattern as in Figure 4.2, provided that subclasses do not overlap entirely. This means that both AAC and ACC are clearly outperformed by CCC and IIC.

Table 4.7: DOE2: Relative bias per design point (average over 50 MC loops). *Blue stands for similar results AAC and ACC, red for similar results of CCC and IIC.*

Design Point(s)	$\hat{\epsilon}^W_{RB,AAC}(s)$	$\hat{\epsilon}^W_{RB,ACC}(s)$	$\hat{\epsilon}^W_{RB,CCC}(s)$	$\hat{\epsilon}^W_{RB,IIC}(s)$	Winner
21	0.170	0.170	0.170	**0.159**	IIC
22	**0.068**	0.091	0.105	0.098	AAC
23	0.048	**0.047**	0.071	0.071	ACC
24	0.069	0.069	0.041	**0.064**	CCC
25	−0.003	−0.003	−0.003	**<0.001**	IIC
26	0.204	0.204	**0.028**	0.038	CCC**
27	0.222	**0.209**	0.240	0.250	ACC
28	0.102	0.102	0.102	0.126	all but IIC
29	> **−10e−4**	> −10e−3	> −10e−3	−0.002	AAC
30	0.077	0.057	0.004	**0.003**	IIC
31	0.027	**0.025**	0.044	0.046	ACC
32	−0.099	−0.098	−0.098	−0.102	all
33	−0.165	−0.164	−0.165	−0.161	all
34	0.109	0.109	**0.047**	0.057	CCC**
35	<0.001	>−0.001	>−0.001	<0.001	AAC, IIC
36	−0.010	**−0.008**	−0.026	−0.027	ACC

** relevant benefit with respect to the absolute bias.

4.3.5.4 CART model DOE1 $\bigcup$ DOE2

Design points are divided into three categories:

(1) design points at which the pair CCC and IIC provides a visible advantage upon the other pair of algorithms in terms of relative bias (1^*, 8, 10, 11, 19, 24, 26, 30 and 34);

(2) design points at which the pair of algorithms CCC and IIC is visibly outperformed by the pair of algorithms AAC and ACC in terms of relative bias (2, 18, 22, 23, 27, 31 and 36);

(3) design points at which all algorithms perform very similarly (3, 4, 5, 9^*, 12^*, 14, 15, 20, 21, 25, 28, 29, 32, 33 and 35).

Appropriate characteristics of the data are searched that enable a good discrimination between the previously defined categories. These data characteristics, so far they are known to the user

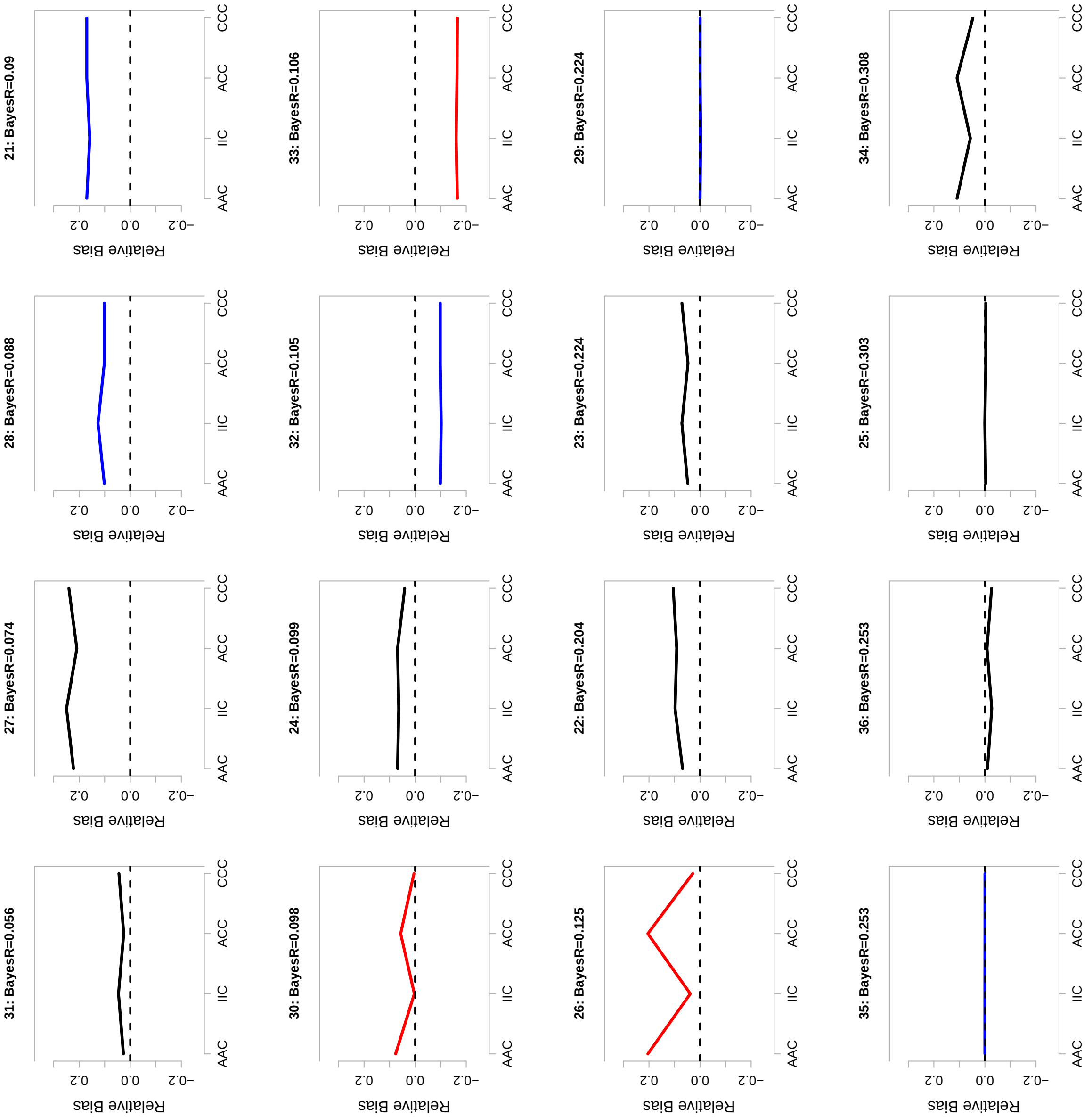


Figure 4.4: DOE2: Relative bias of the misclassification error rates over the weighting methods. *Above each graphic stand the no. of the design point and the corresponding Bayes rate. Blue lines stand for* $\Pi_{C_1,C_2} = (0.5, 0.5)$*, red lines for* $\Pi_{C_1,C_2} = (0.1, 0.9)$*, black lines for* $\Pi_{C_1,C_2} = (0.3, 0.7)$*. Design points have been sorted by row in ascending order of the true error rates. The dashed black line stands for no bias.*

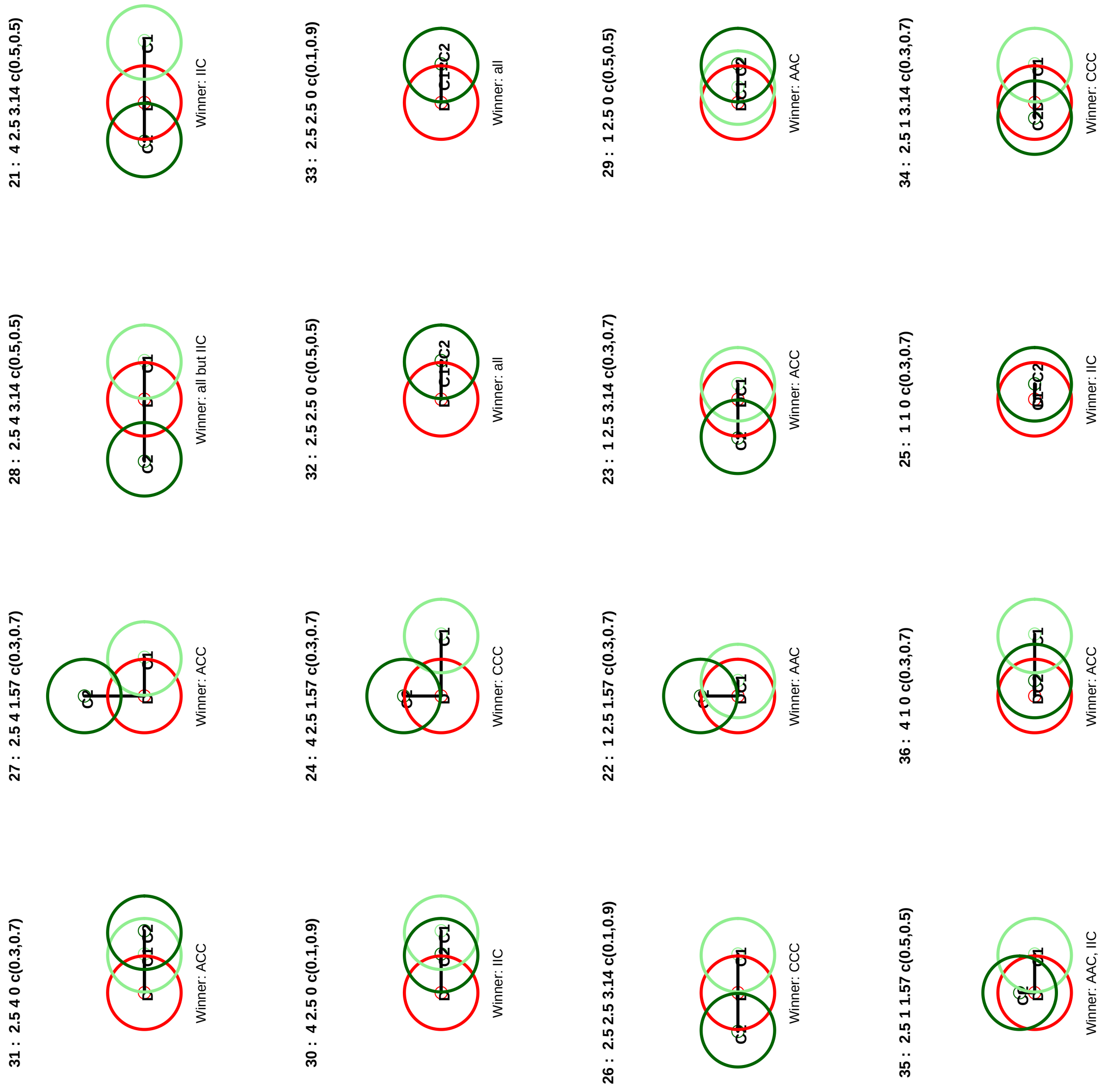


Figure 4.5: Geometrical illustration of the distributional configuration at each additional design point in DOE2. *Above each graphic the factor settings are specified as* ***no. design point: dist.DC$_1$, dist.DC$_2$, $\widehat{C_1DC_2}$, Π_{C_1,C_2}****. Circles represent the* 95% *contours for the class and subclass distributions in the target populations, projected on the first two coordinates; C_1 =dark green, C_2 =light green, D =red. Design points are listed by row in ascending order of the true error rates.*

or at least inferable by some simulations, can be used to predict which algorithm would suit best given a certain distributional configuration of the target population.

After a visual inspection of Figures 4.3 and 4.5, the following features seem to provide a meaningful description of the first category when compared to the rest:

(a) the level of discrepancy between the true subclass prevalences, here also interpretable as the degree of suboptimality of the data at hand;

(b) the angle formed by the subclass centers (μ_{C_1}, μ_{C_2}) with the center of the opposite class (μ_D);

(c) the discrepancy between the subclass degrees of overlap with the opposite class (large for a low-high or high-low setting of the euclidian distances).

A simple CART model is used to assess the category of each design point by means of and-or combinations of conditions on the features described at (a)-(c).

The CART rule claims that, given a high discrepancy between the true subclass prevalences, moderately to highly discrepant degrees of overlap between subclasses (C_1 and C_2) and the opposite class (D), an application of **IIC or CCC** is appropriate. This rule applies correctly to 80% of the cases which met these criteria.

Also, given a medium to high discrepancy between the true subclass prevalences, similar degrees of overlap between subclasses (C_1 and C_2) and the opposite class (D), but an angle of 90° or 180°, the pair of algorithms **IIC or CCC** is selected by the CART method leading to 60% correct assignments.

However, if a medium discrepancy between the true subclass prevalences and moderately to highly discrepant subclass overlaps are given, then the rule decides correctly in 71% of the cases for an application of **AAC or ACC**.

The third category of design points can easily be distinguished from the other categories. It is clearly defined either through a balanced subclass structure in the target population or perfectly overlapped subclasses (i.e. equal overlaps of the subclasses with the opposite class and an angle of 0°).

Only five design points are misclassified by means of the CART rule. Out of them, two are from the first category (design points 24 and 34), two from the second (design points 2 and 18) and one from the third category (design point 5).

The misclassification of design point 18 is easy to explain, since in this case AAC is outperformed by all other weighting methods, inclusively by ACC.

At design point 2, the difference between methods is actually irrelevant in terms of absolute relative bias. AAC and ACC are still preferred since, in contrast to IIC and CCC, they are not negatively (i.e. optimistically) biased.

At point 5, due to the pronounced subclass structure with subclasses being separated by the opposite class, none of the methods is able to perform better than the others.

Also, the misclassifications of design points 24 and 34 are plausible, since at the former point CCC singly outperforms all other methods and at the latter point no relevant difference is available in terms of absolute bias.

Thus, all these misclassifications are caused because of the existence of more than three categories among our design points.

4.3.5.5 Quadratic model DOE1 $\bigcup$ DOE2

The augmented design DOE1 $\bigcup$ DOE2 included only 31 out of 81 possible combinations of the low-medium-high factor levels, namely 15 from DOE1 and 16 from DOE2. A suitable quadratic model is fitted to the results obtained with each weighting algorithm to predict its performance given every possible factor setting.

In addition to the explorative investigation of the connections between design configuration and results which was practiced so far, the quadratic models help to provide an information about how does the performance of each algorithm vary by tuning single design factors or combinations of them.

First, each model included all main, quadratic as well as two-factor interaction effects. All factor effects that were significant in at least one of the four models were pooled to a model which we further call the *smallest common denominator (SCD)*-model.

Our SCD-model has no intercept and includes nine parameters, which are the main factor effects, the quadratic effects of the distance factors and the two-factor interactions distance-distance and distance-angle.

The classical and adjusted R-square measures indicate a good linear fit in the selected parameters which explain around 80% of the response variability in all SCD-models. These measures amount to 0.86 and 0.81 for the model associated with AAC, to 0.83 and 0.77 for the ACC model, to 0.86 and 0.80 for the CCC model and to 0.85 and 0.79 for the IIC model, respectively.

Coefficients and p-values corresponding to all effects in the SCD-model are listed in Table 4.8 for each weighting algorithm. All significance tests are carried out at level 0.05.

The SCD-models are used for an extended comparison of the algorithms. Based on them, predictions are obtained for all 81 different factor settings. For each combination, representing a new observation in the space of all target data configurations, an individual prediction is computed per algorithm starting from the estimated model coefficients.

The model corresponding to AAC includes the prevalence factor with a significantly non-zero coefficient (p-value 0.01).
Although in the model associated to ACC the prevalence factor is not statistically significant, the p-value of 0.079, based on the rather small amount of experimental data, indicates that the impact of this factor on the performance estimate is not completely abstracted by the weighting strategy practiced by this method.
Both AAC and ACC perform a kind of post-weighting, which does not seem to suffice for discarding the effect of a sub-optimal subclass representation in the data at hand.

Table 4.8: $DOE1 \bigcup DOE2$: Comparative view of coefficients and significance of predictors for the relative bias.

Factors	$\hat{\beta}_{AAC}$	p-value$_{AAC}$	$\hat{\beta}_{ACC}$	p-value$_{ACC}$	$\hat{\beta}_{CCC}$	p-value$_{CCC}$	$\hat{\beta}_{IIC}$	p-value$_{IIC}$
Π_{C_1,C_2}	0.061	0.010*	0.042	0.079	−0.014	0.388	−0.008	0.654
dist.DC$_1$	0.078	0.002*	0.090	0.001*	0.068	0.000*	0.069	0.000*
dist.DC$_2$	0.084	0.001*	0.082	0.002*	0.077	0.000*	0.076	0.000*
$\widehat{C_1DC_2}$	0.103	0.000*	0.095	0.001*	0.070	0.000*	0.068	0.001*
dist.DC$_1^2$	0.094	0.008*	0.087	0.019*	0.053	0.041*	0.051	0.054
dist.DC$_2^2$	0.094	0.009*	0.079	0.034*	0.048	0.063	0.055	0.040*
dist.DC$_1$: dist.DC$_2$	0.029	0.266	0.038	0.186	0.072	0.001*	0.072	0.002*
dist.DC$_1$: $\widehat{C_1DC_2}$	0.044	0.112	0.060	0.045*	0.032	0.122	0.027	0.210
dist.DC$_2$: $\widehat{C_1DC_2}$	0.004	0.873	−0.006	0.837	0.042	0.053	0.036	0.097

* significant at level $\alpha = 0.05$.

Besides, the prevalence effects estimated in the models of AAC and ACC indicate that, enlarging the difference in the subclass prevalences from the low (0) to the high level (0.8) causes an increase of the relative bias by about 12.2 percentage points in AAC and 8.4 percentage points in ACC. Consequently, the higher the true error rate, the stronger the bias associated to AAC and ACC due to the suboptimality of the data at hand.

The models of CCC and IIC show a negligible prevalence effect (p-values 0.39 and 0.65 respectively). This result is plausible, since both methods fully exhaust the influence of true subclass prevalences using them both in the phase of rule building and optimization. Therefore, IIC and CCC are almost unsensitive regarding the tuning operation from low to high discrepancy in the true subclass prevalences. They are expected to provide a real advantage upon the other two methods in the context of a highly suboptimal data at hand.

The interaction between the distance factors (dist.DC_1 : dist.DC_2) has a significant impact on the relative performance of IIC and CCC. AAC and ACC remain rather unaffected with respect to this model term. Consequently, a greater benefit by application of IIC and CCC can be especially expected given a low-high relationship between the distance factors. This means, equivalently, a high discrepancy between the subclass degrees of overlap with the opposite class. For instance, given an angle of 90°, a simultaneous tuning of the distance factors from a low-high to a high-high relationship would increase the predicted relative bias of IIC and CCC by about 28.0 percentage points.

Most of the findings based on the absolute predicted values confirm the rule developed by CART which used conclusions drawn from Figures 4.2, 4.3, 4.4 and 4.5.

4.3.5.6 Negative bias discussion

We note some optimistic tendency of the intensively weighting methods ACC, CCC and IIC. They provide under some design configurations negatively biased estimates of the misclassification rate. This is for example the case at design points 2, 8, 18 and 35.

At design point 2 only IIC and CCC result in a negative relative bias. However, this is rather negligible in terms of absolute bias.

At design point 8, only CCC provides a negatively biased result, while the positive, but similar and nearly zero value obtained with IIC can be traced back on the inflation procedure. The absolute bias of both methods is however negligible, indicating a very good performance.

At the design points 18 and 35 AAC provides a positive result. The rather small negative bias achieved by ACC, CCC and IIC (IIC only at design point 18) at the same design points can be traced back on the common property of them to weight during the rule optimization (i.e. on the CV test data sets). This property is common to all algorithms, but AAC. Normally, IIC is also expected to return a negative result at design point 35. However, it results in a positive bias due to its inflation procedure.

The negatively biased results of all methods at the design points 25 (except for IIC), 26, 29, 32, 33 and 36 are explainable by the particular choice of the data sets.

4.3.5.7 Summary

Two pairs of similarly performing algorithms were identified: (1) AAC and ACC and (2) IIC and CCC.

Given a medium or high prevalence factor (i.e. a medium or large gap between the true and observed subclass prevalences), the methods IIC and CCC provide almost always the best results. A higher benefit is achieved especially if the angle factor is at least 90° and an euclidian distance is large while the other one is small. In all these cases the study population is non-representative with respect to the subclass prevalence structure and the target population is characterized by well separated subclasses, thus, a rather pronounced heterogeneous profile.

A similar performance of all algorithms or a slightly better performance of AAC and ACC is associated either with a balanced target subclass structure or with highly overlapped class and subclasses or with an angle of 0°. In such situations just a small benefit, if any, can be expected from IIC and CCC.
However, it is unclear in which situation the application of AAC or ACC should be preferred. So far, it can just be recommended to proceed with care if the subclass centers are positioned collinearly on the same side with respect to the center of the other class. AAC is rather unrecommended, since it results with very few exceptions if not in a similar then in a poorer predicted performance than ACC.

A comparison of the absolute predicted values of the relative bias over the algorithms confirmed essentially the pair of algorithms CCC and IIC as winner over almost all possible combinations of factor levels. We recommend the use of IIC and CCC over all target situations, in order to be always on the safe side. These algorithms provide constantly reliable results. Also when they provide no improvement, they still cause no harm.

The negligible impact of the true subclass prevalences on the final results of IIC and CCC confirmed essentially that these two methods proceed in the right way. By their weighting strategies they eliminate exactly the bias introduced by the mismatch between the true and the observed subclass prevalences.

4.4 Benchmark of weighting algorithms on real data

4.4.1 Data description

The weighting algorithms introduced in Section 4.3.1 are applied to a real data example provided by Roche Diagnostics GmbH. Within a diagnostic study for the early identification of rheumatoid arthritis (RA) 794 patients were recruited in five European centers at general practitioners's office (GP-data).
The GP-data is suitable for a practical illustration of the application of weighting algorithms, since prevalences are available for the disease conditions which appear in an ideal GP-collective

of non-RA patients. Besides, the population obtained from the general practitioner's office is likely to comply with the request of asymptomatic subjects for the screening purpose.

One of the targets is to provide a panel of screening markers and a rule based on it, for the identification of rheumatoid arthritis (RA) at an early stage, when the treatment is more effective and enables the prevention from the irreversible destruction of the joints.

Concentrations of 6 biomarkers, measured in serum samples, are available for building the diagnostic rule. The biomarkers are encoded as M_2, M_4, M_5, M_9, M_{11} and M_{12}. Their values are previously transformed on the decimal logarithmic scale to approach normality, which is desirable in RDA.
The true disease status is provided by the ACR criteria (Arnett *et al.*, 1988). These represent the established *gold standard* for RA.

Among the GP-data, 364 subjects are diagnosed as RA-positive. The collective formed by the remaining 430 comprises patients with some other critical disease conditions of similar symptomatic with RA. These are grouped into four subclasses, further denoted as C_1 to C_4, for which the prevalences in an ideal GP-population is known. This data offers an example of the stratification model (iii) described by Sukhatme & Beam (1994) and mentioned in Section 4.1.

Now, RA itself is supposed to be detected in about 10% of the samples from an ideal screening population. However, high costs in terms of study budget as well as logistic and temporal limitations hampered the collection of a representative GP-population for screening. Available cases and controls were matched from different GP offices and centers in order to achieve reasonable sample sizes as well as the known configuration of the GP panel. The data collection ended up in an improper population, which does not mirror the true prevalences of different disease groups likely to be encountered by a general practitioner. The proportions of subclasses C_3 and C_4 in the control class do not resemble the proportions that would be theoretically found within the target GP population. In Table 4.9 the observed and respectively true proportions of these subclasses within the control class are given.

For both classes, *RA* and controls, shortly denoted as D and C, respectively, equal priors are assumed instead of their observed proportions in the data. This assumption is made to justify the use of the misclassification rate as objective function throughout this work. Under restrictions like equal class sizes and equal costs for misclassification of diseased and non-diseased patients, the RDA rule does not depend on the class priors and costs. Thus, it is equivalent to the rule based on the likelihood ratio function $LR(\boldsymbol{x}) = \frac{P(\boldsymbol{x}|y=1)}{P(\boldsymbol{x}|y=0)}$ when the cutoff point is 1. This rule has the best possible ROC curve among all possible functions of $\boldsymbol{x}$ that minimize the misclassification rate (Pepe, 2003). Consequently, under this assumption, the minimal misclassification rate is associated to an RDA rule with maximal AUC.

Table 4.9: GP-panel of RA-negatives: True and observed subclass prevalences (%)

Prevalence	C_1	C_2	C_3	C_4
observed ($\hat{\pi}_{ik}$)	5.56	4.86	24.30	65.18
true(π_{ik})	3.08	5.46	47.10	44.36

4.4.2 Comparative study on real data

The four weighting algorithms proposed in Table 4.2, are run on the GP-data. An MCCV design with 50 outer MC and 5 inner CV loops is used for the benchmark of algorithms. The two-steps-procedure of finding the optimal rule on the MC training data set and evaluating its performance on the corresponding MC test data set was repeated 50 times with different MC *training : test* splits of the original data in a proportion of 2 : 1.

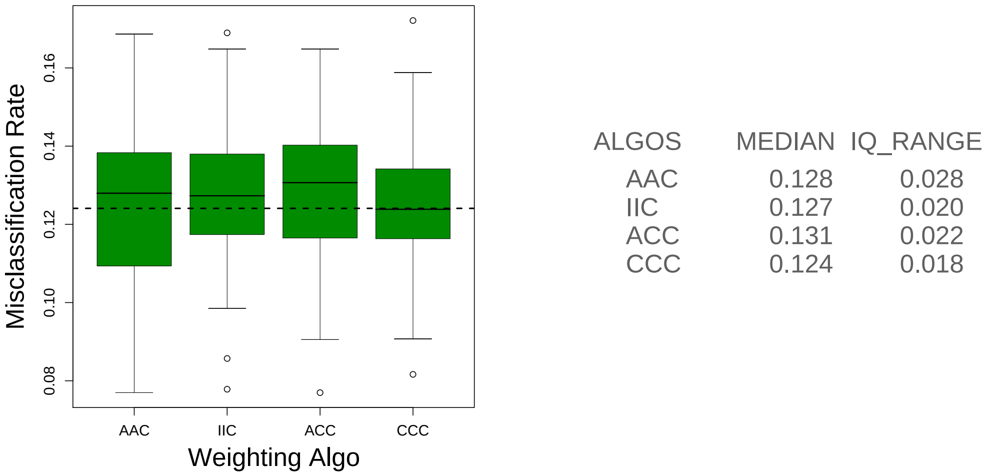
Parallel boxplots of the misclassification rates obtained with the different algorithms on the 50 MC test data sets are shown in Figure 4.6. On this particular task, all weighting algorithms perform similarly, providing median misclassification rates between 12% and 13%. The rather negligible gap between the median CV estimates of the misclassification rate and the median test estimates obtained over the 50 MC loops (right from Figure 4.6) confirm a realistic rule choice by all methods (AAC 12.0% vs. 12.8%, ACC 12.3% vs. 13.1%, IIC 11.7% vs. 12.7%, CCC 12.0% vs. 12.3%). However, it is noticeable that the minimal gap is achieved by CCC, which indicates the greater reliability of this method.

The differences between algorithms regarding the variability of their results should be remarked, since they can be attributed more to the working principle of an algorithm than to some distributional particularities of the data at hand. In spite of its smaller average misclassification rate, AAC provides more unstable results in comparison to the other algorithms, like the greater variability of its MC misclassification rates indicates (see also the IQR-estimates in Figure 4.6).

The best algorithm on this task, if regarded both from the perspective of its median test performance as well as of its stability is CCC.

Although IIC works in a very similar manner to CCC, its performance is here slightly inferior.

IIC is presumably a bit affected in its optimization process by the very small sample sizes of subclasses C_1 and C_2 in the CV test data sets (two and respectively three observations). The problem of redundant information in these very small subclasses might be also related to the performance loss of IIC.



ALGOS	MEDIAN	IQ_RANGE
AAC	0.128	0.028
IIC	0.127	0.020
ACC	0.131	0.022
CCC	0.124	0.018

Figure 4.6: Rheumatoid arthritis data: Boxplots of the misclassification error rates over 50 MC loops. *The dashed line corresponds to the mean of AAC, here taken as reference; right from the plot medians and inter-quartile-ranges have been specified.*

The regularization parameters obtained for each weighting method over the 50 MC loops as well as the composition of the optimal marker panels are monitored. Means and medians of λ and γ estimates as well as their variability in terms of standard deviation and interquartile range are listed in Table 4.10.

For all weighting algorithms, the median regularization parameters are 0. This corresponds to the quadratic discriminant analysis (QDA). While AAC tends to perform no regularization, the larger γ estimates and interquartile ranges obtained for ACC, IIC and CCC, show that these methods do not always agree upon the use of a QDA rule given the target data distribution. In

Table 4.10: Average results on the GP-Data

Method	**Mean**		**Median**		**StdDev**		**IQR**	
	λ	γ	λ	γ	λ	γ	λ	γ
AAC	0	0.12	0	0	0	0.29	0	0
IIC	0	0.37	0	0	0	0.45	0	1
ACC	0	0.32	0	0	0	0.44	0	1
CCC	0	0.28	0	0	0	0.42	0	0.5

contrast to AAC, these methods take into account the subclass prevalence structure at least in the phase of the rule optimization. Thus, they adapt also the rule parameters and features with respect to the target subclass configuration.

Figure 4.7 illustrates the selected biomarker combinations and their absolute frequencies in 50 MC loops for each weighting algorithm.
The combination rule based on M_9 and M_{11} obtained the majority of votes with all algorithms. This is selected in 58% of the MC iterations with AAC, in 50% with CCC, in 36% with ACC and in 32% with IIC. However, the methods ACC, CCC and IIC decide much more often than AAC for the single marker model based on M_9. Thus, they signalize that a single marker model may suffice for a reliable diagnosis of RA on the target population.

Three-marker combinations are more rarely picked up by the IIC method when compared to the similarly working procedure CCC. This suggests that the extra bias in the parameter estimates expected in the context of rare subclasses (insufficient data coupled with redundant information) may lead to biased feature combinations.

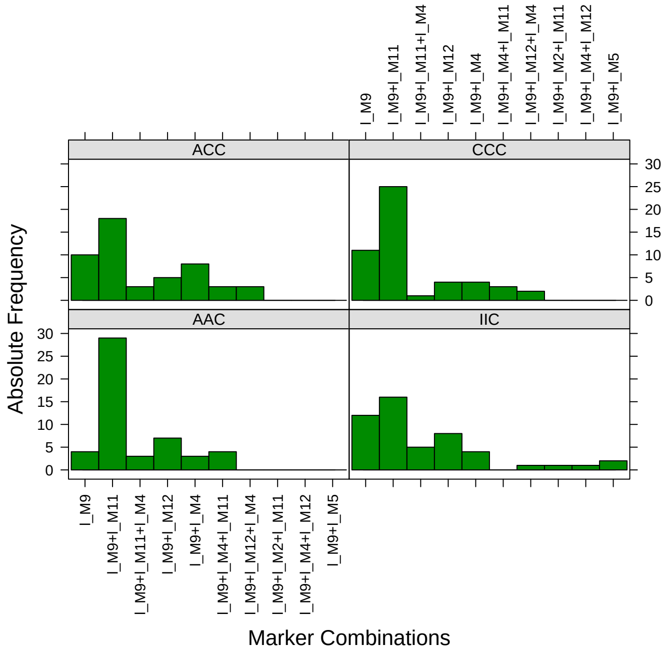


Figure 4.7: Rheumatoid arthritis data: Multiplicity plots-*absolute frequencies of the optimal biomarker combinations selected in* 50 *MC loops per weighting algorithm.*

4.5 Conclusions

Throughout this chapter the problem of building diagnostic rules when the target population is subject to heterogeneity is addressed. It is assumed that the subclass structure and the true subclass prevalences in the target population are known.
Usually, the data set used for learning the classification rule, called study population or data at hand, is just a small excerpt from the target population. Based on it, valid and generalizable diagnostic rules should be provided. However, depending on the way in which the data are collected, this data set might offer a suboptimal picture of the subclass structure in the target population. This is the case, for instance, when the observed subclass prevalences in the data at hand do not resemble the true subclass prevalences in the target population.

A good practical example for this non-representativeness problem is given by the screening situation. Since a prospective study design, which is recommended in this case, is not always financially and logistically feasible, the data may be collected in the frame of a retrospective case-control study. Therefore, in the resulting study population subclass proportions can be strongly biased.

Using the suboptimal data set for learning can affect the diagnostic rules in two different ways, especially if the gap between the observed and true prevalences is large for some subclasses: (1) the rule might not be valid, resulting in a suboptimal panel of features and/or suboptimal parameter estimates (2) the rule might not be generalizable, thus its estimated performance might be strongly biased.

In this context, the statistical literature up to date proposes some approaches based on a *post-weighting* of the classification rules. The interested reader is referred to (Sukhatme & Beam, 1994) and (Obuchowski & Zhou, 2002). They take the true subclass prevalences first in the phase of rule validation into account, when they adjust the estimates of classification performance for this information.

We propose another way to tackle the problem of non-representativeness of the data at hand. The idea is to account for the true subclass prevalence information not only in the phase of rule validation, but already in the phase of rule construction and optimization. Four weighting algorithms, AAC, IIC, ACC and CCC, which embed the true subclass prevalences as weighs at different stages of a feature selection algorithm are proposed. They are tailored for the use with RDA (Regularized Discriminant Analysis), but their weighting principles are adaptable also to other discriminant analysis approaches.

The algorithm AAC performs a post-weighting of the misclassification rate by means of the true subclass prevalences, thus it weights the estimates of misclassification error only on the validation data set. This method corresponds to the classical *post-weighting* procedure.
The algorithm ACC takes into account the target subclass prevalence structure already in the process of rule optimization, performing a post-weighting on the cross-validation test data sets.

Algorithms IIC and CCC account for the true subclass prevalences both in the process of rule optimization as well as in the process of rule building by different weighting strategies.
The algorithm CCC applies the true subclass prevalences to compute weighted estimates of the class distribution parameters on the CV training data sets and of the misclassification rates on the CV test data sets.
IIC enforces on the CV training and test data sets the expected subclass sizes according to the true subclass prevalences by randomized and stratified sampling with or without replacement from the existent subclass collectives.

The procedure of post-weighting on the validation data set is common to all algorithms.

Since their theoretical description does not allow for a direct comparison with respect to the classification performance, the benchmark of the four selected weighting algorithms is based on a simulation design.

The first class D (e.g. disease class) is assumed to be homogenous, while the second class C (e.g. control class) is assumed to be heterogeneous with two subclasses, C_1 and C_2. Four features are available, two informative and the other two non-informative regarding both the class and the subclass discrimination.

The subclass prevalences in the data at hand are assumed to be equal. Thus, the mismatch between the study and the target population depends on the discrepancy between the true subclass prevalences.

Four factors are suspected to affect the performance of the weighting algorithms. These are the absolute difference between the true subclass prevalences, the euclidian distances from subclass centers to the center of the opposite class D and the angle formed by subclass centers with the center of the opposite class.

The question of interest is if and how the design factors impact on the benefit of weighting with one of the proposed methods. The design factors are used to describe beneficial conditions for the application of each pair of algorithms as well as situations where no particular algorithm results in a superior performance.

Therefore, these factors are taken into account in the comparison of algorithms. The simulation design is generated as a design of experiments, in which each point corresponds to a combination of the factor levels.

Two pairs of algorithms with similar impact of their weighting strategies on the relative bias of the error estimates are identified: $\{AAC, ACC\}$ and $\{IIC, CCC\}$. The latter pair outperforms the former one in terms of absolute relative bias over the majority of design points. In the few cases, in which ACC or AAC outperform the other algorithms, their advantage is rather unimportant.

The explorative search after connections between the benefit of weighting and the design configuration by graphical means or by a simple CART rule lead to similar conclusions.

A performance gain of IIC or CCC is associated to a high or medium discrepancy between the true subclass prevalences in the target population. This is the case of a high or moderate non-representativeness of the data at hand with respect to the true subclass prevalence structure, since the observed subclass prevalences are here assumed to be equal. However, the prevalence condition is necessary, but not sufficient for a relevant performance improvement. A stronger benefit by IIC and CCC is traced back on an angle of at least 90° and highly discrepant overlaps between subclasses C_1 and C_2 and the homogeneous class D.

The design points which result in similar performances of all algorithms are associated either with a balanced true subclass prevalence structure, thus the representativeness of the data at hand, or with perfectly overlapped subclasses. The latter situation corresponds to an angle of 0° and equal degrees of overlap between subclasses and the opposite class.

By graphical means, it is additionally remarked that if, given a non-balanced target subclass structure, the algorithm pair {AAC, ACC} is at least as good as {IIC, CCC}, this may be caused by hardly separable distributions of C_1, C_2 and D.

Linear models are used to assess the importance of the design factors with respect to the weighting benefit. They are useful also to understand how the factors impact on the performance of the algorithms by tuning single or combinations of design factors.
The performance of the algorithms AAC and ACC is strongly influenced by the prevalence factor and therefore, by the non-representativeness of the data at hand. The effect of this factor is completely neutralized by the weighting strategies of IIC and CCC. Therefore, the prevalence information represents already a good criterion for selecting the proper weighting algorithm. A high discrepancy between the true and observed subclass prevalences gives a strong indication for the application of CCC or IIC. This is a comfortable remark, since the prevalence is usually the only information available about the target population.

The distance and angle factors, which offer a meaningful description of the heterogeneous structure in the target population, have a relevant effect on the performance of all weighting algorithms. However, in case of IIC and CCC this is still smaller than in case of AAC and ACC, since the former two algorithms target not only a recovery of the original prevalence structure, but also of the original class distributions. Also from this point of view, they are preferred to the latter two algorithms.
The interaction between the distance factors is clearly relevant in the models associated to IIC and CCC. Given a high level of the prevalence factor, thus a large mismatch between the true and observed subclass prevalences, a considerable reduction of the relative bias of these two methods may be achieved especially when one subclass has a small and the other subclass a high overlap with the homogeneous class (a distance factor is on the high and the other one is on the low level, respectively).

An analysis of the distance-angle interaction effects indicates another three cases in which IIC and CCC are expected to perform better than AAC and ACC:

(1) when the angle is 0° and there is a small overlap between the preponderant target subclass and the homogeneous class and a large overlap between the less prevalent subclass and

the homogenous class (i.e. dist.DC_2 is on the high and dist.DC_1 is on the low level, respectively);

(2) when the angle is 90° and one of the subclasses is much more overlapped with the homogenous class than the other one (distance factors are on different low-high levels);

(3) when the angle is 180° and there is a large overlap between the preponderant target subclass and the homogeneous class (equivalently dist.DC_2 is on the low level).

These characteristics of the distributions in the target population are not available. But, if the class and subclass sizes in the data at hand, assuming this was carefully collected, are reasonable, then the necessitated distance and angle values can be well and easily estimated and used to give a hint about the expected amount of benefit if weighting by means of IIC and CCC.

The performance of IIC or CCC observed by graphical methods or predicted for a level combination of the design factors is, if not better, then at least comparable to the observed or predicted performance of AAC or ACC. In many cases, especially under highly unbalanced true subclass prevalences, or equivalently, a large mismatch between the true and observed subclass prevalences, the application of these algorithms provides a real benefit.
Between CCC and IIC no relevant differences can be established. However, IIC has an additional uncontrollable variability due to its inflation procedure. Therefore, CCC with its straightforward computational workflow is recommended in all situations in which the study population is suboptimal. Its application guarantees, if not always a high benefit, then at least safe results in case of data non-representativeness.

The algorithms are verified also on the real data from a rheumatoid arthritis screening study. They perform on this task rather similar, however a careful analysis of the MC results reveals some potentially important differences.
The worst performance in terms of the average MC test estimate of the misclassification error is achieved with ACC. The best performance in terms of median MC test estimates of the misclassification error as well as regarding the stability of the results can be attributed to CCC. The method AAC achieves in this context by far the highest instability, which makes it undesirable in spite of its median performance which is comparable with that obtained by CCC.

Both IIC and CCC show on the real data a greater stability in the classification results than the other pair of algorithms and are so far clearly confirmed also by the practical results.

Chapter 5

Validating diagnostic rules in case-control studies

The problem of non-representativeness or when becomes prevalence weighting of classification errors too dangerous?

The diagnostic rule is usually learned on a small excerpt of the target population which represents the study population or the data at hand. The observed subclass prevalences may be very different from the true subclass prevalences. Not taking into account the true subclass prevalence structure can cause strongly biased estimates of the misclassification error on the one hand, and lead to wrong diagnostic rules on the other hand.

The simulation study performed in Chapter 4 proves that weighting by means of the true subclass prevalences can help to reduce the bias in the misclassification rates. Methods which use weighted distribution parameters and error estimates of the heterogeneous class already in the rule building and optimization process provide over the simulations at least as good results as those which accounted for the true subclass prevalence structure only in the validation phase.

In this chapter we focus on the theoretical survey of the statistical properties, like bias and variance of weighted estimates, aiming to get a sustained insight into their domain of applicability. This theoretical research is also motivated by the implacable bias-variance trade-off. By estimating the error rate in a heterogeneous class as weighted sum of the subclass error rates, the variance may increase due to the contribution of up-weighted subclasses.

In Section 5.1 some definitions and notations are introduced to help understanding the coming theoretical discussion. In Section 5.2 the benefit of weighted error estimates with respect to the unweighted ones is assessed by a comparison between their mean squared errors. In Section 5.3 the impact of weighting on estimates of the distribution parameters is investigated. Section 5.4 presents a discussion of the theoretical results and filters out the most advantageous situations for the use of weighted estimates.

5.1 Background and notations

The case of a homogeneous disease class and of a heterogeneous control class with two subclasses is considered. This simple situation enables a first insight into the theoretical properties of weighted estimates which may explain some findings of the simulation studies performed in Chapter 4. Classes are identified again by 1(=disease) and 2(=control) or labeled as D and C, respectively, while the control subclasses are named C_1 and C_2.

Like in Chapter 4, class and subclass means in the target population are denoted by $(\boldsymbol{\mu}_i)_{i=1,2}$, and $(\boldsymbol{\mu}_{2,k})_{k=1,2}$, while $\Sigma_i,\ i = 1, 2$, and $\Sigma_{2,k},\ k = 1, 2$, stand for target class and subclass covariance matrices, respectively. The corresponding estimates are distinguished using the hat-sign.
The true subclass prevalences, thus the subclass probabilities in the target control population, and the observed subclass prevalences, thus the subclass proportions within the data at hand, are $\pi_{2,k},\ k = 1, 2$, and $\hat{\pi}_{2,k}$, respectively.
$N_i,\ i = 1, 2$, and $N_{2,k},\ k = 1, 2$, stand for class and subclass sizes, either in the training or in the validation data depending on the context. The relationship between class and subclass sample sizes is approximately the same in training, validation, CV training and CV test data sets due to stratified random sampling.

Table 5.1 offers an easily manageable summary of new notations for important quantities which appear in the coming discussion around the existence of a weighting benefit.

The quotients of the subclass prevalences in the target population and in the data at hand are denoted as f_t and f_d, respectively. They measure the degree of unbalance in the target and data at hand, respectively.
The absolute relative discrepancy between these two quotients is $|\rho^*|$, which is used to quantify the degree of mismatch between the true and observed subclass structures.
The difference u_π of the true subclass prevalences represents the prevalence factor used in the simulation study from Chapter 4. Given a balanced subclass prevalence structure in the study population, i.e. $f_d = 1$, a high value of $f_t > 1$ is equivalent to a high value of u_π and corresponds to a high value of $|\rho^*|$, which is then close to 1.

Notation	Domain	Definition
${\pi_{2k}}^{[a]}$	$(0,1)$	true prevalence of subclass $C_k,\ k = 1,2$ within class C
$\hat{\pi}_{2k}$	$(0,1)$	observed prevalence of subclass $C_k,\ k = 1,2$ within class C
u_π	$[0,1)$	absolute difference between the subclass prevalences within class C
f_t	$[1,\infty)$	quotient of the true subclass prevalences
f_d	$(0,f_t)$	quotient of the observed subclass prevalences
ρ^*	$(-1,0)$	relative difference between f_d and f_t with respect to f_t
${p_{2k}}^{[b]}$	$[0,1]$	error probability in subclass C_k
p_2	$[0,1]$	error probability in class C
u_p	$[0,1]$	absolute difference between the subclass error probabilities
r	$\mathbb{R}^+(rp_{21} = p_{22} \leq 1)$	quotient between the subclass error probabilities of the subclasses C_2 and C_1
σ_k	$[0,0.25]$	variance of the indicator variable for the misclassification event in subclass $C_k,\ k = 1,2$
v_k	$[0,(4N_{2k})^{-1}]$	variance of the misclassification rate within subclass C_k
s	$\mathbb{R}^+$	quotient of σ_2 and σ_1

[a] =known subclass prevalences

[b] =unknown subclass error probabilities (to be estimated)

Table 5.1: Notations for important theoretical factors and associated interpretations.

According to Table 5.1:

$$\begin{aligned} f_t &= \frac{\pi_{22}}{\pi_{21}} & f_d &= \frac{\hat{\pi}_{22}}{\hat{\pi}_{21}} & \rho^* &= \frac{f_d - f_t}{f_t} > -1 \\ u_\pi &= |\pi_{22} - \pi_{21}| < 1 & r &= \frac{p_{22}}{p_{21}} & \sigma_k &= p_{2k}(1 - p_{2k}). \end{aligned} \tag{5.1.1}$$

In order to simplify the analysis without loss of generality, the convention $\pi_{22} \geq \pi_{21}$ is made. Thus, the attention is focused on situations where $f_t \geq 1$. By this convention, C_2 represents always the preponderant subclass in the target population when the target subclass structure is unbalanced.

The second convention is that $\rho^* < 0$. In this situation $f_d < f_t$ which is equivalent to $\hat{\pi}_{22} < \pi_{22}$. Thus, the preponderant subclass C_2 of the target population is always under-sampled (or under-represented) in the data at hand.

The case $\rho^* > 0$ ($\hat{\pi}_{22} > \pi_{22}$) is not investigated here. This situation is less likely to be encountered in the diagnostic practice than the previous one.

The true and observed subclass prevalences and some helpful relationships between f_t and f_d as well as between f_t and u_π can be easily derived from equations (5.1.1):

$$\begin{aligned} \pi_{21} &= \frac{1}{1+f_t} & \pi_{22} &= \frac{f_t}{1+f_t} \\ \hat{\pi}_{21} &= \frac{1}{1+f_d} & \hat{\pi}_{22} &= \frac{f_d}{1+f_d} \\ f_d &= (1+\rho^*)f_t & f_t &= \frac{1+u_\pi}{1-u_\pi}. \end{aligned} \tag{5.1.2}$$

Remark 5.1.1.

(i) Given the classification task for a particular target population, the quotient of true subclass prevalences f_t is known, while the quotient of observed subclass prevalences f_d is variable (depending on the data at hand). So, their relative difference ρ^* helps to keep track on how f_d varies relatively to f_t;

(ii) Given the general classification task, the quotient f_t is variable (depending on the target population); it is a strictly monotonically increasing function of the difference between the true subclass prevalences u_π.

The question of interest is how the weighting procedure impacts on the estimates of misclassification error or of class distribution parameters in the case that a suboptimal prevalence structure in the data is used for learning the classification rule.
The following particular subcases of the general case $\rho^* < 0$ are considered in more detail:

- Subcase 1 : *unbalanced true subclasses, balanced observed subclasses, thus* $f_t > 1$ and $f_d = 1$;

- Subcase 2 : *unbalanced true subclasses, at least diametrically opposite unbalanced observed subclasses, thus* $f_t > 1$ and $f_d = \frac{1}{bf_t}$ with $b \in [1, \infty)$.

Especially Subcase 1 is relevant for our theoretical survey, not only because it corresponds to the simulation designs from the precedent chapter, but it also resembles a situation which is commonly encountered in the differential diagnosis practice.

When $b = 1$, Subcase 2 addresses the case, in which the observed subclass prevalences are interchanged with respect to the true ones. Thus, $\hat{\pi}_{22} = \pi_{21}$ and correspondingly, $\hat{\pi}_{21} = \pi_{22}$, which gives rise to a highly suboptimal situation. When $b > 1$, the mismatch between the true and observed subclass prevalences is even more pronounced, since $\hat{\pi}_{22} < \pi_{21}$.

Table 5.2 offers an example of the target and observed subclass prevalences for each of these three particular situations.

Table 5.2: Illustration of the relevant subcases

Quotient of subclass prevalences	**General case** $\rho^* < 0$		
	Subcase 1	Subcase 2 $(b = 1)$	Subcase 2 $(b > 1)$
$f_t = \frac{\pi_{22}}{\pi_{21}}$	9	9	9
$f_d = \frac{\hat{\pi}_{22}}{\hat{\pi}_{21}}$	1	$\frac{1}{9}$	$\frac{1}{99}$

5.2 Weighted versus unweighted misclassification error estimates

In Chapter 4 some approaches were proposed to apply weighted misclassification rates not only in the phase of rule validation (4.2.16), but of rule optimization (4.2.12), too. Both optimization and validation are critical steps in designing a rule for practical classification tasks. Therefore, a careful analysis of the statistical behavior of weighted estimates should be carried out in advance to a final recommendation for practical problems. This chapter investigates the impact of weighting on estimates of the misclassification error addressing simultaneously the context of optimization and validation.

Under certain circumstances the bias reduction achieved by weighting may be countered by a considerable increase in the variability of the weighted estimates. This would discourage from using them. Therefore, the effect of weighting misclassification rates is studied in two steps. In the first step the difference between the means of weighted and unweighted estimates is evaluated. In the second step the global effect of weighting is assessed by a simultaneous comparison of mean squared errors of weighted and unweighted estimates. This is equivalent to a simultaneous comparison of their means and variances.

For the next investigations, the reader is reminded of the two important conventions made at the begin of this section. According to the first convention, subclass C_2 is at least equally prevalent to subclass C_1 in the target population, thus $f_t \geq 1$. According to the second convention, subclass C_2 is the under-represented subclass in the data at hand, i.e. $\rho^* < 0$, which is equivalent to $\hat{\pi}_{22} < \pi_{22}$.

5.2.1 Impact of weighting on the means of the misclassification error estimates

Like established in Chapter 4, *unweighted* estimates use the *natural weights*, thus the available subclass proportions in the data at hand. In *weighted* estimates the true subclass prevalences are applied as weights.

The density function of the heterogeneous class C is a mixture of the subclass density functions. The mixing proportions are the true subclass prevalences π_{21} and π_{22}. Given some classification rule δ, the following relationship holds:

$$\underbrace{P(\delta = 1|C)}_{=p_2} = \pi_{21}\underbrace{P(\delta = 1|C_1)}_{=p_{21}} + \pi_{22}\underbrace{P(\delta = 1|C_2)}_{=p_{22}}, \tag{5.2.1}$$

since subclasses C_1 and C_2 are disjoint.
Note that, using the jargon from Chapter 4, the true subclass misclassification probabilities p_{21} and p_{22} inform actually about the subclass degrees of overlap with the opposite class in the target population, given some rule δ.

The misclassification rates $\hat{\epsilon}_{2k}$, $k = 1, 2$, in the subclasses C_1 and C_2 determined by δ on an independent test data set, are unbiased estimates of the subclass error probabilities p_{21} and p_{22} (i.e. $E(\hat{\epsilon}_{2k}) = p_{2k}$). Linear combinations of them with coefficients given by the observed and true subclass prevalences yield the *unweighted* $\hat{\epsilon}_{unw}$ and the *weighted* $\hat{\epsilon}_w$ estimates of the misclassification probability p_2 (compare to (4.2.15) and (4.2.16)), respectively:

$$\begin{aligned} \hat{\epsilon}_{unw} &= \hat{\pi}_{21}\hat{\epsilon}_{21} + \hat{\pi}_{22}\hat{\epsilon}_{22} \\ \hat{\epsilon}_w &= \pi_{21}\hat{\epsilon}_{21} + \pi_{22}\hat{\epsilon}_{22}. \end{aligned} \tag{5.2.2}$$

Using the representation (5.2.2) and the decomposition formula (5.2.1), the weighted estimate is clearly unbiased, too. Thus, $E(\hat{\epsilon}_w) = p_2$. This represents the first advantage of weighted upon unweighted estimates.
It holds:

$$\begin{aligned}\hat{\epsilon}_w - \hat{\epsilon}_{unw} =& (\pi_{21} - \hat{\pi}_{21})\hat{\epsilon}_{21} + (\pi_{22} - \hat{\pi}_{22})\hat{\epsilon}_{22}\\ =& \frac{f_d - f_t}{(1+f_d)(1+f_t)}(\hat{\epsilon}_{21} - \hat{\epsilon}_{22})\\ =& \frac{\rho^* f_t}{[1+(1+\rho^*)f_t](1+f_t)}(\hat{\epsilon}_{21} - \hat{\epsilon}_{22}). \qquad (5.2.3)\end{aligned}$$

Hence, the difference between the theoretical means of the weighted and unweighted error estimates is:

$$E(\hat{\epsilon}_w) - E(\hat{\epsilon}_{unw}) = \frac{\rho^* f_t}{[1+(1+\rho^*)f_t](1+f_t)}(p_{21} - p_{22}), \qquad (5.2.4)$$

and their absolute difference is given by:

$$|\underbrace{E(\hat{\epsilon}_w) - E(\hat{\epsilon}_{unw})}_{=-Bias(\hat{\epsilon}_{unw})}| = \underbrace{\frac{-\rho^* f_t}{[1+(1+\rho^*)f_t](1+f_t)}}_{:=h(f_t;\rho^*)} \underbrace{|p_{21} - p_{22}|}_{=u_p}.$$

This difference depends on a function $h(f_t;\rho*)$ and the absolute difference between the estimates of the subclass error probabilities. The function $h(f_t;\rho^*) : [1,\infty) \times (-1,0) \longrightarrow (0,\infty)$ is strictly monotonically decreasing in ρ^* for every $f_t \geq 1$. This affirmation is sustained by the sign of its first derivative with respect to ρ^*:

$$\frac{\partial h(f_t;\rho^*)}{\partial \rho^*} = \frac{-f_t}{[1+(1+\rho^*)f_t]^2} < 0.$$

This result provides evidence for the fact that the larger the difference between the quotients f_d and f_t of the observed and true subclass prevalences, the bigger the absolute difference between weighted and unweighted misclassification error estimates.
In other words, the discrepancy between the observed and true subclass structures ρ^* is deciding for the evaluation of the weighting benefit.

Besides, it holds:

$$\frac{u_p}{2} = \lim_{\substack{\rho^*\to -1\\ f_t\to 1}} |Bias(\hat{\epsilon}_{unw})| \leq \lim_{\rho^*\to -1} |Bias(\hat{\epsilon}_{unw})| \leq \lim_{\substack{\rho^*\to -1\\ f_t\to\infty}} |Bias(\hat{\epsilon}_{unw})| = u_p.$$

Hence, in the context of a large mismatch between the observed and true subclass structures, i.e. when ρ^* close to -1, the absolute bias reduction by means of a weighted estimate is bounded. The more unbalanced the target subclass structure, the greater the bias reduction within the specified bounds. The larger the discrepancy u_p between the true subclass error probabilities, the higher the minimal bias reduction which can be expected by weighting ($\frac{u_p}{2}$).

5.2.2 Impact of weighting on the mean squared error of the misclassification error estimates

In Section 5.2.1 it is proved that a greater bias reduction through weighted estimates can be especially expected given a large mismatch between the observed and the true subclass prevalences and a large difference between the true subclass error probabilities.

However, the improvement in the bias of the misclassification error estimates by weighting may be counter-productive, which means that it is likely to be obtained at the price of a substantial variance enhancement. In view of this bias-variance trade-off, meaningful conditions are needed to describe the target situations in which weighted estimates are superior to the unweighted ones with respect to the mean squared error (MSE). This comprises simultaneous information on the bias and the variance of the estimates.

Using the notations from Table 5.1 and the relationships from (5.1.1) and (5.1.2) between ρ^*, f_d, and f_t, the difference between the variances of the weighted and unweighted misclassification error estimates is given by:

$$\begin{aligned}
Var(\hat{\epsilon}_w) - Var(\hat{\epsilon}_{unw}) =& Var(\pi_{21}\hat{\epsilon}_{21} + \pi_{22}\hat{\epsilon}_{22}) - Var(\hat{\pi}_{21}\hat{\epsilon}_{21} + \hat{\pi}_{22}\hat{\epsilon}_{22}) \\
=& (\pi_{21}^2 - \hat{\pi}_{21}^2)Var(\hat{\epsilon}_{21}) + (\pi_{22}^2 - \hat{\pi}_{22}^2)Var(\hat{\epsilon}_{22}) + 2(\pi_{21}\pi_{22} - \hat{\pi}_{21}\hat{\pi}_{22})\underbrace{Cov(\hat{\epsilon}_{21}, \hat{\epsilon}_{22})}_{=0} \\
=& \frac{(f_d - f_t)(2 + f_d + f_t)}{(1 + f_d)^2(1 + f_t)^2}Var(\hat{\epsilon}_{21}) + \frac{(f_t - f_d)[2f_d f_t + f_d + f_t]}{(1 + f_d)^2(1 + f_t)^2}Var(\hat{\epsilon}_{22}) \\
=& \frac{\rho^* f_t[2 + (2 + \rho^*)f_t]}{[1 + (1 + \rho^*)f_t]^2(1 + f_t)^2}Var(\hat{\epsilon}_{21}) - \frac{\rho^* f_t[2(1 + \rho^*)f_t^2 + (2 + \rho^*)f_t]}{[1 + (1 + \rho^*)f_t]^2(1 + f_t)^2}Var(\hat{\epsilon}_{22}) \\
=& \left[\frac{\rho^* f_t}{[1 + (1 + \rho^*)f_t]^2(1 + f_t)^2}\right] \cdot \Big\{[2 + (2 + \rho^*)f_t]\underbrace{Var(\hat{\epsilon}_{21})}_{:=v_1} \\
& - [2(1 + \rho^*)f_t^2 + (2 + \rho^*)f_t]\underbrace{Var(\hat{\epsilon}_{22})}_{:=v_2}\Big\}.
\end{aligned} \tag{5.2.5}$$

The subclass misclassification error estimates $\hat{\epsilon}_{21}$ and $\hat{\epsilon}_{22}$ are based on disjoint sets of observations in the validation data set, which correspond to the distinct subclass labels C_1 and C_2. Note that they are computed on the validation data set which is independent from the training data set. Therefore, an erroneous assignment of an observation from subclass C_1 is independent from an erroneous assignment of an independently sampled observation from subclass C_2, given a particular rule developed on the training data set. Thus, it is plausible that the covariance $Cov(\hat{\epsilon}_{21}, \hat{\epsilon}_{22})$ of the subclass misclassification error estimates assessed on the validation data is 0.

Starting from the well-known computational formula of the MSE as a sum between the variance and the squared bias, the difference between the MSEs of weighted and unweighted misclassification error estimates corresponds to:

$$\begin{aligned} MSE(\hat{\epsilon}_w) - MSE(\hat{\epsilon}_{unw}) =& \Big[Var(\hat{\epsilon}_w) - Var(\hat{\epsilon}_{unw}) + \underbrace{Bias(\hat{\epsilon}_w)^2}_{=0} - Bias(\hat{\epsilon}_{unw})^2\Big] \\ =& \Big\{Var(\hat{\epsilon}_w) - Var(\hat{\epsilon}_{unw}) - [E(\hat{\epsilon}_{unw}) - E(\hat{\epsilon}_w)]^2\Big\}. \end{aligned} \tag{5.2.6}$$

Replacing (5.2.4) and (5.2.5) into (5.2.6) yields:

$$\begin{aligned} MSE(\hat{\epsilon}_w) - MSE(\hat{\epsilon}_{unw}) =& \Big[\underbrace{\frac{\rho^* f_t}{[1 + (1+\rho^*)f_t]^2(1+f_t)^2}}_{:=g(\rho^*, f_t)}\Big] \cdot \Big\{[2 + (2+\rho^*)f_t]v_1 \\ & - [2(1+\rho^*)f_t + 2 + \rho^*]f_t v_2 - \rho^* f_t u_p^2\Big\}. \end{aligned} \tag{5.2.7}$$

Generally, weighting provides a benefit if the following inequality holds:

$$MSE(\hat{\epsilon}_w) < MSE(\hat{\epsilon}_{unw}),$$

or equivalently:

$$[2 + (2+\rho^*)f_t]v_1 - [2(1+\rho^*)f_t + 2 + \rho^*]f_t v_2 > \rho^* f_t u_p^2, \tag{5.2.8}$$

since $\rho^* < 0$.

The theoretical variances $v_k,\ k = 1, 2$ of the subclass misclassification error rates depend on the subclass error probabilities and sizes. Since the number of misclassifications in each subclass is a binomially distributed variable (i.e. $N_{2k}\hat{\epsilon}_{2k} \sim \text{Bin}(p_{2k}, N_{2k}),\ k = 1, 2$), the theoretical variances v_k of the subclass misclassification rates are defined as:

$$v_k = \frac{\sigma_k}{N_{2k}} = \frac{p_{2k}(1 - p_{2k})}{N_{2k}},\ k = 1, 2. \tag{5.2.9}$$

The subclass sizes can be derived from the size of the heterogeneous class N_2, using the quotient f_t of the true subclass prevalences and the relative suboptimality ρ^* of the data at hand:

$$N_{21} = \frac{N_2}{[1 + (1 + \rho^*)f_t]} \qquad N_{22} = \frac{N_2(1 + \rho^*)f_t}{[1 + (1 + \rho^*)f_t]}. \tag{5.2.10}$$

Remark 5.2.1. If the special subcases are considered, these expressions simplify to:

(1) **Subcase 1**: $N_{21} = N_{22} = \frac{N_2}{2}$;

(2) **Subcase 2 ($b \geq 1$)**: $N_{21} = \frac{N_2 b f_t}{1+bf_t}$ and $N_{22} = \frac{N_2}{1+bf_t}$.

Using the quotient $s = \frac{\sigma_2}{\sigma_1}$ between the variances of the misclassification events in subclass C_2 and C_1 (see Table 5.1), the necessary condition for the existence of a weighting benefit (5.2.8) can be reformulated as:

$$\underbrace{\left[\frac{1 + (1 + \rho^*)f_t}{\rho^* f_t}\right]\left\{[2 + (2 + \rho^*)f_t] - \frac{[2(1 + \rho^*)f_t + 2 + \rho^*]}{1 + \rho^*}s\right\}}_{:=\tilde{G}(s,\rho^*,f_t)} < \frac{N_2 u_p^2}{\sigma_1}. \tag{5.2.11}$$

The theoretical variances v_k, $k = 1, 2$, are replaced by their formula in terms of ρ^*, f_t and the size of the heterogeneous class N_2. Also $\sigma_1 > 0$ is assumed, which is equivalent to the fact that p_{21} is different from 0 or 1.

Given some true and observed subclass structures, the existence of a weighting benefit depends on how large the right hand side term of condition (5.2.11) is. Thus, the chances for an advantageous weighting are greater when the class sample size N_2 is large, when subclass error probabilities are highly discrepant (i.e. $|u_p|$ is large) and the error probability p_{21} in the less prevalent subclass is either close to 0 or close to 1 (i.e. σ_1 is small). For small values of $s < 1$, the chances of a weighting benefit increase additionally. This means that a particularly favorable situation for weighting is given when p_{21} is small and $p_{22} > 1 - p_{21}$ or p_{21} is large and $p_{22} < 1 - p_{21}$.

Figures 5.1 (a) and (b) show the interpretation of s in terms of the subclass error probabilities for values below and above 1.

Given some fixed $p_{21} < 0.5$, if $s < 1$, then p_{22} is an element of $[0, p_{21}) \bigcup (1 - p_{21}, 1]$. Very small values of $s < 1$ are associated to values of p_{22} which are close to 0 (perfect separation between C_2 and D) or 1 (complete overlap between C_2 and D).

Given some fixed $p_{21} < 0.5$, if $s > 1$, then s is upper bounded by $0.25\sigma_1^{-1}$ and p_{22} is an element of $(p_{21}, 1 - p_{21})$. Thus, values of $s > 1$ close to its upper bound correspond to values of p_{22}

which are close to 0.5.

Thus, given a fixed value of p_{21}, the larger $s > 1$ when p_{21} and p_{22} are on the same side of 0.5, or the smaller $s < 1$, the larger the discrepancy between the subclass error error probabilities and therefore, u_p.

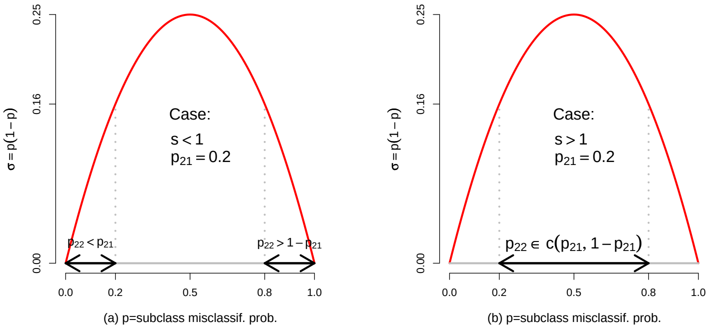


Figure 5.1: Interpretation of $s = \frac{p_{22}(1-p_{22})}{p_{21}(1-p_{21})}$, the quotient of the subclass variances of the misclassification event. *Fig. (a) when* $s < 1$ *and* $p_{21} < 0.5$, *the error probability* p_{22} *in* C_2 *belongs to* $[0, p_{21}) \bigcup (1 - p_{21}, 1]$. *Fig. (b) when* $s > 1$ *and* $p_{21} < 0.5$, p_{22} *belongs to* $(p_{21}, 1 - p_{21})$.

Two alternative descriptions of condition (5.2.11) may be used for investigating the existence of a benefit from weighting:

(1. alternative)

$$\tilde{G} < \frac{N_2 u_p^2}{\sigma_1} \quad (\sigma_1 \neq 0) \tag{5.2.12}$$

(2. alternative)

$$G =: \frac{\tilde{G}\sigma_1}{u_p^2} < N_2 \quad (u_p \neq 0). \tag{5.2.13}$$

Note that condition (5.2.11), or equivalently, the sign of one of the differences $\tilde{G} - \frac{N_2 u_p^2}{\sigma_1}$ and $G - N_2$, is useful only to verify whether the weighting benefit exists, or not. The particular value of these differences gives no hint about the amount of benefit since:

$$MSE(\hat{\epsilon}_w) - MSE(\hat{\epsilon}_{unw}) = \frac{\rho^* f_t \sigma_1}{N_2} \cdot g(\rho^*, f_t) \cdot \left[\tilde{G} - \frac{N_2 u_p^2}{\sigma_1}\right]$$

or in terms of G, equivalently:

$$MSE(\hat{\epsilon}_w) - MSE(\hat{\epsilon}_{unw}) = \frac{\rho^* f_t u_p^2}{N_2} \cdot g(\rho^*, f_t) \cdot [G - N_2] \,.$$

Thus, the amount of benefit associated to a particular configuration of the target population and of the data set at hand can be only assessed by considering the entire difference between the mean squared errors. However, our interest is focussed here just on the existence of a weighting benefit.

N_2 is detached on the right hand side of the inequations (5.2.13) and (5.2.12) since it depends on the particular study. A low value of N_2 can be used as *worst case*-reference to check the existence of a weighting benefit in the context of a small sample size.

With the notations from Table 5.1 the pro and contra arguments for the two alternative conditions are resumed in Table 5.3. Both formulations of the benefit condition, (5.2.13) and (5.2.12), are more or less afflicted with difficulties regarding the analysis of their domain of validity. For instance, the parameters s, σ_1 and u_p are connected to each other. They are all defined in terms of p_{21} and p_{22}. This shows that in the first alternative formulation (5.2.12) of condition (5.2.11) the right and left hand side terms are not independent, which is the greatest disadvantage of the $\tilde{G}$-formulation.
In spite of this drawback, the definition of a *worst case*-reference for the right hand side term enables in this case a reduction of the problem to only three arguments: s, ρ^* and f_t. With this simplification the inequation based on $\tilde{G}$ can be handled much easier than its counterpart based on G, which allows for no reductions in its quadruple of arguments.

In the G-formulation of the benefit condition all parameters of interest are included in G. There is no spurious dependence between the left and the right hand side terms. Anyway, a shortcoming is that the number of four arguments on the left hand side cannot be reduced further. This also hinders a precise and at the same time easy to understand mathematical description of the validity domain of this condition.

The G-alternative is suitable for a general investigation of the existence of the weighting benefit. For the special subcases we prefer however, the $\tilde{G}$-alternative. A simplified analysis is possible by defining a *worst case* reference for the right hand side term and taking advantage on the reduced parameter set in the left hand side term. But, one has to take care in the interpretation of the results, since a worst-case reference is used.

	$\tilde{G}$**-alternative (5.2.12)**	G**-alternative (5.2.13)**
Condition	$\tilde{G} < \frac{N_2 u_p^2}{\sigma_1}$	$G < N_2$
pro's	• simplified functional form of $\tilde{G}$ • a *worst case*-reference for the right term allows a problem reduction to 3 arguments (s, ρ^* and f_t) *Attention! The reference depends on* $s = \frac{\sigma_2}{\sigma_1}$ *!*	• independent left and right terms • compressed form of the left term • *worst case*-reference only for N_2
contra's	• dependence between left and right terms	• complicated left term at least 4 parameters necessary

Table 5.3: Overview of the G and $\tilde{G}$-conditions: Pro and contra arguments. *A negative sign of* $\tilde{G} - \frac{N_2 u_p^2}{\sigma_1}$ *or of* $G - N_2$ *indicates a win and a positive sign a loss in the efficiency of the weighted estimates with respect to the unweighted ones.*

Generally, G can be expressed as a function that, beside the invariable arguments f_t and ρ^*, involves the error probability in the less prevalent target subclass p_{21} and the quotient of the true subclass error probabilities r:

$$\begin{aligned} G(p_{21}, r, \rho^*, f_t) = & \left[\frac{1 + (1+\rho^*)f_t}{\rho^* f_t}\right] \cdot \left[\frac{1 - p_{21}}{(r-1)^2 p_{21}}\right] \cdot \\ & \cdot \left\{ [2 + (2+\rho^*)f_t] - \frac{[2(1+\rho^*)f_t + 2 + \rho^*]}{1+\rho^*} \left[\frac{r(1 - r p_{21})}{1 - p_{21}}\right] \right\}, \end{aligned} \qquad (5.2.14)$$

with the restrictions $r \neq 1$ and $p_{21} \leq r^{-1}$.

When all arguments have known values, function G is easy to evaluate. Then the minimally requested size of the heterogeneous class to ensure a weighting benefit is $N_2^{min} = [G] + 1$. If in the practical application the actual size of the heterogeneous class is smaller than N_2^{min}, N_2 might be appropriately enhanced keeping ρ^* (thus, actually f_d) unchanged.

Normally, at least the values of f_t and ρ^* are available. If another argument is unknown, then the evaluation of the G-condition becomes complicated. A description of the domain of an argument which is associated to the weighting benefit is not possible without involving some other arguments due to the complexity of G.

Tables 5.4 and 5.5 offer potentially useful criteria to decide upon using weighting or preferably avoiding it, when $r \leq 0.5$ and $r \geq 2$, respectively. Domains of the four arguments of G are presented, for which the function has a negative value. Therefore, these domains are associated to a sure weighting benefit.

However, both tables are designed just for orientating purposes. They address a user with a rather vague idea of the real values of the function arguments.

Values of r within $(0.5, 2)$ are not considered in these tables. Given such r-values, one should proceed with caution, since function G is positive on some interval around 1, like it is shown in Table A.1.

Table 5.4: Discussion around the weighting benefit based on the G-alternative when $r \leq 0.5$. *Domains of p_{21}, ρ^*, f_t, and $r \leq 0.5$ associated to a negative sign of $G(p_{21}, r, \rho^*, f_t)$, i.e with a sure weighting benefit.*

Row	p_{21}	ρ^*	$f_t = \frac{\pi_{22}}{\pi_{21}}$	r
1	$(0, 0.5]$[1]	$[-0.5, 0)$[2]	$[1, 2)$[4]	$(0, 0.45]$
2	$(0, 0.4]$	$[-0.7, -0.5)$[3]	$[1, 2)$	$(0, 0.37]$
3	$(0, 0.5]$	$[-0.5, 0)$	$[2, 99)$[5]	$(0, 0.47]$
4	$(0, 0.4]$	$[-0.7, -0.5)$	$[2, 99)$	$(0, 0.39]$
5	$(0, 0.5]$	$[-0.5, 0)$	$[99, \infty)$[6]	$(0, 0.50]$
6	$(0, 0.5]$	$[-0.7, -0.5)$	$[99, \infty)$	$(0, 0.40]$
7	$(0.5, 0.6]$	$[-0.25, 0)$	$[1, 2)$	$(0, 0.49]$
8	$(0.5, 0.6]$	$[-0.6, -0.5)$	$[1, 2)$	$(0, 0.30]$
9	$(0.5, 0.6]$	$[-0.25, 0)$	$[2, 99)$	$(0, 0.49]$
10	$(0.5, 0.55]$	$[-0.7, -0.5)$	$[2, 99)$	$(0, 0.30]$
11	$(0.5, 0.6]$	$[-0.5, 0)$	$[99, \infty)$	$(0, 0.39]$
12	$(0.5, 0.6]$	$[-0.7, -0.5)$	$[99, \infty)$	$(0, 0.32]$

[1] small to moderate error probability of the less prevalent target subclass, C_1.

[2] small to moderate absolute degree of suboptimality $|\rho^*|$.

[3] moderate to large absolute degree of suboptimality $|\rho^*|$.

[4] the target subclass structure is balanced or moderately unbalanced.

[5] the target subclass structure is highly unbalanced.

[6] the target subclass structure is extremely unbalanced.

Mathematical proofs for the statements provided in Tables 5.4 and 5.5 are available in Appendix A.1. Here only the proofs for two particular cases of interest are considered: (1) $r = 1$ (2) $\sigma_1 = 0$.

Table 5.5: Discussion around the weighting benefit based on the G-alternative when $r \geq 2$. *Domains of p_{22}, ρ^*, f_t, and $r \geq 2$ associated to a negative sign of $G(p_{22}r^{-1}, r, \rho^*, f_t)$, i.e with a sure weighting benefit.*

Row	r	ρ^*	$f_t = \frac{\pi_{22}}{\pi_{21}}$	p_{22}
1	$[2, 5]$	$[-0.2, 0)$	$[1, 2)$	$[0.85, 1]$
2	$[2, 4.5]$	$[-0.7, -0.5)$	$[1, 2)$	$[0.91, 1]$
3	$[2, 5]$	$[-0.2, 0)$	$[2, 99)$	$[0.85, 1]$
4	$[2, 4.5]$	$[-0.7, -0.5)$	$[2, 99)$	$[0.91, 1]$
5	$[2, 5]$	$[-0.2, 0)$	$[99, \infty)$	$[0.85, 1]$
6	$[2, 5]$	$[-0.8, -0.5)$	$[99, \infty)$	$[0.90, 1]$

When $r = 1$, thus the subclass error probabilities are equal, weighted error estimates should be strictly avoided, like it is stated by Proposition 5.2.1.

Proposition 5.2.1. Given equal subclass error probabilities, no benefit is achievable by weighting.

Proof. *The true subclass error probabilities are equal (i.e. $p_{21} = p_{22}$). Equivalently, r and s are* 1*, u_p is* 0 *(see their definition in Table 5.1).*
Condition (5.2.11) resumes to:

$$[(\rho^* + 2)f_t + 2] - \frac{[2(\rho^* + 1)f_t + \rho^* + 2]}{\rho^* + 1} > 0.$$

This inequality allows a further simplification to a condition which is never accomplished:

$$f_t < -\frac{1}{1 + \rho^*}.$$

When $\sigma_1 = 0$, thus the error probability p_{21} of the less prevalent target subclass is either 0 or 1, a general advice is to avoid weighting when the absolute degree of suboptimality of the data at hand $|\rho^*|$ is rather small. According to Proposition 5.2.2 values of $|\rho^*|$ below 0.7 may lead to suboptimal weighted estimates.
If p_{21} is 0, then the chances for a beneficial weighting increase when p_{22} approaches 1 and if p_{21} is 1, the chances increase for p_{22} approaching 0.

Proposition 5.2.2. Let the error probability p_{21} in the less prevalent target subclass be 0 or 1. Then, maximal chances for a weighting benefit are available for ρ^* within some small interval around $\rho_0^* = -\frac{\sqrt{2}(1+f_t)}{1+\sqrt{2}(1+f_t)}$. In particular, weighting may be dangerous for an absolute degree of suboptimality $|\rho^*|$ up to 0.7.

Proof. *The condition (5.2.8) for the existence of a benefit by weighting becomes:*

$$\underbrace{\left(\frac{1-p_{22}}{p_{22}}\right)^{\pm 1}\left[\frac{-1}{\rho^*(1+\rho^*)f_t}\right][2(1+\rho^*)f_t+2+\rho^*][1+(1+\rho^*)f_t]}_{:=H(p_{22},\rho^*,f_t)} < N_2, \tag{5.2.15}$$

where the ±*-power of the first component in the left hand side term depends on whether* p_{21} *is* 0 *or* 1.

Let f_t *and* p_{22} *be constant. Thus, the subclass prevalence structure in the target population is fixed. The derivative of the left hand side term of the inequation (5.2.15) with respect to the relative degree of suboptimality of the data set at hand* ρ^* *is:*

$$\frac{\partial H}{\partial \rho^*} = \left(\frac{1-p_{22}}{p_{22}}\right)^{\pm 1}\left[\frac{1}{f_t\rho^{*2}(1+\rho^*)^2}\right]\left[2(1+f_t)^2+4(1+f_t)^2\rho^*+(1+4f_t+2f_t^2)\rho^{*2}\right].$$

The zeros of the quadratic expression in brackets are:

$$\begin{aligned}\rho^*_{0,1}(f_t) =& \frac{-2(1+f_t)^2 \pm \sqrt{2}(1+f_t)}{(1+4f_t+2f_t^2)}\\ =& \frac{-\sqrt{2}(1+f_t)[\sqrt{2}(1+f_t)\mp 1]}{[\sqrt{2}(1+f_t)-1][\sqrt{2}(1+f_t)+1]}\\ =& \frac{-\sqrt{2}(1+f_t)}{[\sqrt{2}(1+f_t)\pm 1]}.\end{aligned} \tag{5.2.16}$$

Only $\rho^*_0(f_t) = -\frac{\sqrt{2}(1+f_t)}{1+\sqrt{2}(1+f_t)}$ *is an element of* $(-1,0)$*, while* $\rho^*_1(f_t) < -1$*. Consequently, this derivative is negative on the left hand side of* ρ^*_0 *and positive on its right hand side. This indicates that H achieves its minimum at* $\rho^*_0(f_t)$*. The chance for an accomplishment of the benefit condition increases around this point. On both sides of it, the chances for a benefit by weighting decrease. Besides, it holds:*

$$\lim_{f_t\to 1}\rho^*_0(f_t) = \frac{-2\sqrt{2}}{2\sqrt{2}+1} \approx -0.73,$$

and

$$\lim_{f_t\to\infty}\rho^*_0(f_t) = -1,$$

and ρ^*_0 *is monotonically decreasing with respect to* f_t*. Therefore, the smaller the absolute degree of suboptimality* $|\rho^*|$ *in comparison to* 0.7*, the more dangerous the weighted estimates.*

5.2.3 Preliminary discussion of subcases

Function G, which is used to prove the existence of a benefit by weighting the error estimates, has a complex mathematical structure in its four arguments. We focus on the special subcases which are exemplified in Table 5.2.

Function $\tilde{G}$ depends on only three parameters. Therefore, the $\tilde{G}$-alternative formulation of the benefit condition (5.2.12) is used in the further theoretical discussion to take even more advantage on the already simplified form of the particular subcases.

In the following, we assume that a fixed combination of s and σ_1 is given.
Each of the subclass error probabilities $p_{2k},\ k = 1, 2$, satisfies:

$$p_{2k}(1 - p_{2k}) = \sigma_k.$$

Thus, p_{21} and p_{22} are solutions of the equations:

$$x^2 - x + \sigma_1 = 0$$

and

$$x^2 - x + s\sigma_1 = 0,\ 4s\sigma_1 \leq 1,$$

respectively.

We define the *worst case*-reference for the right hand side term of condition (5.2.12) using the minimal gap u_p between the subclass error probabilities. This is achieved when p_{22} lies on the same side of 0.5 like p_{21}. Formally, this means that

$$(p_{21} - 0.5)(p_{22} - 0.5) > 0.$$

Thus, the *worst case* reference is:

$$\frac{N_2 u_p^2}{\sigma_1} = N_2 \cdot \underbrace{\left(\frac{\sqrt{1 - 4\sigma_1 s} - \sqrt{1 - 4\sigma_1}}{2\sqrt{\sigma_1}} \right)^2}_{:=q_{min}}. \tag{5.2.17}$$

Now, assume p_{22} lies on the other side of 0.5 than p_{21}. Formally, this means that

$$(p_{21} - 0.5)(p_{22} - 0.5) < 0.$$

The right hand side term becomes:

$$\frac{N_2 u_p^2}{\sigma_1} = N_2 \cdot \underbrace{\left(\frac{\sqrt{1-4\sigma_1 s} + \sqrt{1-4\sigma_1}}{2\sqrt{\sigma_1}}\right)^2}_{:=q_{max}}. \tag{5.2.18}$$

This term mirrors the best scenario given fixed values of N_2, s and σ_1, and is further distinguished from the former reference by calling it *best case* reference.

Notation. Given s and σ_1, q is generally defined as the quotient between the squared difference of the true subclass error probabilities, u_p^2, and the variance of the misclassification event in the over-represented subclass, σ_1:

$$q := \frac{u_p^2}{\sigma_1}.$$

The notations q_{min} and q_{max} are used to make the difference between the situations in which q appears in the *worst* and the *best case* reference, respectively. Thus,

$$q_{min,max} =: \left(\frac{\sqrt{1-4\sigma_1 s} \mp \sqrt{1-4\sigma_1}}{2\sqrt{\sigma_1}}\right)^2. \tag{5.2.19}$$

Remark 5.2.2 (*Worst case reference*)**.**

(i) The function q_{min} and therefore, the *worst case* reference, is strictly monotonically decreasing on $(0,1]$ and strictly monotonically increasing on $[1,\infty)$ in the argument s, the quotient of the probabilities for an error event in the predominant and in the less prevalent target subclasses.

(ii) For all s, the function q_{min} and therefore, the *worst case* reference, is strictly monotonically increasing with respect to σ_1, the variance of the misclassification event in the small target subclass.

Remark 5.2.3 (*Best case reference*)**.**

The function q_{max} and therefore, the *best case* reference, is strictly monotonically decreasing both in s and σ_1.

The monotonicity of the *worst* and *best case* references with respect to s and σ_1 stated by Remarks 5.2.2 and 5.2.3 is proved in Appendix A.2.

Note 1 (*Generalization starting from the worst case reference*)**.** If the benefit condition is accomplished with respect to the *worst case* reference (i.e. for the minimal value of u_p given s and σ_1), then it is accomplished with respect to the *best case* reference (i.e. for the maximal value of u_p given s and σ_1), too. This is clear since $q_{min}(s, \sigma_1) < q_{max}(s, \sigma_1)$.
Proposition 5.2.3 enables a generalization of these observations when σ_1 is variable. It helps to draw more general conclusions from the next graphical investigations, although they are based on a constant value of σ_1.

Proposition 5.2.3. Assume $s < 1$ and the variance of the error event in subclass C_1, $\sigma_1 \in (0, 0.25]$, are constant.

(i) If the weighting benefit is sure relatively to the *worst case* reference given $\sigma_1 = \sigma_{const}$, then it is sure relatively to this reference for every $\sigma_1' \in [\sigma_{const}, 0.25]$;

(ii) If the weighting benefit is sure relatively to the *best case* reference given $\sigma_1 = \sigma_{const}$, then it is sure relatively to this reference for every $\sigma_1' \in (0, \sigma_{const}]$.

(iii) If the weighting benefit is sure relatively to the *worst case* reference given $\sigma_1 = \sigma_{const}$, then it is sure relatively to the *best case* reference for every $\sigma_1' \in (0, 0.25]$.

Proof. *(i) The weighting benefit is sure with respect to the* worst case *reference, given $s < 1$ and some constant value of σ_1, iff:*

$$\tilde{G} < N_2 q_{min}(s, \sigma_{const}). \qquad (*)$$

Using the monotonic behavior of q_{min} with respect to σ_1, stated in Remark 5.2.2, it holds:

$$\tilde{G} < N_2 q_{min}(s, \sigma_{const}) \leq N_2 q_{min}(s, \sigma_1'), \qquad (**)$$

for any $\sigma_1' \in [\sigma_{const}, 0.25]$.

(ii) Due to the strictly decreasing monotonicity of q_{max} with respect to σ_1 established by Remark 5.2.3 it results:

$$\tilde{G} < N_2 q_{max}(s, \sigma_{const}) \leq N_2 q_{max}(s, \sigma_1'),$$

for every $\sigma'_1 \leq \sigma_{const}$.

(iii) Starting from $(*)$ *and using the monotonic behavior of* q_{max} *with respect to* σ_1, *stated in Remark 5.2.3, it holds:*

$$\tilde{G} < N_2 q_{min}(s, \sigma_{const}) < N_2 q_{max}(s, \sigma_{const}) < N_2 q_{max}(s, \sigma''_1),$$

for any $\sigma''_1 \in (0, \sigma_{const})$. *Starting from* $(**)$, *it results:*

$$\tilde{G} < N_2 q_{min}(s, \sigma_{const}) \leq N_2 q_{min}(s, \sigma'_1) < N_2 q_{max}(s, \sigma'_1),$$

for any $\sigma'_1 \in [\sigma_{const}, 0.25]$.
Thus, if the benefit condition is accomplished with respect to the worst case *reference at least for one value of* σ_1, *then it is accomplished also with respect to the* best case *reference for any* $\sigma_1 \in (0, 0.25]$, *provided that* s *remains constant.*

Corollary 5.2.4.

(i) The results (i)-(iii) apply given any $s > 1$ and any constant value of σ_1, $\sigma_{const} \in (0, 0.25s^{-1})$, too.

(ii) Assume p^0_{21} is the subclass error probability associated to σ_{const} from Proposition 5.2.3. Then, keeping s constant:

(a) The benefit in Proposition 5.2.3 (i) is sure given any p_{21} between p^0_{21} and $1 - p^0_{21}$.

(b) The benefit in Proposition 5.2.3 (ii) is sure given any $p_{21} \leq p^0_{21} < 0.5$ or $p_{21} \geq p^0_{21} > 0.5$.

(c) The benefit in Proposition 5.2.3 (iii) is sure given any p_{21} when $s < 1$, and given any p_{21} for which $\sigma_1 \leq 0.25s^{-1}$ when $s > 1$.

Proof. *If* $s > 1$, *the request that* $\sigma_1 = \sigma_{const} < 0.25s^{-1}$ *guarantees that the term* $\sqrt{1 - 4\sigma_1 s}$ *from the definition of the* worst *and* best case *references is well defined.*

(i) The monotonicity of q_{min} *and* q_{max} *with respect to* σ_1 *does not depend on* s. *Thus, the proofs for cases (i)-(iii) in Proposition 5.2.3, where* $s < 1$, *apply also here.*

(iia) Here it should be reminded that $\sigma_{const} = p^0_{21}(1 - p^0_{21})$ *and* $\sigma'_1 = p_{21}(1 - p_{21})$. *According to Proposition 5.2.3 (i), if the benefit is sure relatively to the* worst case *reference when* $\sigma_1 = \sigma_{const}$, *then it is sure for every* $\sigma_1 = \sigma'_1 \geq \sigma_{const}$. *Since the function* $f(x) = x(1 - x)$ *is symmetrical around* 0.5 *and monotonically increasing on* $(0, 0.5)$ *and decreasing on* $(0.5, 1)$, *the benefit is*

sure for any p_{21} between p_{21}^0 and $1 - p_{21}^0$.

(iib) Again the properties of function f are used, starting from $\sigma_1' \leq \sigma_{const}$.

(iic) This result is clear starting from Proposition 5.2.3 (iii) and using (iia) and (iib).

Note 2 (*General graphical settings*)**.** In the next graphical investigations to the subcases, the *worst* and *best case* references are computed for $N_2 = 100$. The variance σ_1 of the misclassification event in the less prevalent target subclass C_1 is set on some feasible value, in each of the cases $s > 1$ and $s < 1$. Without restricting the generality of this investigation we assume that the corresponding error probability p_{21} is smaller than 0.5.
When $s > 1$, $\sigma_1 = 0.05$ is used to compute both references. Correspondingly, the misclassification error probability in C_1 is $p_{21} \approx 0.05$.
When $s < 1$, σ_1 is set on a medium level, i.e. $\sigma_1 = 0.125$. Correspondingly, the misclassification error probability in C_1 is $p_{21} \approx 0.15$.

5.2.4 Subcase 1

In this case the target population is unbalanced while the data set at hand is balanced with respect to the subclass prevalence structure, which is equivalent to $f_t > 1$ and $f_d = 1$. This situation is usually encountered in the diagnostic practice.

The relative difference between the true and observed subclass prevalence structures ρ^* is now expressed just in terms of the quotient of the true subclass prevalences, f_t:

$$\rho^* = \frac{1 - f_t}{f_t}.$$

We replace ρ^* in the $\tilde{G}$-condition (5.2.12) by its representation in terms of f_t. The weighting benefit is available if the condition:

$$\underbrace{\frac{2[(3f_t + 1)s - (3 + f_t)]}{f_t - 1}}_{=\tilde{G}(s, \frac{1-f_t}{f_t}, f_t)} < \frac{N_2 u_p^2}{\sigma_1} \tag{5.2.20}$$

is accomplished.

5.2.4.1 Graphical investigation

Given a fixed value of s, the quotient of the true subclass prevalences f_t is tuned over the range $[1 + 10^{-4}, 20]$ in steps of 10^{-4}. The *worst* and *best case* references are based on $N_2 = 100$ and a σ_1 which is specified like in Note 2, depending on whether $s > 1$ or $s < 1$.

Figures 5.2 and 5.3 show $\tilde{G}$ (solid lines) with respect to the *worst case* reference (dotted lines) when $s \geq 1$ and $s \leq 1$, respectively. The same plots with respect to the *best case* reference for the same s-values can be viewed in Appendix B, in Figures B.1 and B.2, respectively.
Tables 5.6 and 5.7 present the low and upper bounds of the f_t-intervals associated to a weighting benefit when $s \geq 1$ and $s \leq 1$, respectively. These describe the domains on which $\tilde{G}$ lies below the *worst* and *best case* references in Figures 5.2-5.3 and B.1-B.2, respectively.

Let $\boldsymbol{s \geq 1}$. With the settings from Note 2, this means that $p_{21} \approx 0.05$ and $p_{22} \in [0.05, 0.5]$ or $p_{22} \in [0.5, 0.95]$, depending on whether the *worst* or the *best case* reference is considered.

It is important to notice that, given a constant value of σ_1, then every value $s \geq 1$ is associated to a discrepancy of:

$$100u_p = 100(p_{22} - p_{21}) = 100\left(\frac{\sqrt{1-4\sigma_1} - \sqrt{1-4\sigma_1 s}}{2}\right) \tag{5.2.21}$$

percentage points between the subclass error probabilities, if they are situated on the same side of 0.5 (*worst case* reference). This increases obviously with increasing values of s.
If the subclass error probabilities are situated on different sides of 0.5 (*best case* reference), then the discrepancy between them is of:

$$100u_p = 100(p_{22} - p_{21}) = 100\left(\frac{\sqrt{1-4\sigma_1} + \sqrt{1-4\sigma_1 s}}{2}\right)$$

percentage points, given s. This increases obviously with decreasing values of s.

In this way, from left to right, the p_{22} values which correspond to the first row of Table 5.6 (*worst case*) are tuned between $0.05(\approx p_{21})$ and 0.5. At 0.5 the discrepancy u_p between the subclass error probabilities is maximal.
Similarly, but from right to left, the p_{22} values which correspond to the second row of Table 5.6 (*best case*) are tuned between 0.5 and $0.95(\approx 1 - p_{21})$, where u_p is maximal.

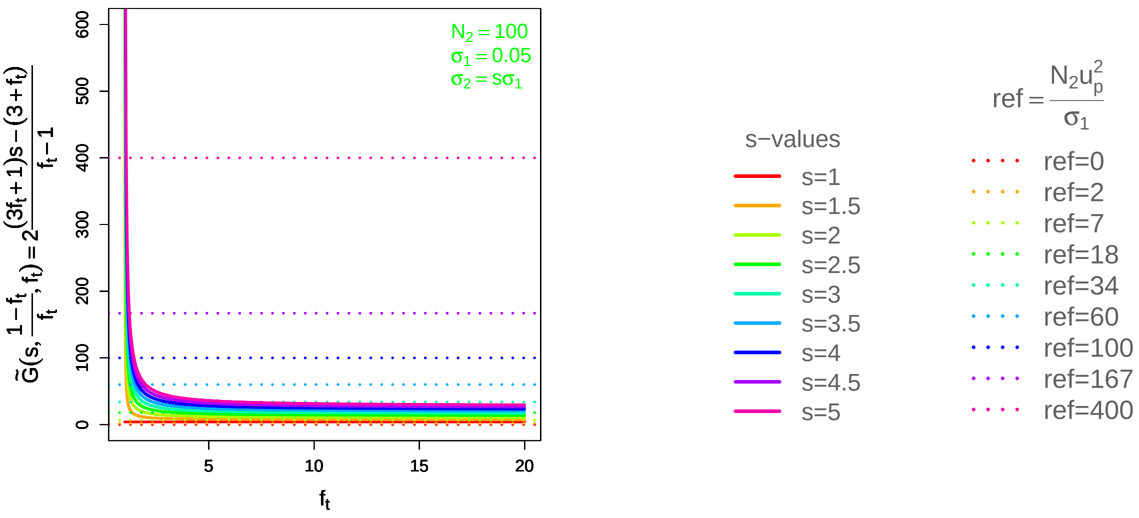


Figure 5.2: Subcase 1: Relation between $\tilde{G}(s, \frac{1-f_t}{f_t}, f_t)$ and the *worst case* reference when $s \geq 1$, $\sigma_1 = 0.05$ and $N_2 = 100$. *Given $s \geq 2.5$, weighting is beneficial for a quotient of the true subclass prevalences within $[f_t^l(s), \infty)$; the larger s, the smaller f_t^l.*

Table 5.6: Subcase 1: Lower limits for the intervals $(f_t^l(s), \infty)$, on which weighting is beneficial according to Figure 5.2, given $s \geq 1$, $\sigma_1 = 0.05$, $N_2 = 100$.

***s*-values**	1	1.5	2	2.5	3	3.5	4	4.5	5
$f_t^l(s)^{(1)}$	–	–	–	3.64	1.87	1.49	1.31	1.20	1.09
$f_t^l(s)^{(2)}$	1.0001	1.0001	1.01	1.01	1.01	1.02	1.03	1.04	1.09

(1) The *worst case* reference is used (5.2.17).

(2) the *best case* reference is used (5.2.18).

Remark 5.2.4 (**Case $s \geq 1$, i.e. the error probability of the preponderant target subclass lies between the true error and the hit probability of the less prevalent target subclass. Formally, $p_{22} \in [0.05, 0.95]$**).

(i) According to Figure 5.2, a minimal degree of unbalance f_t^l of the target subclass structure is necessary in order to achieve a benefit by means of weighted error estimates (for its value see Table 5.6). This means that a minimal degree of mismatch between the target population and the data at hand is required.

(ii) Table 5.6 indicates that the larger the discrepancy u_p between the subclass error probabilities, the lower f_t^l and therefore, the larger the f_t-interval which is associated to the existence of a weighting benefit (first row from the left to the right, second row from the right to the left).

(iii) When $p_{22} \approx 1 - p_{21}$ the weighted estimates provide some benefit also when the mismatch between the true and observed subclass prevalences is rather negligible (in Table 5.6, $f_t = 1 + 10^{-4}$ for $s = 1, 1.5$).

(iv) Assume $s \geq 3$ in the first row of Table 5.6. According to (5.2.21), this corresponds to a minimal discrepancy of 13 percentage points between the subclass error probabilities ($p_{22} \geq 0.13 + p_{21}$). Then weighted estimates provide a benefit given any unbalanced target subclass structure with $\pi_{22} \geq 2\pi_{21}$ ($f_t^l \leq 1.87$).

Let $\boldsymbol{s \leq 1}$. With the settings from Note 2, this means that $p_{21} \approx 0.15$ and $p_{22} \in [0, 0.15]$ or $p_{22} \in [0.85, 1]$, depending on whether the *worst* or the *best case* reference is considered.

Given a constant value of σ_1, then every value $s < 1$ is associated to a discrepancy of:

$$100u_p = 100(p_{21} - p_{22}) = 100\left(\frac{\sqrt{1 - 4\sigma_1 s} - \sqrt{1 - 4\sigma_1}}{2}\right) \tag{5.2.22}$$

percentage points between the subclass error probabilities, if they are situated on the same side of 0.5 (*worst case* reference). This increases obviously with decreasing values of s.
If the subclass error probabilities are situated on different sides of 0.5 (*best case* reference), then the discrepancy between them is, exactly like in case $s \geq 1$, of:

$$100u_p = 100(p_{22} - p_{21}) = 100\left(\frac{\sqrt{1 - 4\sigma_1 s} + \sqrt{1 - 4\sigma_1}}{2}\right)$$

percentage points. This increases obviously with decreasing values of s.

From right to left, the p_{22} values which correspond to the first row of Table 5.7 (*worst case*) are tuned between $0.15(\approx p_{21})$ and 0. At 0 the discrepancy u_p between the subclass error probabilities achieves its maximum.
Similarly, also from right to left, the p_{22} values which correspond to the second row of Table 5.7 (*best case*) are tuned between $0.85(\approx 1 - p_{21})$ and 1, where u_p achieves its maximum.

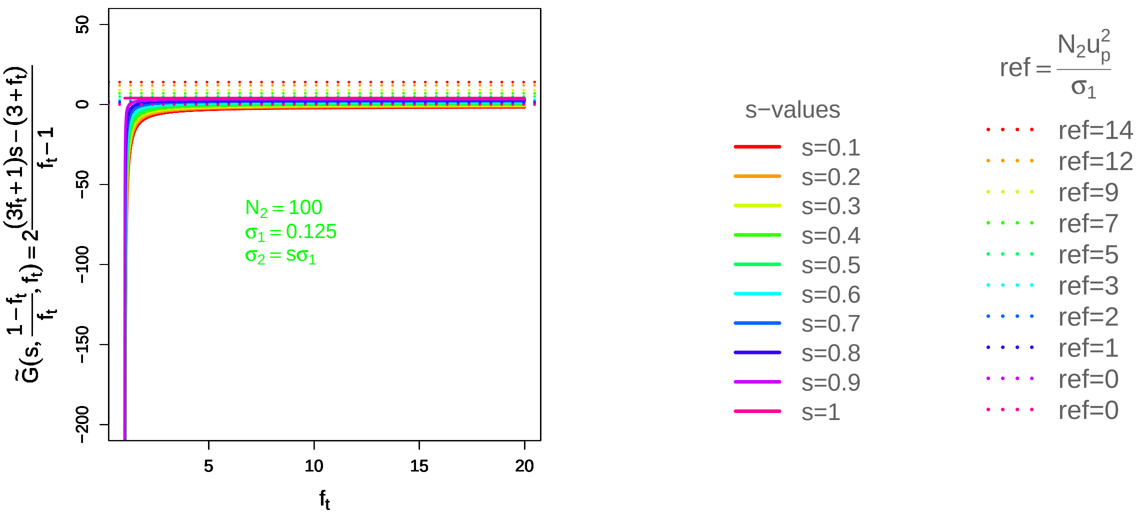


Figure 5.3: Subcase 1: Relation between $\tilde{G}(s, \frac{1-f_t}{f_t}, f_t)$ and the *worst case* reference when $s \leq 1$, $\sigma_1 = 0.125$ and $N_2 = 100$ and $f_t \leq 20$. *Given* $s \leq 0.7$, *weighting is beneficial for every quotient of the true subclass prevalences up to* 11.

Table 5.7: Subcase 1: Upper limits for the intervals $(1, f_t^h(s))$ on which weighting is beneficial given $s \leq 1$, $\sigma_1 = 0.125$, $N_2 = 100$, and $f_t \leq 20$.

***s*-values**	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
$f_t^h(s)^{(1)}$	> 20	> 20	> 20	> 20	> 20	> 20	11.21	1.85	1.25	–
$f_t^h(s)^{(2)}$	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20

(1) The *worst case* reference is used (5.2.17).

(2) The *best case* reference is used (5.2.18).

Remark 5.2.5 (**Case** $s \leq 1$, **i.e.** $p_{22} \in [0, 0.15] \bigcup [0.85, 1]$).

(i) According to Figure 5.3, the weighting benefit is possible up to a certain degree of unbalance f_t^h of the target subclass structure, or equivalently, up to a certain degree of mismatch between the target population and the data at hand.

(ii) The larger the discrepancy u_p between the subclass error probabilities, the higher the maximally allowed degree of unbalance f_t^h (see Table 5.7 from right to left).

(iii) Assume $s \leq 0.6$ and $p_{22} \in [0, 0.15]$ (first row of Table 5.7). Using (5.2.22), there is a minimal discrepancy u_p of 6.5 percentage points between the subclass error probabilities. Then, weighted estimates provide a benefit at least for any degree of unbalance f_t up to 20.

(iv) Assume $p_{22} \in [0.85, 1]$ (second row in Table 5.7). Since $p_{21} \approx 0.15$, this is equivalent to a minimal discrepancy of 70 percentage points between the subclass error probabilities. Then, weighted estimates provide a benefit at least for any degree of unbalance f_t up to 20.

5.2.4.2 Theoretical investigation

First, recall that q was used to denote the quotient $\frac{u_p^2}{\sigma_1}$ from the right hand side term of the $\tilde{G}$-condition for the existence of the weighting benefit.
Further, q^{-1} is used to denote the reciprocal or multiplicative inverse of q, thus:

$$q^{-1} = \frac{1}{q}.$$

The inverse function for $\tilde{G} : \mathcal{A} \to \mathcal{B}$ is denoted by $\tilde{G}_{inv} : \mathcal{B} \to \mathcal{A}$. Thus:

$$\tilde{G} \circ \tilde{G}_{inv} = 1_{\mathcal{B}}$$

and

$$\tilde{G}_{inv} \circ \tilde{G} = 1_{\mathcal{A}},$$

where operator '$\circ$' stands for the composition of functions, and $1_{\mathcal{A}}$ and $1_{\mathcal{B}}$ are the identity functions on $\mathcal{A}$ and $\mathcal{B}$, respectively.

Let $\boldsymbol{s > 1}$. Hence, the error probability p_{22} of the preponderant target subclass lies between the error and the hit probability of the less prevalent target subclass. Formally, $p_{22} \in (\min(p_{21}, 1 - p_{21}),\ \max(p_{21}, 1 - p_{21}))$.

Proposition 5.2.5 (Case $s > 1$)**.**

(i) If weighting is beneficial, then the size of the heterogeneous class satisfies $N_2 > (6s - 2)q^{-1}$.

(ii) If $N_2 > (6s - 2)q^{-1}$, then weighting provides a benefit beginning with a certain degree of unbalance of the target subclass structure, thus on an interval (f_t^l, ∞):

$$f_t^l(s, N_2) = \frac{N_2 q + 2s - 6}{N_2 q - 6s + 2}.$$

(iii) The larger N_2, the larger the interval associated to a beneficial weighting. The larger s, the larger the interval associated to a beneficial weighting with respect to the *worst case* reference. The lower s, the larger the interval associated to a beneficial weighting with respect to the *best case* reference.

Proof.

(i) The necessary condition for weighting to be beneficial is provided by (5.2.12), which requests that:

$$\tilde{G} < N_2 q.$$

The first derivative of $\tilde{G}$ with respect to f_t is:

$$\frac{\partial \tilde{G}}{\partial f_t} = \frac{8(1-s)}{(f_t-1)^2}. \tag{5.2.23}$$

Hence, when $s > 1$, $\tilde{G}$ is strictly monotonically decreasing with respect to f_t. It falls from:

$$\lim_{f_t \to 1} \tilde{G} = \infty$$

down to

$$\lim_{f_t \to \infty} \tilde{G} = 6s - 2 \; (> 4).$$

Thus, when $N_2 q \leq 6s - 2$, weighting yields no benefit no matter of the degree of unbalance of the target subclass structure f_t.

(ii) When $N_2 > (6s-2)q^{-1}$, due to the monotonicity of $\tilde{G}$, weighting is clearly beneficial on an interval of the form $(f_t^l(s, N_2), \infty)$. The lower bound $f_t^l(s, N_2)$ is obtained by solving the equation:

$$\tilde{G} = N_2 q$$

with respect to f_t.
Equivalently:

$$f_t^l(s, N_2) = \tilde{G}_{inv}(N_2 q), \tag{5.2.24}$$

where $\tilde{G}_{inv}$ is the inverse function for $\tilde{G}$ with respect to the argument f_t.

(iii) We use the equation (5.2.24). Since $\tilde{G}$ is monotonically decreasing with respect to f_t, also $\tilde{G}_{inv}$ is monotonically decreasing with respect to the image of f_t by $\tilde{G}$.
Besides, $\tilde{y} := N_2 q$ is strictly monotonically increasing with respect to N_2. Therefore, it holds:

$$\frac{\partial f_t^l(s, N_2)}{\partial N_2} = \underbrace{\frac{\partial \tilde{G}_{inv}}{\partial \tilde{y}}}_{<0} \underbrace{\frac{\partial \tilde{y}}{\partial N_2}}_{>0} < 0, \tag{5.2.25}$$

which means that f_t^l becomes smaller for increasing values of N_2.

Now, when the worst case *reference is considered, q refers actually to q_{min} (see notation (5.2.19)). Using Remark 5.2.2, it follows that, $\tilde{y} := N_2 q_{min}$ is strictly monotonically increasing with respect to s on* $(1, \infty)$.
This yields:

$$\frac{\partial f_t^l(s, N_2)}{\partial s} = \underbrace{\frac{\partial \tilde{G}_{inv}}{\partial \tilde{y}}}_{<0} \underbrace{\frac{\partial \tilde{y}}{\partial s}}_{>0} < 0, \tag{5.2.26}$$

which indicates that $f_t^l(s, N_2)$ is monotonically decreasing with respect to s in the context of the worst case *reference.*

When the best case *reference is considered, q refers to q_{max} (see notation (5.2.19)). Using Remark 5.2.3, $\tilde{y} := N_2 q_{max}$ is strictly monotonically decreasing with respect to s. Hence:*

$$\frac{\partial f_t^l(s, N_2)}{\partial s} = \underbrace{\frac{\partial \tilde{G}_{inv}}{\partial \tilde{y}}}_{<0} \underbrace{\frac{\partial \tilde{y}}{\partial s}}_{<0} > 0, \tag{5.2.27}$$

which indicates that $f_t^l(s, N_2)$ becomes smaller for decreasing values of s in the context of the best case *reference.*

Let $\boldsymbol{s < 1}$. Hence, the error probability of the less prevalent target subclass lies between the error and the hit probability in the preponderant target subclass. Formally, $p_{21} \in (\min(p_{22}, 1 - p_{22}),\ \max(p_{22}, 1 - p_{22}))$.

Proposition 5.2.6 (Case $s < 1$)**.**

(i) A sufficient condition for a beneficial weighting given any degree of unbalance f_t of the target subclass structure is that:

$$s \leq 0.333.$$

(ii) If $N_2 > 4q^{-1}$, then the weighting benefit is sure for all $f_t > 1$.

(iii) If $N_2 > (6s - 2)q^{-1}$, then the weighting benefit is sure up to a certain discrepancy between the true subclass prevalences:

$$f_t^h(s, N_2) = \frac{N_2 q + 2s - 6}{N_2 q - 6s + 2}.$$

The smaller s and the larger N_2, the larger $f_t^h(s, N_2)$.

Proof.

(i) The derivative of $\tilde{G}$ with respect to f_t from (5.2.23) indicates that, when $s < 1$, $\tilde{G}$ is strictly monotonically increasing with respect to f_t between:

$$\lim_{f_t \to 1} \tilde{G} = -\infty, \tag{5.2.28}$$

and

$$\lim_{f_t \to \infty} \tilde{G} = 6s - 2. \tag{5.2.29}$$

If $s \leq 0.333$, then $6s - 2$ belongs to $(-2, 0]$. Thus $\tilde{G} < 0$ all-over and therefore, the weighting benefit is sure independently of f_t and N_2.

(ii) If $s > 0.333$, the weighting benefit is still sure for all $f_t > 1$, if $N_2 > (6s - 2)q^{-1}$, where the latter inequation term is maximally equal to $4q^{-1}$.

(iii) According to Remarks 5.2.2 and 5.2.3, both the worst *and the* best case *references are strictly monotonically decreasing with respect to $s < 1$. Using the ascending monotonicity of $\tilde{G}_{inv}$ and the descending monotonicity of $\tilde{y} = N_2 q$ with respect to s, it holds:*

$$\frac{\partial f_t^h(s, N_2)}{\partial s} = \underbrace{\frac{\partial \tilde{G}_{inv}}{\partial \tilde{y}}}_{>0} \underbrace{\frac{\partial \tilde{y}}{\partial s}}_{<0} < 0,$$

which means that f_t^h is monotonically decreasing with respect to s. Its monotonicity with respect to N_2 is proved like in Proposition 5.2.5.

Corollary 5.2.7 (Case $s < 1$)**.**

(i) Assume $N_2 \geq 24$ observations in the heterogeneous class and the error probability of the less prevalent target subclass, p_{21}, lies outside of the range of medium values $(0.4, 0.6)$. Then the weighting benefit is sure for every pair of subclass error probabilities (p_{21}, p_{22}) separated by 0.5.

(ii) Assume $N_2 \geq 100$ observations in the heterogeneous class. Then the weighting benefit is sure for all pairs of subclass error probabilities which differ by at least 10 percentage points, thus for $|u_p| = |p_{22} - p_{21}| > 0.1$.

Proof.

(i) From (5.2.28) and (5.2.29) it results that the weighting benefit is sure for all $f_t > 1$ if:

$$(6s - 2)q^{-1} < N_2. \tag{$*$}$$

When the subclass error probabilities lie on different sides of 0.5*, q refers to the* best case *reference, and therefore $q = q_{max}$.*
The condition $p_{21} \notin (0.4, 0.6)$ indicates that the variance $\sigma_1 = p_{21}(1 - p_{21})$ of the error event in the less prevalent target subclass is at most 0.24*.*
Using the monotonicity of q_{max} with respect to σ_1 stated in Remark 5.2.3, it follows that:

$$q_{max}^{-1}(\sigma_1, s) \leq q_{max}^{-1}(0.24, 1) = 6, \qquad (**)$$

for every σ_1 up to 0.24*.*
Besides, for $s < 1$, it holds:

$$6s - 2 < 4. \qquad (***)$$

From $(*)$ - $(***)$ it results:

$$(6s - 2)q_{max}^{-1} < 24.$$

This indicates that in this context the weighting benefit is sure for every size $N_2 \geq 24$ of the heterogeneous class.

(ii) If $|p_{22} - p_{21}| > 0.1$, then it holds:

$$q^{-1} = \frac{\sigma_1}{|p_{22} - p_{21}|^2} \leq \frac{0.25}{0.01} = 25.$$

Thus, using point (ii) from Proposition 5.2.6 the benefit is sure for every $N_2 \geq 100$ since:

$$(6s - 2)q^{-1} < 4q^{-1} \leq 100.$$

Proposition 5.2.8 (Case $s = 1$)**.**

(i) $\tilde{G}$ is constant and equal to 4.

(ii) No benefit is achievable by weighting when the subclass error probabilities lie on the same side of 0.5, i.e. with respect to the *worst case* reference.

(iii) When the subclass error probabilities are situated on different sides of 0.5, i.e. when the *best case* reference is considered, weighting provides a benefit if and only if:

$$N_2 > \frac{4\sigma_1}{1 - 4\sigma_1}.$$

In this case, if $p_{21} \notin (0.4, 0.6)$, then the weighting benefit is sure starting already from $N_2 = 25$ observations per class.

Proof.

(i) It results easily replacing $s = 1$ in (5.2.20).

(ii) The necessary condition for the existence of a benefit by weighting (5.2.12) becomes:

$$N_2 q > 4.$$

When the worst case reference is considered, q refers to q_{min}, which is zero under $s = 1$. Therefore, the benefit condition is never accomplished.

(iii) When the best case reference is considered, q refers to q_{max}. Replacing $s = 1$ in (5.2.18) yields:

$$q_{max} = \frac{1 - 4\sigma_1}{\sigma_1}.$$

If $p_{21} \notin (0.4, 0.6)$, then $\sigma_1 \leq 0.24$ and the minimally necessary N_2 is straightforward to compute.

5.2.4.3 Summary

The situation when $\tilde{\pi}_{21} = \tilde{\pi}_{22} = 0.5$ is regarded. This subcase is first graphically investigated starting from a fixed value of σ_1 (i.e. from a fixed p_{21}), and tuning s below or above 1 (i.e. varying p_{22}). Then a general theoretical investigation is used to justify and generalize the graphical approach. Without loss of generality, the error probability p_{21} in the less prevalent target subclass is assumed to be less than 0.5.

The reader is advised of the possibility to extend the results from Tables 5.6 and 5.7 also to other values of p_{21}, using Proposition 5.2.3 and Corollary 5.2.4.

In general, both graphical and theoretical investigations indicate that:

- The larger the discrepancy $|p_{21} - p_{22}|$ between the subclass error probabilities is, the larger is also the range of the degree of mismatch between the target population and the data at hand for which weighting provides a benefit (i.e. the f_t-interval associated to the weighting benefit).
- If $p_{22} \in (p_{21}, 1 - p_{21})$, then a minimal degree of unbalance in the target population should be available for a beneficial weighting. Too low degrees of unbalance in the target population (e.g. $f_t < 2$) may be dangerous especially when $p_{21} < p_{22} \ll 0.5$.
- If $p_{22} \in [0, p_{21})$, then starting from a certain degree of unbalance of the target subclass structure weighted estimates are inefficient in comparison to the unweighted ones. However, a highly unbalanced target subclass structure (e.g. $f_t \geq 2$) is dangerous only when the subclass error probabilities are pretty similar.

Proposition 5.2.5 supports theoretically Remark 5.2.4 when $s > 1$, i.e. when $p_{22} \in (p_{21}, 1-p_{21})$. For instance, the statement from point (ii) confirms Remark 5.2.4 (i), according to which a minimal degree of unbalance of the target subclass structure is required for a beneficial weighting;

the statement from point (iii) confirms Remark 5.2.4 (ii), since if $s > 1$, the discrepancy between the subclass error probabilities becomes larger for increasing values of s given the *worst case* reference, and for decreasing values of s, given the *best case* reference.

Proposition 5.2.6 supports theoretically Remark 5.2.5 when $s < 1$, i.e. when p_{22} is either an element of $[0, p_{21})$ or of $(1 - p_{21}, 1]$. The statement from point (iii) matches Remark 5.2.5(i) according to which weighting is beneficial up to a maximal degree of unbalance of the target subclass structure; it also matches Remark 5.2.5(ii), since if $s < 1$, the discrepancy between the subclass error probabilities increases for decreasing values of s.

The strongest result is provided by Corollary 5.2.7 to Proposition 5.2.6, when $s < 1$.
First, this states that, if $p_{21} \leq 0.4$ and $p_{22} > 1 - p_{21}$, the weighting benefit is sure already for a size $N_2 = 24$ of the heterogeneous class.
Second, when $N_2 \geq 100$, weighting is beneficial given any pair of subclass error probabilities p_{21} and p_{22} which differ by at least 10 percentage points. This result is practically confirms the practical findings from Remark 5.2.5(iii) and (iv).

Also, weighted estimates are always superior to unweighted ones given $s \leq 0.333$. This upper bound corresponds to a certain minimal discrepancy between the subclass misclassification probabilities, which can be easily computed for any particular value of p_{21} by the formula (5.2.22).

The larger the class sample size N_2, the larger the f_t-intervals associated to a weighting benefit.

5.2.5 Subcase 2

The data at hand is in this case highly suboptimal with respect to the target subclass prevalence structure. The quotient of the observed subclass prevalences is $f_d = \frac{1}{bf_t}$, with $b \in [1, \infty)$. For instance, while in the target population one subclass (here, C_2) represents 90% of the heterogeneous class, the same subclass appears at most in 10% of the samples of the heterogeneous class in the data set at hand.

The relative difference between the true and observed subclass prevalence structures ρ^* has the expression:

$$\rho^* = \frac{1 - bf_t^2}{bf_t^2},$$

in terms of the quotient of the true subclass prevalences, f_t and the degree of diametral opposition, b.

We replace ρ^* in the $\tilde{G}$-condition (5.2.12) by its representation in terms of b and f_t. The weighting benefit is available if the condition:

$$\underbrace{\left(\frac{1+bf_t}{-1+bf_t^2}\right)\left[(1+2f_t+bf_t^2)s-\frac{1+2bf_t+bf_t^2}{bf_t}\right]}_{=\tilde{G}(s,\frac{1-bf_t^2}{bf_t^2},f_t)} < \frac{N_2 u_p^2}{\sigma_1}, \tag{5.2.30}$$

is accomplished.

5.2.5.1 Graphical investigation

For a fixed value of s, b is varied between 1 (diametrally opposite observed subclass prevalences with respect to the true ones) and 9 (the preponderant target subclass is extremely undersampled). The quotient of the true subclass prevalences f_t is tuned over $[1.1, 20]$ in steps of 0.01. Again, N_2 and σ_1 are chosen like established in Note 2.

Figures 5.4 and 5.5 illustrate $\tilde{G}$ (solid line) with respect to the *worst case* reference (dotted line of same color) when $s \geq 1$ and $s \leq 1$, respectively. The same plots with respect to the *best case* reference can be viewed in Appendix B, in Figures B.3 and B.4.
Table 5.8 ($s \geq 1$) presents the upper and lower limits of the f_t-intervals associated to a weighting benefit with respect to the *worst case* reference in Figure 5.4 when $b = 1$ and $b = 9$ (rows 1, 2 and 5, 6, respectively). It presents also the upper and lower limits of the f_t-intervals associated to a weighting benefit in Figure B.3 where the *best case* reference is used (rows 3, 4 and 7, 8, respectively).
Table 5.9 ($s \leq 1$) presents the upper limits of the f_t-intervals associated to a weighting benefit with respect to the *worst case* (Figure 5.5) and the *best case* reference (Figure B.4) when $b = 1$ and $b = 9$.

Let $\boldsymbol{s \geq 1}$. With the settings from Note 2, this means that $p_{21} \approx 0.05$ and $p_{22} \in [0.05, 0.5]$ or $p_{22} \in [0.5, 0.95]$, depending on whether the *worst* or the *best case* reference is considered.
Recall the interpretation of the s-values from Table 5.6. This applies also to Table 5.8. The discrepancy u_p between the subclass error probabilities increases from left to the right in Table 5.8, thus for increasing values of s, when the *worst case* reference is considered. However, when the *best case* reference is considered, u_p increases from right to the left in Table 5.8, thus for decreasing values of s.

Remark 5.2.6 (Case $s \geq 1$, i.e. the error probability of the preponderant target subclass

lies between the true error and the hit probability of the less prevalent target subclass. Formally, $p_{22} \in [0.05, 0.95]$).

(i) Figure 5.4 indicates that a benefit by weighted error estimates is possible only for a degree of target unbalance within some certain range (f_t^l, f_t^h). This means that a too low or too high degree of unbalance in the target population may be dangerous in this context.

(ii) According to Table 5.8, the larger the degree of diametral opposition b between the true and observed subclass prevalences, the shorter the range of the degree of target unbalance which is favorable for weighting (f_t^h gets smaller).

(iii) The larger the discrepancy u_p between the subclass error probabilities, the lower f_t^l and the higher f_t^h, thus the larger the f_t-interval which is associated to the existence of a weighting benefit.

(iv) When $p_{22} \in [0.5, 0.95]$, thus p_{22} lies on the other side of 0.5 with respect to p_{21}, weighting is beneficial for any degree of diametral opposition $b \leq 9$ at least up to a degree of unbalance $f_t = 8.55$ (see last row in Table 5.8).

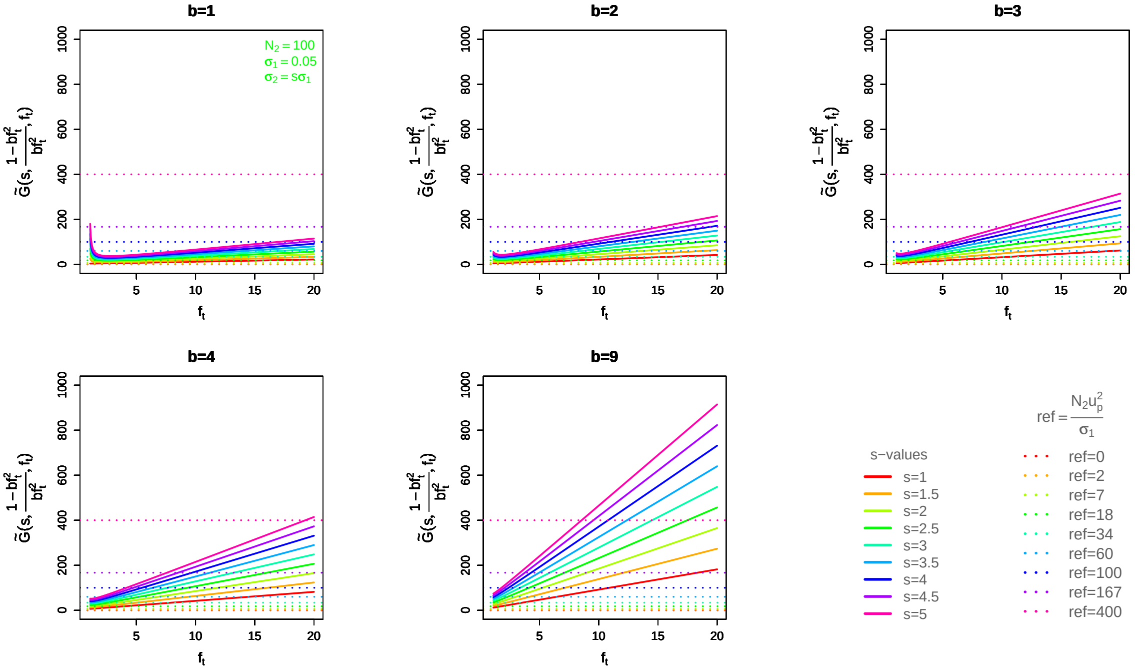


Figure 5.4: Subcase 2: Relation between $\tilde{G}$ and the *worst case* reference when $s \geq 1$, $\sigma_1 = 0.05$, $N_2 = 100$, and $f_t \leq 20$. *The larger b, or the larger f_t, or the smaller s, the less chances for a weighting benefit; when $b = 1$, the benefit is sure for $s \geq 3$ and f_t within $(f_t^l(s), f_t^h(s))$; this interval gets larger for increasing values of s.*

Table 5.8: Subcase 2: Limits of the f_t intervals associated to a weighting benefit when $s \geq 1$. *Weighting provides a benefit at least for a degree of target unbalance f_t within some certain interval* $(f_t^l(s), f_t^h(s))$.

	***s*-values**	1	1.5	2	2.5	3	3.5	4	4.5	5
$\boldsymbol{b} = 1$	$f_t^l(s)^{(1)}$	–	–	–	2.16	1.38	1.22	1.15	1.1	1.1
	$f_t^h(s)^{(1)}$	–	–	–	3.20	8.36	14.22	> 20	> 20	> 20
	$f_t^l(s)^{(2)}$	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1
	$f_t^h(s)^{(2)}$	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20
$\boldsymbol{b} = 9$	$f_t^l(s)^{(1)}$	–	–	–	–	–	1.1	1.1	1.1	1.1
	$f_t^h(s)^{(1)}$	–	–	–	–	–	1.45	2.38	3.76	8.55
	$f_t^l(s)^{(2)}$	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1
	$f_t^h(s)^{(2)}$	> 20	> 20	> 20	> 20	> 20	> 20	> 20	17.77	8.55

(1) The *worst case* reference is used (5.2.17).

(2) The *best case* reference is used (5.2.18).

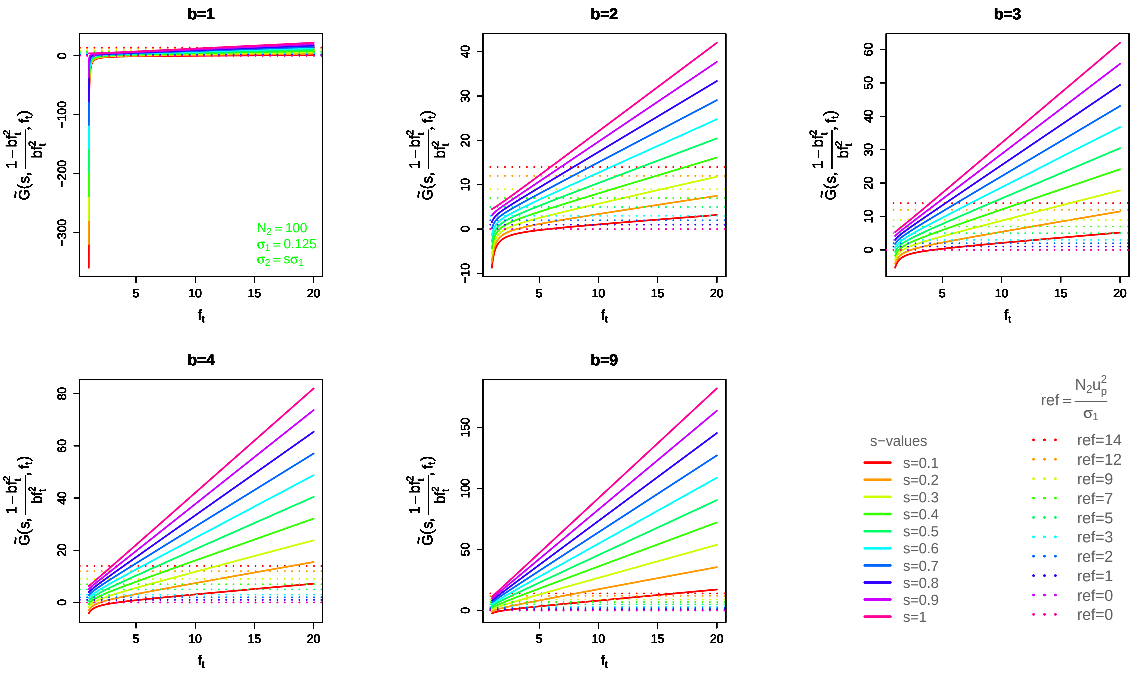


Figure 5.5: Subcase 2: Relation between $\tilde{G}$ and the *worst case* reference when $s \leq 1$, $\sigma_1 = 0.125$, $N_2 = 100$, and $f_t \leq 20$. *The larger b, or the larger f_t, or the larger s, the less chances for a weighting benefit; a benefit, if any, is obtained on some interval of the form* $(1, f_t^h(s))$.

Table 5.9: Subcase 2: Upper limits of the f_t-intervals associated to a beneficial weighting when $s \leq 1$. *Weighting provides a benefit given a degree of target unbalance f_t within some interval $(1, f_t^h(s))$.*

	s-values	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
b = 1	$f_t^h(s)^{(1)}$	> 20	> 20	> 20	17.26	9.36	4.63	2.10	1.34	1.11	–
	$f_t^h(s)^{(2)}$	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20
b = 9	$f_t^h(s)^{(1)}$	16.81	6.84	3.61	2.07	1.24	–	–	–	–	–
	$f_t^h(s)^{(2)}$	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20	> 20

$^{(1)}$ The *worst case* reference is used (5.2.17).

$^{(2)}$ the *best case* reference is used (5.2.18).

Let $\boldsymbol{s \leq 1}$. With the settings from Note 2, this means that $p_{21} \approx 0.15$ and $p_{22} \in [0, 0.15]$ or $p_{22} \in [0.85, 1]$, depending on whether the *worst* or the *best case* reference is considered.
Recall the interpretation of the s-values from Table 5.7. This applies also to Table 5.9. The discrepancy u_p between the subclass error probabilities increases with decreasing values of s, both when the *worst* and the *best case* reference is considered.

Remark 5.2.7 (Case $s \leq 1$, i.e. $p_{22} \in [0, 0.15] \bigcup [0.85, 1]$).

(i) Figure 5.5 indicates that the weighting benefit is possible up to a certain degree of unbalance f_t^h of the target subclass structure. When $p_{22} \in [0, 0.15]$, a high degree of target unbalance may be dangerous.

(ii) According to Table 5.9, the larger the degree of diametral opposition b between the true and observed subclass prevalences, the larger the minimal discrepancy u_p between the subclass error probabilities which is necessary for the existence of a weighting benefit.

(iii) The larger b and the smaller u_p, the smaller the maximal degree of target unbalance f_t^h up to which weighting provides a benefit.

5.2.5.2 Theoretical investigation

Proposition 5.2.9 (Case $s > 1$ and $b = 1$)**.**

(i) A necessary condition for a beneficial weighting is that $N_2 > \tilde{G}(f_t^0)q^{-1}$; for $s \geq 1.5$, f_t^0 is some number within $(2, 3)$ depending on s.

(ii) If condition (i) is accomplished, then the weighting benefit is sure for $f_t \in (f_t^l, f_t^h) \subset (1, \infty)$. The larger N_2, the larger the f_t-interval associated to a beneficial weighting.

(iii) When the *worst case* reference is considered, the larger $s > 1$ (i.e. the larger the gap between the subclass error probabilities), the larger the f_t-interval associated to a sure weighting benefit.

When the best case reference is considered, the smaller $s > 1$ (i.e. the larger the gap between the subclass error probabilities), the larger the interval associated to a sure weighting benefit.

Proof.

(i) Using the expression of $\tilde{G}$ in (5.2.30) it is clear that given $s > 1$, $b = 1$, and $f_t > 1$ this function takes only positive values ($\tilde{G} > 0$).
The first derivative of $\tilde{G}$ with respect to f_t is:

$$\frac{\partial \tilde{G}}{\partial f_t} = \frac{1}{f_t^2(-1+f_t)^2} \cdot \left[\underbrace{-1 + 2f_t + 3(1-s)f_t^2 - 2sf_t^3 + sf_t^4}_{:=P(s,f_t)} \right].$$

If $s > 1$, then $P(s, f_t)$ has just one zero within $(1, \infty)$, $f_t^0(s)$. This can be easily proved using the sign and monotonicity of the first and second derivatives of P with respect to its second argument.
Moreover P and therefore also the derivative of $\tilde{G}$ with respect to f_t has a negative sign to the left and a positive sign to the right of $f_t^0(s)$. This indicates that $\tilde{G}$ has a minimum at $\tilde{G}(s, 1, f_t^0(s)) > 0$, while it is strictly monotonically decreasing to the left and strictly monotonically increasing to the right of it. It holds:

$$\lim_{f_t \to 1} \tilde{G}(s, 1, f_t) = \infty$$

and

$$\lim_{f_t \to \infty} \tilde{G}(s, 1, f_t) = \infty.$$

Hence, using (5.2.12), a necessary condition for the weighting benefit to exist even on a small neighborhood around f_t^0 is that:

$$N_2 > \tilde{G}(s, 1, f_t^0(s))q^{-1}.$$

By solving $P(s, f_t^0(s)) = 0$ with respect to f_t^0, it results:

$$f_t^0(1.5) = 2.24 > 2$$

and

$$\lim_{s \to \infty} f_t^0(s) = 3.027 \ (\approx 3).$$

(ii) When the condition in (i) is accomplished, the weighting benefit is restricted to values of f_t within some interval $(f_t^l(s, N_2), f_t^h(s, N_2))$ around f_t^0, whose lower and upper limits are obtained by solving the equation:

$$\tilde{G}(s, 1, f_t) = N_2 q$$

with respect to f_t.

We denote by $\tilde{G}_{inv|(1,f_t^0)}$ the restriction of the inverse function for $\tilde{G}$ on $(1, f_t^0)$ and by $\tilde{G}_{inv|(f_t^0,\infty)}$ its restriction on (f_t^0, ∞). Then f_t^l and f_t^h are given by:

$$f_t^l = \tilde{G}_{inv|(1,f_t^0)} \circ (N_2 q)$$

and

$$f_t^h = \tilde{G}_{inv|(f_t^0,\infty)} \circ (N_2 q).$$

Now, due the monotonicity of $\tilde{G}$ below and above f_t^0, which is proved at point (i), $\tilde{G}_{inv|(1,f_t^0)}$ is a strictly monotonically decreasing, while $\tilde{G}_{inv|(f_t^0,\infty)}$ a strictly monotonically increasing function. Besides, $N_2 q$ is monotonically increasing with respect to N_2. This indicates, in a similar way to (5.2.25), that f_t^l is monotonically decreasing, while f_t^h is monotonically increasing with respect to N_2. Thus, the larger N_2, the larger the f_t-interval associated to a beneficial weighting.

(iii) *When the* worst case *reference is considered, q refers to q_{min} (see Notation 5.2.19). Then, $N_2 q_{min}$ is strictly monotonically increasing with respect to $s > 1$. Similarly to (5.2.26) it results that $\frac{\partial f_t{}^l}{\partial s} < 0$ and $\frac{\partial f_t{}^h}{\partial s} > 0$, thus the lower limit of the beneficial f_t-interval is strictly monotonically decreasing, while the upper limit is strictly monotonically increasing with respect to $s > 1$.*

When the best case *reference is considered, q refers to q_{max} (see Notation 5.2.19). Then, $N_2 q_{max}$ is strictly monotonically decreasing with respect to $s > 1$. Similarly to (5.2.27) it results that $\frac{\partial f_t{}^l}{\partial s} > 0$ and $\frac{\partial f_t{}^h}{\partial s} < 0$, thus the beneficial f_t-interval becomes larger for decreasing values of $s > 1$, when the* best case *reference is considered.*

Proposition 5.2.10 (Case $s < 1$)**.**

(i) A sufficient condition for a beneficial weighting is:

$$s \leq s_0(b, f_t) := \frac{1}{bf_t} \cdot \left(\frac{1 + 2bf_t + bf_t^2}{1 + 2f_t + bf_t^2} \right) \;(< 1),$$

which means also that a minimal discrepancy between the subclass error probabilities is required.

(ii) Given a fixed value of $s < 1$, the weighting benefit is sure for every b within some interval $[1, b^+(s))$ up to a certain degree of unbalance of the target subclass structure, thus for any f_t

within $(1, f_t^h(s,b)]$, where:

$$b^+(s) = \frac{3(1-s) + \sqrt{9(1-s)^2 + 4s}}{2s}. \tag{5.2.31}$$

For every $b \in [1, b^+(s))$, the maximal degree of unbalance of the target subclass structure for which the weighting benefit still exists is the only solution of the equation:

$$sb^2 f_t^3 + (2s-1)bf_t^2 + (s-2)bf_t - 1 = 0 \tag{5.2.32}$$

which is greater than 1.

(iii) The lower $s < 1$, or equivalently, the larger the gap between the subclass error probabilities, the larger also the intervals $[1, b^+(s))$ and $(1, f_t^h(s,b)]$ associated to a sure weighting benefit.

Proof.

(i) The weighting benefit is sure when the difference in brackets in condition (5.2.30) is negative. This is equivalent to the request that s is smaller than $s_0(b, f_t)$, which lies below 1, $\forall b \geq 1$, $\forall f_t > 1$. When $s < 1$, the upper limit s_0 imposed on s means also that a minimal discrepancy between the subclass error probabilities is required for a beneficial weighting.

(ii) Condition (i) is equivalent to:

$$P(s,b,f_t) =: sb^2 f_t^3 + (2s-1)bf_t^2 + (s-2)bf_t - 1 \leq 0.$$

The first derivative in terms of f_t:

$$\frac{\partial P(s,b,f_t)}{\partial f_t} = 3sb^2 f_t^2 + 2(2s-1)bf_t + (s-2)b$$

has always two zeros:

$$f_t^{-,+}(s,b) = \frac{1 - 2s \mp \sqrt{(1+s)^2 + 3s(2-s)(b-1)}}{3sb},$$

where $f_t^- < 0$ and $f_t^+ > 0$.

When $f_t^+ \leq 1$, $P(s,b,f_t)$ is strictly monotonically increasing with respect to f_t over $(1,\infty)$. Otherwise, $P(s,b,f_t)$ is strictly monotonically decreasing on $(1, f_t^+]$ and strictly monotonically increasing on (f_t^+, ∞).

It holds

$$\lim_{f_t \to \infty} P(s,b,f_t) = \infty. \qquad (*)$$

When $f_t \to 1$,

$$P(s, b, f_t) \to sb^2 + 3(s-1)b - 1$$

which has two zeros at

$$b^{-,+}(s) = \frac{3(1-s) \mp \sqrt{9(1-s)^2 + 4s}}{2s}.$$

When $s < 1$, *it holds* $b^{-}(s) < 0$ *and* $b^{+}(s) > 1$.

Since $b^{+}(s) > 1$, *it holds* $\forall b \in [1, b^{+}(s))$:

$$P(s, b, 1) < 0. \qquad (**)$$

Using $(*)$ *and* $(**)$, *and the monotonicity of* P, *it results that* $P(s, b, f_t)$ *has one zero* $f_t^h(s, b)$ *on* $(1, \infty)$, $\forall b \in [1, b^{+}(s))$. *This means that* $P(s, b, f_t) \leq 0 \; \forall f_t \in (1, f_t^h(s, b)]$.

(iii) Obviously, $P(s, b, f_t)$ *is monotonically increasing in* s. *We split the analysis into two cases.*

(1. Case) $f_t^{+} \leq 1$

For every arbitrary $b \in [1, b^{+}(s))$, P *is monotonically increasing with respect to* s *and* f_t. *Assume that for* $s < s'$ *it would hold* $f_t^h(s, b) < f_t^h(s', b)$. *This would lead to:*

$$0 = P(s, b, f_t^h(s, b)) < P(s', b, f_t^h(s, b)) < P(s', b, f_t^h(s', b)) = 0,$$

which is a contradiction. Thus, f_t^h *should be monotonically decreasing with respect to* s.

(2. Case) $f_t^{+} > 1$

For every arbitrary $b \in [1, b^{+}(s))$,

$$P(s, b, f_t^{+}(s, b)) < P(s, b, 1) < 0.$$

Thus, $f_t^h(s, b) \in (f_t^{+}(s, b), \infty)$, *on which* P *is strictly monotonically increasing. From here applies the rationale of the first case.*

Besides, it holds:

$$b^{+}(s) = y(s) + \sqrt{y(s)^2 + \frac{1}{s}}$$

and

$$y(s) = \frac{3(1-s)}{2s}.$$

Since y and $\frac{1}{s}$ are strictly monotonically decreasing functions in s, and $y > 0$ for $s < 1$, it results easily that b^+ is also strictly monotonically decreasing with respect to $s < 1$. Thus, the smaller $s < 1$, the larger also the interval $[1, b^+(s))$, for which the weighting benefit is sure up to some degree of unbalance of the target subclass structure $f_t^h(s, b)$.

Corollary 5.2.11 (Case $s < 1$)**.**

(i) In particular, when $b = 1$, the weighting benefit is sure if the condition:

$$s \leq \frac{1}{f_t}$$

is accomplished. Equivalently, the weighting benefit is sure for every degree f_t of unbalance of the target subclass structure within an interval $(1, \frac{1}{s}]$.

(ii) Very small values of s, e.g. $s \leq 0.2$, are especially favorable for the existence of a weighting benefit.

Proof.

(i) Clear from (i) in Proposition 5.2.10.

(ii) It holds:

$$\lim_{s \to 0} b^+(s) = \infty$$

and

$$\lim_{s \to 1} b^+(s) = 1.$$

For instance, if $s = 0.2$, then $b^+ \approx 12.4$; if $s = 0.5$, then $b^+ \approx 3.56$; if $s = 0.9$, then $b^+ \approx 1.23$.

Proposition 5.2.12 (Case $s < 1$ and $b = 1$)**.** The weighting benefit is sure up to some certain degree of unbalance of the target subclass structure, which depends on s and on the size N_2 of the heterogeneous class, i.e. for $f_t \in (1, f_t^h(s, N_2))$. The lower s (i.e. the larger the gap between the subclass error probabilities) and the larger N_2, the larger f_t^h and, therefore, the larger the interval associated to a sure weighting benefit.

Proof. *Using (5.2.12), the weighting benefit is generally provided if*

$$\tilde{G} < N_2 q.$$

$\tilde{G}$ is continuous and strictly monotonically increasing with respect to f_t on $(1, \infty)$ between

$$\lim_{f_t \to 1} \tilde{G}(s, 1, f_t) = -\infty$$

and

$$\lim_{f_t \to \infty} \tilde{G}(s, 1, f_t) = \infty.$$

Hence, $\tilde{G}$ has exactly one zero $f_t^0(s)$ within $(1, \infty)$ as well as a negative sign to the left and a positive sign to the right of it, respectively. This indicates that, when $b = 1$, the weighting benefit is sure given some fixed $s < 1$ at least for every $f_t \in (1, f_t^0(s)]$.

Actually, for a given size N_2 of the heterogeneous class, the weighting benefit is provided on a larger interval $(1, f_t^h(s, N_2))$, where $f_t^h(s, N_2) > f_t^0(s)$ is the solution of

$$\tilde{G}(s, 1, f_t) = N_2 q.$$

Therefore,

$$f_t^h(s, N_2) = \tilde{G}_{inv}(N_2 q).$$

When $s < 1$ is fixed, since $\tilde{G}$ is strictly monotonically increasing with respect to f_t, $\tilde{G}_{inv}$ is strictly monotonically increasing, too. Thus, the larger N_2, the larger $N_2 q$ and therefore the larger $\tilde{G}_{inv}(N_2 q) = f_t^h(s, N_2)$, too.
Also, according to Remarks (5.2.2) and (5.2.3), q is strictly monotonically decreasing with respect to $s < 1$. Thus, $N_2 q$ is monotonically decreasing with respect to s and therefore, $\tilde{G}_{inv}(N_2 q) = f_t^h(s, N_2)$ is strictly monotonically decreasing with respect to s. Thus, the smaller $s < 1$, the larger $f_t^h(s, N_2)$ and therefore, the larger the f_t-interval associated to a weighting benefit.

5.2.5.3 Summary

The situation when $\hat{\pi}_{22} \leq \pi_{21}$ is regarded. This subcase is first graphically investigated starting from a fixed value of σ_1 (i.e. from a fixed p_{21}), and tuning s below or above 1 (i.e. varying p_{22}). Then a general theoretical investigation is used to justify and generalize the graphical approach. Without loss of generality, the error probability p_{21} in the less prevalent target subclass is assumed to be less than 0.5.

Also here, the results from Tables 5.8 and 5.9 can be extended to other values of p_{21}, using Proposition 5.2.3 and Corollary 5.2.4.

In general, both graphical and theoretical investigations indicate that:

- A necessary condition for a beneficial weighting is a minimal discrepancy $|p_{21} - p_{22}|$ between the subclass misclassification probabilities; the larger the degree of diametral opposition b between the true and observed subclass structures, the larger also the minimally required discrepancy between p_{21} and p_{22}.
- Given some certain discrepancy between the subclass error probabilities, the smaller $\hat{\pi}_{22}$ in comparison with π_{21}, the smaller the favorable f_t-range and therefore, the less chances for a weighting benefit.
- Given a fixed degree of diametral opposition b, the larger the discrepancy u_p between the subclass error probabilities, the larger the range of the degree of target unbalance f_t for which weighted estimates are better than unweighted ones.
- If $p_{22} \in (p_{21}, 1 - p_{21})$, then a benefit by weighted error estimates is possible only for a degree of target unbalance within some certain interval. Too low and too high values of f_t may be dangerous especially when $p_{21} < p_{22} \ll 0.5$ and $\hat{\pi}_{22} \ll \pi_{21}$.
- If $p_{22} \in [0, p_{21})$, then some weighting benefit is possible only up to a certain degree of target unbalance. A high unbalance of the target subclass structure ($f_t \geq 2$) may be dangerous especially when $\hat{\pi}_{22} \ll \pi_{21}$.

Proposition 5.2.9 supports theoretically Remark 5.2.6 when $s > 1$, i.e. when $p_{22} \in (p_{21}, 1 - p_{21})$. The statement from point (ii) confirms Remark 5.2.6(i), according to which weighting is beneficial only when the degree of unbalance of the target subclass structure belongs to a certain range (f_t^l, f_t^h); the statement from point (iii) confirms Remark 5.2.6 (iii), which observes that this range becomes larger when the difference between the subclass error probabilities increases. Too low or too high degrees of target unbalance are obviously unrecommended.

Proposition 5.2.12 supports theoretically Remark 5.2.7 (i) in the context of diametrally opposite observed unbalanced subclasses ($b = 1$, i.e. $\hat{\pi}_{22} = \pi_{21}$), when $s < 1$. In this case p_{22} is either an element of $[0, p_{21})$ or of $(1 - p_{21}, 1]$. Accordingly, the weighting benefit is possible up to some certain degree of unbalance of the target subclass structure. This gets lower for increasing values of s, i.e. for decreasing values of the distance $|p_{22} - p_{21}|$ between the subclass error probabilities. Consequently, weighting may be dangerous given a highly unbalanced target subclass structure and almost equal subclass error probabilities.

Proposition 5.2.10(i) confirms theoretically Remark 5.2.7(ii), according to which, a minimal gap between the subclass error probabilities is required for a beneficial weighting; this gets larger for increasing values of the degree of diametral opposition b between the true and observed subclass structures. Proposition 5.2.10(ii) provides a computation scheme for the b

(5.2.31) and f_t (5.2.32)-domains associated to a sure weighting benefit starting from an estimation of s.

Like the Table 5.9 indicates, the most profitable situations for weighting when $s < 1$ are those characterized by a small error probability p_{21} (e.g. $p_{21} \leq 0.15$) and an error probability $p_{22} \geq 1 - p_{21}$. Given at least 100 observations per class, the benefit is sure in this case up to a degree of unbalance of the target subclass structure of $f_t = 20$ and a degree of diametral opposition between the true and observed subclass structures of $b = 9$.

Corollary 5.2.11 shows that, when $s < 1$, weighted estimates are always superior to the unweighted ones in the context of diametrally opposite observed unbalanced subclasses ($b = 1$) given a relation of at least inverse proportionality between s and f_t, i.e. $s \leq \frac{1}{f_t}$. Besides, values of $s \leq 0.2$ are especially favorable, since a weighting benefit is possible in this case up to a degree of diametral opposition $b \approx 12$.

The larger the class sample size N_2, the larger the b- and f_t-intervals associated to a weighting benefit.

5.3 Weighted versus unweighted class means

The simulation studies from Chapter 4 show the superiority of weighting methods which account for the true subclass prevalences not only in the process of rule validation and optimization, but of rule building as well. In the case of a non-representative data at hand, when observed and true subclass prevalences are clearly different, such methods (IIC and CCC) achieve less biased performance estimates than methods which weight only on validation data sets (AAC and ACC). They compute weighted parameter estimates for the heterogeneous class distributions starting from the plug-in estimates of these parameters in the subclasses and using the true subclass prevalences as weights.

The weighted mean estimate of the heterogeneous class aims at a correction of the unweighted estimate for the target data situation. In this section we focus on the impact of weighting on the univariate mean estimate of the heterogeneous class. By *univariate* we refer to some candidate feature X for the classification task.

The reader is reminded of the conventions adopted in Section 5.1. Accordingly, the quotient of the true subclass prevalences f_t is larger than 1, which is equivalent to $\pi_{22} > \pi_{21}$. Therefore, the target subclass structure is unbalanced and subclass C_2 is always the preponderant target subclass. Also, the degree of suboptimality of the data set at hand ρ^* is assumed to be negative.

This means that $\hat{\pi}_{22} < \pi_{22}$, or equivalently, the preponderant target subclass C_2 is always underrepresented in the data set at hand.

In the following, some additional notations are introduced. Like in Section 5.2, the weighted and unweighted mean estimates of the heterogeneous class C are denoted as $\hat{\mu}_w$ and $\hat{\mu}_{unw}$, respectively. The subclass mean estimates and respectively the true subclass means are $\hat{\mu}_{2k}$ and μ_{2k}, $k = 1, 2$, respectively. The absolute difference between the true subclass means is denoted as $u = |\mu_{21} - \mu_{22}|$. The subclass variances of the candidate feature are σ_k^{μ}, $k = 1, 2$. The quotient between the feature variance in the preponderant target subclass, C_2, and the feature variance in the less prevalent target subclass, C_1, is $s_\mu = \frac{\sigma_2^\mu}{\sigma_1^\mu}$.

The *weighted* mean estimate of the candidate feature in the heterogeneous class is computed as a linear combination of the empirical subclass means with weights given by the true subclass prevalences:

$$\hat{\mu}_w = \pi_{21}\hat{\mu}_{21} + \pi_{22}\hat{\mu}_{22}.$$

The empirical subclass means are unbiased. Using the true subclass prevalences as weights, the weighted mean estimate is unbiased with respect to the true mean of the heterogeneous class, too.

The *unweighted* mean estimate is naturally weighted by means of the observed subclass prevalences:

$$\hat{\mu}_{unw} = \hat{\pi}_{21}\hat{\mu}_{21} + \hat{\pi}_{22}\hat{\mu}_{22}.$$

The weighted mean estimates $\hat{\mu}_w$ reduce the bias caused by not taking into account the true subclass prevalences. However, their efficiency depends on their variance, too. Hence, a comparison between the mean squared errors of the weighted and unweighted mean estimates is more suitable than just one of their expectations.

Since computations of the subclass mean estimates are based on disjoint pools of observations, it is plausible that the estimated subclass means are uncorrelated. Thus, $Cov(\hat{\mu}_{21}, \hat{\mu}_{22}) = 0$ and therefore the benefit condition (5.2.11) from Section 5.2 applies also in the context of weighted class means.
Adapting condition (5.2.11) to the context of weighted means, weighted mean estimates are

more efficient than the unweighted ones if:

$$\underbrace{\left[\frac{1+(1+\rho^*)f_t}{\rho^* f_t}\right]\left\{[2+(2+\rho^*)f_t] - \frac{[2(1+\rho^*)f_t+2+\rho^*]}{1+\rho^*}s_\mu\right\}}_{=\tilde{G}(s_\mu,\rho^*,f_t)} < \frac{N_2u^2}{\sigma_1^\mu}. \tag{5.3.1}$$

The theoretical discussion regarding the weighted error estimates in Section 5.2 applies here in a similar way. However, assuming σ_1^μ to be constant, the weighting benefit is assessed independently of it when s_μ is varied below and above 1. Therefore, the left and right hand side terms of this inequation are now independent. This is a consistent reduction in the complexity of the problem.
Another simplification is that s_μ, the quotient between the variances of the candidate feature in the subclasses, has now a unique interpretation (compare to Figure 5.1).

Now, if the user knows the target subclass distributions, then he can easily compute the value of the left and right hand side terms in (5.3.1) and compare them. Thus, the assessment of the weighting benefit is rather uncomplicated in this case.

If the user has no a-priori information about the target subclass distributions, he can compute means and variance estimates of the subclass means by resampling techniques based on the observations in the data set at hand.

Our mathematical investigations of the benefit condition (5.3.1) should enable a user with only some vague idea about the target distributions to make an adequate decision about the further analysis workflow.

It holds:

$$\lim_{f_t\to\infty}\tilde{G}(s_\mu,\rho^*,f_t) = \infty, \qquad (\rho^* = const.,\ s_\mu = const.) \tag{5.3.2}$$

$$\lim_{\rho^*\to -1}\tilde{G}(s_\mu,\rho^*,f_t) = \infty \qquad (f_t = const.,\ s_\mu = const.) \tag{5.3.3}$$

$$\lim_{\rho^*\to 0}\tilde{G}(s_\mu,\rho^*,f_t) = \begin{cases}\infty & \text{if } s_\mu > 1\\ -\infty & \text{if } s_\mu < 1.\end{cases} \qquad (f_t = const.) \tag{5.3.4}$$

The following remarks should help to decide on whether weighted mean estimates suit a particular diagnostic situation, or not. Remember that σ_1^μ is constant throughout this section.

Remark 5.3.1.

(i) From (5.3.2) it is clear that in the context of particularly high values of f_t, thus of highly unbalanced target subclass structures, the danger of inefficient weighted estimates increases.

(ii) The relation (5.3.3) indicates that a large degree of mismatch between the true and observed subclass prevalences, i.e. a high $|\rho^*|$, can result in inefficient weighted estimates, too.

(iii) In general, the smaller the true variance of the candidate feature in the preponderant target subclass than in the less prevalent target subclass, the higher the chances for a beneficial weighting.

(iv) In general, the larger u, the gap between the target subclass means, the more chances for a beneficial weighting.

(v) From *(iii)* and *(iv)* it results that weighted mean estimates are especially efficient under a pronounced target subclass structure.

(vi) The larger the size N_2 of the heterogeneous class, the lower the minimal quotient between u and $\sqrt{\sigma_1^\mu}$ which is required for a beneficial weighting; thus, the larger the maximally allowed degree of overlap between the subclasses (subclass structure becomes less important).

Using the benefit condition (5.3.1), μ_{22} should lie outside of an interval $(\mu_{21}-\eta\sqrt{\sigma_1^\mu}, \mu_{21}+\eta\sqrt{\sigma_1^\mu})$ with $\eta = \sqrt{\frac{\tilde{G}(s_\mu,\rho^*,f_t)}{N_2}}$, given $\tilde{G} > 0$; in the context of approximately normally distributed subclasses, a large u has also the interpretation that the mean of the under-represented subclass is rather unspecific for the distribution of the over-represented subclass.

5.3.1 Subcase 1

We address again the special case when the target subclass structure is unbalanced ($f_t > 1$), while the observed subclass structure is balanced ($f_d = 1$).

We remind also that the condition (5.3.1) is derived using the unbiasedness property of the plug-in means. This means that N_2 is automatically considered to have a reasonable size in order to provide reliable mean estimates.

Assume that the heterogeneous class has at least $N_2 = 50$ samples, the variance in the preponderant target subclass is at most 5 times larger than the variance in the less prevalent target subclass (i.e. $s_\mu \leq 5$), and $\mu_{22} \notin (\mu_{21} - \sqrt{\sigma_1^\mu}, \mu_{21} + \sqrt{\sigma_1^\mu})$. Table 5.10 indicates the minimal values of the quotient of the true subclass prevalences f_t which are necessary for a beneficial weighting under these settings (see the upper plot in Figure 5.6). Obviously, the same table applies when $N_2 = 100$ and $\mu_{22} \notin (\mu_{21} - \sqrt{0.5\sigma_1^\mu}, \mu_{21} + \sqrt{0.5\sigma_1^\mu})$ (see Remark 5.3.1(vi)).

Table 5.10: Subcase 1: Lower limits of the f_t-intervals associated to a weighting benefit when $s_\mu \geq 1$ and $N_2 \frac{u^2}{\sigma_1^\mu} \geq 50$. *For a beneficial weighting a minimal degree of unbalance in the target subclass structure* $f_t^l(s_\mu)$ *is required.*

s_μ**-values**	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0
$f_t^l(s_\mu)$	1.1	1.1	1.2	1.33	1.48	1.65	1.86	2.13	2.46

Remark 5.3.2 (Case $s_\mu > 1$, or equivalently, the variance of the candidate feature is higher in the preponderant than in the less prevalent target subclass (see upper plot in Figure 5.6 and the corresponding Table 5.10)).

(i) Like noticed from the upper plot in Figure 5.6 and specified in Table 5.10, a minimal degree of unbalance f_t^l of the target subclass structure is required for a beneficial weighting; this is equivalent to the request of a minimal degree of suboptimality of the data at hand, since $f_d = 1$.

(ii) The larger s_μ, thus the larger the variance of the preponderant target subclass with respect to the variance of the less prevalent target subclass, the larger the minimally required degree of unbalance of the target subclass structure, too (i.e. also the larger the minimally required degree of mismatch between the study and the target population).

Remark 5.3.3 (Case $s_\mu = 1$, or equivalently, the candidate feature has equal variances in the target subclasses).

In this case $\tilde{G} = 4$ as stated by Proposition 5.2.8(i). Thus, the benefit is sure for every degree of unbalance f_t of the target subclass structure if the degree of overlap between the two target subclasses $\frac{u}{\sqrt{\sigma_1^\mu}}$ is higher than $\frac{2}{\sqrt{N_2}}$.

The previous remark claims that given a heterogeneous class with at least 50 observations, the relative gap between the subclass means $\frac{u}{\sqrt{\sigma_1^\mu}}$ should just exceed 0.28.

Assume a normal distribution in the small target subclass. Then the former condition requires that the mean of the preponderant target subclass does not belong to the 22% of the most spe-

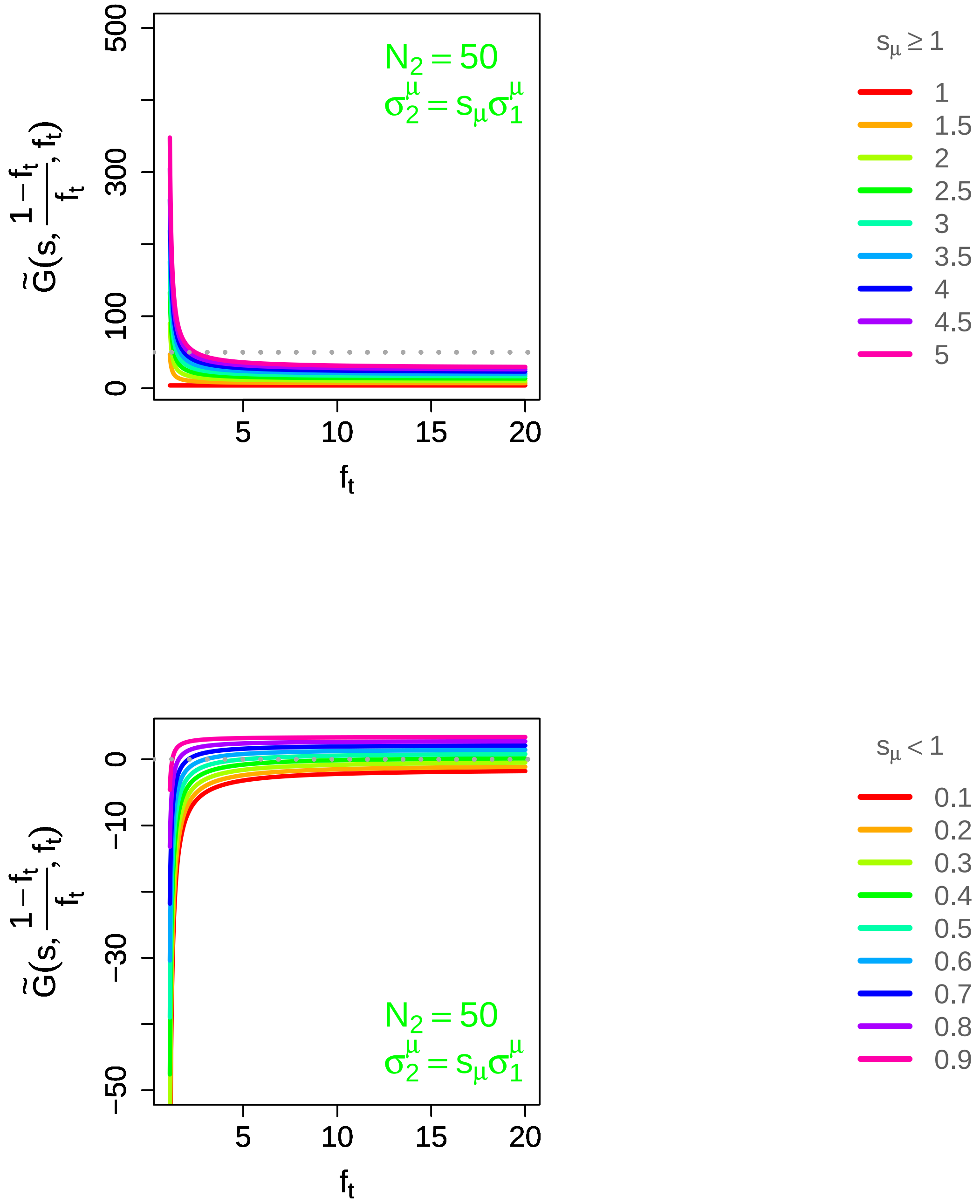


Figure 5.6: Subcase 1: Weighted means - behavior of $\tilde{G}(s, \frac{1-f_t}{f_t}, f_t)$ when $f_t \leq 20$. Upper plot: *The gray dotted line corresponds to $N_2 = 50$ and $\frac{u}{\sqrt{\sigma_1^\mu}} = 1$. When $s_\mu \in [1, 5]$ the weighting benefit is sure for any $f_t \geq 2.5$.* Lower Plot: *The gray dotted line corresponds to* 0. *Independently of the target subclass distribution structure, when $s_\mu < 1$, the weighting benefit is sure up to $s = 0.5$ for any $f_t \leq 5$.*

cific observations for this distribution; equivalently, $\mu_{22} \notin (\mu_{21} - 0.28\sqrt{\sigma_1^\mu}, \mu_{21} + 0.28\sqrt{\sigma_1^\mu})$.

Table 5.11: Subcase 1: Upper limits of the f_t-intervals associated to a weighting benefit when $s_\mu < 1$ and $N_2\frac{u^2}{\sigma_1^\mu} \geq 0$. *Weighting is beneficial up to a certain degree of unbalance in the target subclass structure* $f_t^h(s_\mu)$.

s_μ**-values**	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
$f_t^h(s_\mu)$	>20	>20	>20	12.99	4.99	2.99	2.09	1.57	1.23

Remark 5.3.4 (**Case** $s_\mu < 1$**, or equivalently, the variance of the candidate feature is lower in the preponderant subclass than in the less prevalent target subclass (see the lower plot in Figure 5.6 and the corresponding Table 5.11)).**

(i) Like noticed from the lower plot in Figure 5.6 and specified in Table 5.11, weighted estimates are inefficient compared to unweighted ones starting from a certain degree of unbalance f_t^h of the target subclass structure.

(ii) The smaller the variance of the large target subclass compared to the variance of the small target subclass (i.e. the smaller s_μ), the higher the minimal degree of target unbalance f_t^h at which weighting becomes dangerous.

Assume that $s_\mu \leq 0.5$ which means that the variance in the preponderant target subclass represents at most 50% of the variance in the less prevalent target subclass. Then weighting is beneficial at least up to a five times higher prevalence of the preponderant target subclass (i.e. for every $f_t \leq 5$, or equivalently, $\pi_{22} \leq 5 \cdot \pi_{21}$). See Table 5.11 and the lower plot in Figure 5.6.

5.3.2 Subcase 2

According to this special case, both the observed and the target subclass prevalence structures are unbalanced and $\hat{\pi}_{22} \leq \pi_{21}$. Tables 5.12 and 5.13 present the ranges of the degree of unbalance f_t for which weighting is beneficial under the settings from Figures 5.7 and 5.8, respectively.

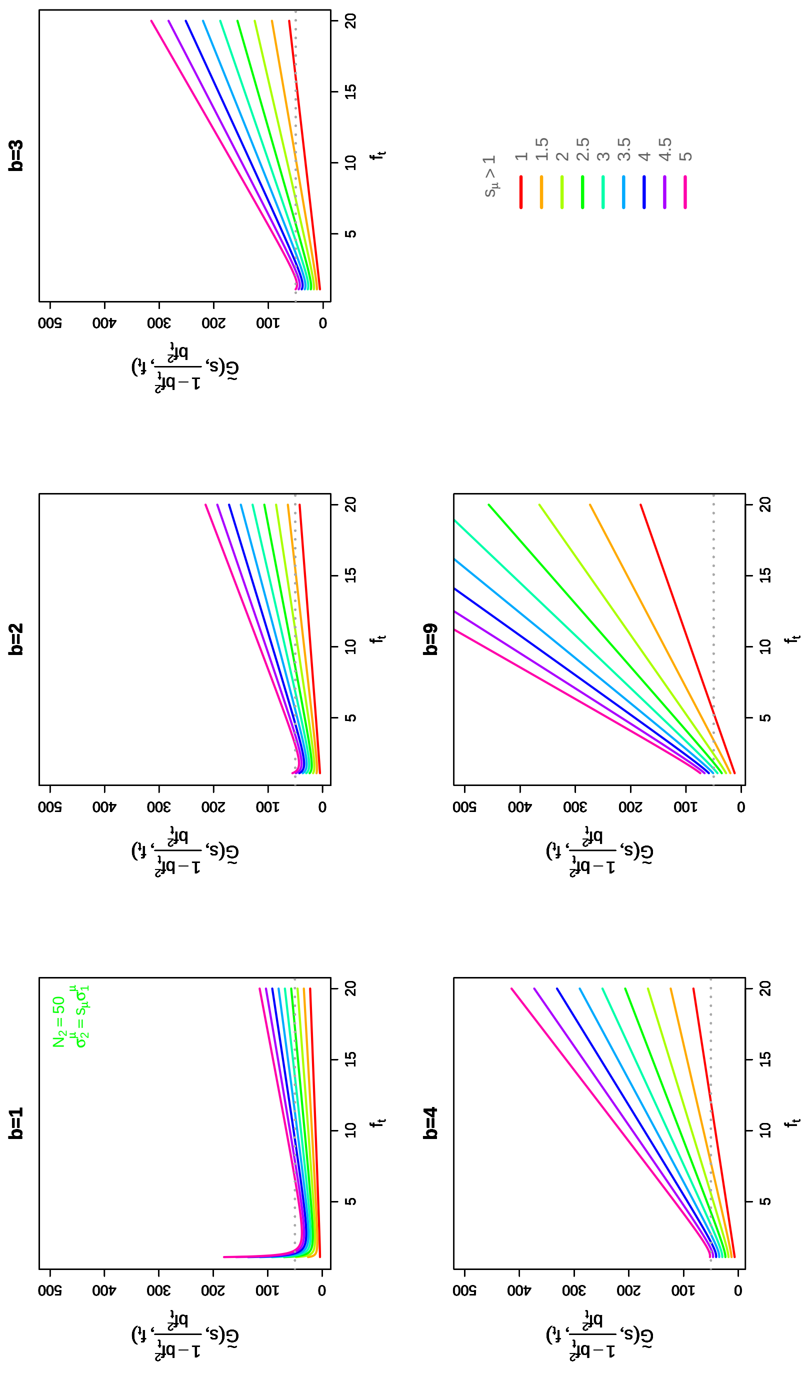


Figure 5.7: Subcase 2: Weighted means - behavior of $\tilde{G}(s, \frac{1-bf_t^2}{bf_t^2}, f_t)$ when $s_\mu \geq 1$ and $f_t \leq 20$. *The gray dotted line corresponds to $N_2 = 50$ and $\frac{u}{\sqrt{\sigma_1^\mu}} = 1$. When $b = 1$ and $s_\mu \leq 2$, thus the variance in the preponderant target subclass is at most two times larger than the variance in the less prevalent target subclass, weighting is beneficial for any degree of unbalance $f_t \leq 20$. When $b = 9$, with the same setting, weighting is sure up to $f_t = 2.44$.*

Table 5.12: Subcase 2: Limits of the f_t-intervals associated to a weighting benefit when $s_\mu \geq 1$, $N_2\frac{u^2}{\sigma_1^\mu} \geq 50$, and $b \in \{1, 9\}$. *Weighting is beneficial for a degree of unbalance in the target subclass structure within* (f_t^l, f_t^h).

	s_μ-values	1	1.5	2	2.5	3	3.5	4	4.5	5
$b = 1$	$f_t^l(s_\mu)$	1.1	1.1	1.1	1.16	1.22	1.29	1.37	1.47	1.59
	$f_t^h(s_\mu)$	>20	>20	>20	17.22	13.76	11.26	9.36	7.85	6.59
$b = 9$	$f_t^l(s_\mu)$	1.1	1.1	1.1	1.1	1.1	–	–	–	–
	$f_t^h(s_\mu)$	5.33	3.41	2.44	1.83	1.41	–	–	–	–

Remark 5.3.5 (**Case $s_\mu \geq 1$, or equivalently, the variance of the candidate feature is higher in the preponderant than in the less prevalent target subclass (see Figure 5.7 and the corresponding Table 5.12)**).

(i) Like noticed from Figure 5.7 and indicated by Table 5.12, weighting is especially feasible in the case of a small to moderate degree of unbalance in the target population given just diametrally opposite subclass prevalences in the data at hand with respect to the target. This is the case when $b = 1$.

(ii) Weighting is beneficial up to a degree f_t^h of unbalance in the target population and a degree b of diametral opposition between the quotients of true and observed subclass prevalences.

(iii) Weighting is especially dangerous for large values of s_μ, thus for a much higher true variance of the candidate feature in the preponderant than in the less prevalent target subclass. Another disadvantageous condition is a large b, thus a high degree of more than inverse proportionality between the target and observed subclass structures.

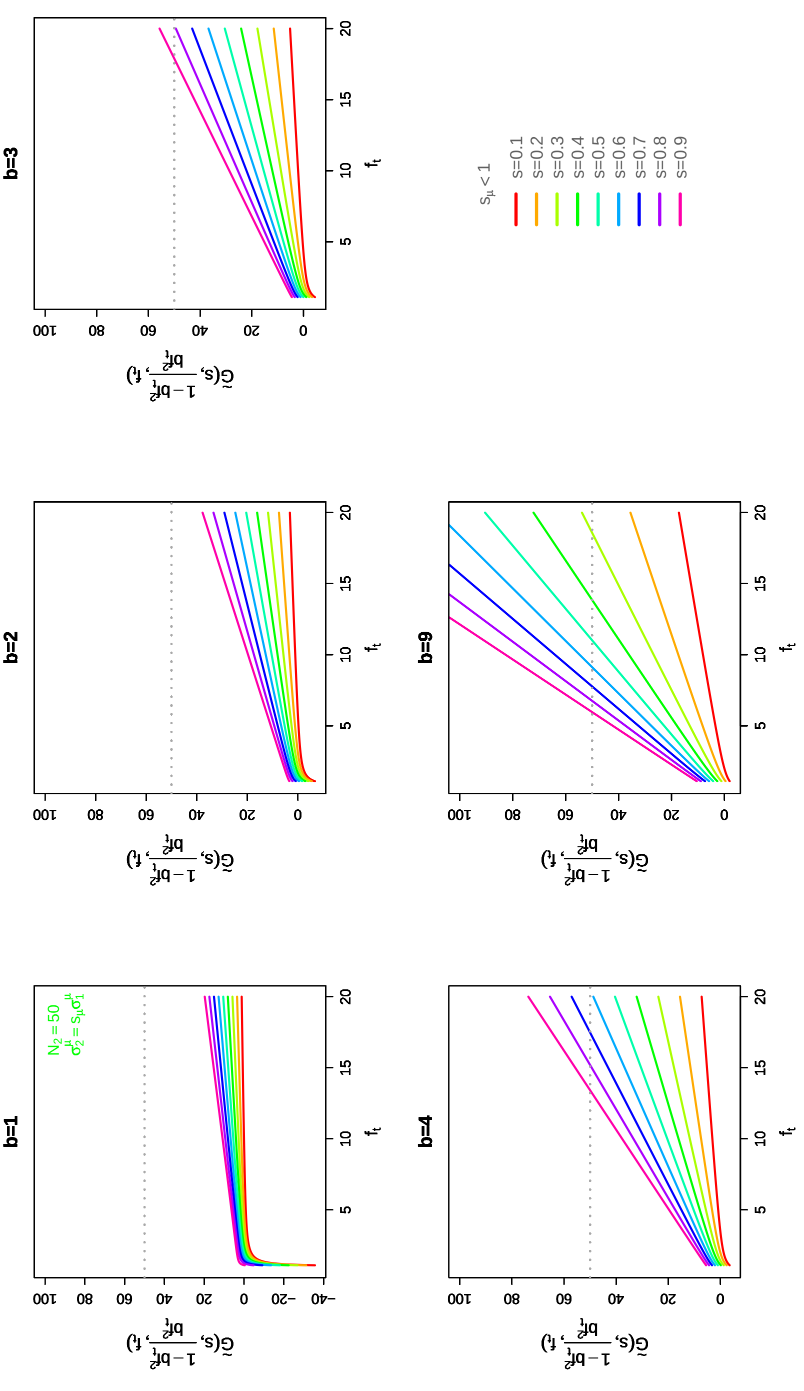


Figure 5.8: Subcase 2: Weighted means - behavior of $\tilde{G}(s, \frac{1-bf_t^2}{bf_t^2}, f_t)$ when $s_\mu < 1$ and $f_t \le 20$. *The gray dotted line corresponds to* $N_2 = 50$ *and* $\frac{u}{\sqrt{\sigma_1^\mu}} = 1$. *When* $b = 1$ *weighting is beneficial for any degree of unbalance* $f_t \le 20$. *When* $b = 9$ *and* $s \le 0.9$ *weighting is beneficial for any* $f_t \le 5.96$.

Table 5.13: Subcase 2: Upper limits of the f_t-intervals associated to a weighting benefit when $s_\mu < 1$, $N_2\frac{\mu^2}{\sigma_1^\mu} \geq 50$, and $b \in \{1, 9\}$. *Weighting is beneficial up to a certain degree of unbalance in the target subclass structure f_t^h.*

	s_μ-values	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
$b = 1$	$f_t^h(s_\mu)$	>20	>20	>20	>20	>20	>20	>20	>20	>20
$b = 9$	$f_t^h(s_\mu)$	>20	>20	18.58	13.85	11.02	9.12	7.77	6.75	5.96

Remark 5.3.6 (Case $s_\mu < 1$, or equivalently, the variance of the candidate feature is smaller or equal in the preponderant subclass than in the less prevalent target subclass (see Figure 5.8 and the corresponding Table 5.13)).

The smaller the variance σ_2^μ of the preponderant target subclass with respect to the variance σ_1^μ of the less prevalent target subclass, the higher the chances for a weighting benefit.

5.4 Conclusions

The theoretical study of how non-representativeness in the data impacts on weighted estimates of classification errors is the focus throughout this chapter. Our second interest is to check how does the data non-representativeness impact on the weighted parameter estimates. Here only the weighted mean estimates are considered.

The target population has two classes. One of them, labeled as D or 1, is homogeneous, while the other one, C or 2, is heterogeneous with two subclasses, C_1 and C_2. The true subclass prevalences in the target population, π_{21} and π_{22}, corresponding to C_1 and respectively C_2, are known.

The data at hand, also called study population, represents a small excerpt of the target population collected during some study and used to build the classification rule. It has the same class and subclass composition as the target population. But, like it is usually the case in the diagnostic practice, subclasses appear in the study population according to different observed prevalences than the true ones. These are $\hat{\pi}_{21}$ and $\hat{\pi}_{22}$.

The degree of mismatch between the target population and the data at hand is quantified by a measure of comparison between the quotients of the true and observed subclass prevalences. Given an unbalanced target subclass structure ($\pi_{21} \neq \pi_{22}$), subclass C_2 is by convention the preponderant subclass, i.e. $\pi_{22} > \pi_{21}$. Also by convention, C_2 is the under-represented subclass in the data at hand, i.e. $\hat{\pi}_{22} < \pi_{22}$. The situation of a large target subclass being under-sampled in the data set at hand is considered as it is more likely to be relevant for the diagnostic practice.

In Chapter 4 the dangers related to building classification rules based on strongly biased observed subclass prevalences are highlighted. Two modalities to account for the true subclass prevalence structure are presented:

(a) in the phase of rule validation/optimization by computation of weighted error estimates as sums of the subclass analogs with the true subclass prevalences as weights;

(b) in the phase of rule building by computation of weighted class distribution parameters: weighted class means and covariance matrices according to the true subclass prevalences.

Simulation studies provide evidence for the superior performance of rules which account in both ways for the true subclass structure. They approach better the expected classification errors in the target population especially when the degree of mismatch between the data at hand and the target population is moderate to high. Therefore, in the context of a moderate to high non-representativeness of the data set at hand, weighting seems to be the right solution.

However, the question of interest is if the benefits associated to a bias reduction are not endangered by some simultaneous variance inflation. Variance of the weighted estimates might be enhanced by the contribution of the up-weighted subclasses, thus, of those subclasses being under-represented in the data set at hand. The theoretical research in this chapter aims at a description of suitable as well as potentially dangerous situations for the application of weighted estimates. In this way a prime theoretical basis is obtained that might be of help in the decision-making about weighting or not.

The most important quantities used to assess the weighting benefit are the degree of mismatch between study and target population and the degree of unbalance in the target population. Given the collected data and known true subclass prevalences, both quantities are known. The latter is computed as quotient between the true prevalences of the preponderant (C_2) and the less prevalent target subclass (C_1) ($f_t = \frac{\pi_{22}}{\pi_{21}}$). The former is computed as the relative difference between the quotients of observed ($f_d = \frac{\hat{\pi}_{22}}{\hat{\pi}_{21}}$) and true subclass prevalences (f_t) with respect to the target population ($\frac{f_d - f_t}{f_t}$).

Weighted classification errors. The weighted error estimate in the heterogeneous class is computed as the weighted sum of the subclass misclassification rates with weights given by the true subclass prevalences. Suppose the user knows the true subclass errors in the target population. Then the inequality (5.2.12) can be easily evaluated and provides a concrete decision on whether to weight or not.

If a user has no idea about the true subclass errors, then the necessary information can be obtained by adequate simulations.
Our theoretical research provides some orientating information for a user that has a rather vague idea about the dimension of the subclass errors in the target. Tables 5.4 and 5.5 offer potentially useful criteria to decide upon using weighting or preferably avoiding it.

The subclass error probabilities are denoted by p_{21} and p_{22}. They are also referred as error probabilities of the less prevalent and preponderant target subclass, respectively.

In general, weighted estimates have good chances to outperform the unweighted ones given a reasonable size of the heterogeneous class, a high discrepancy between the subclass error probabilities and a value of the error probability p_{21} in the less prevalent subclass which is close to 0 or 1.

Further it is assumed without loss of generality that p_{21} has some constant value below 0.5, while p_{22} is variable.
When the distance between the true subclass error probabilities grows, also the range of the degree of mismatch between data at hand and target population becomes larger, for which the weighted estimates are superior to the unweighted ones. Extremely high or extremely low degrees of mismatch between the target population and the data at hand present under certain situations an increased danger potential.

Given equal subclass error probabilities, i.e. $p_{21} = p_{22}$, weighting provides no benefit. The higher the gap between the subclass error probabilities, the higher the chances to achieve better error estimates of the heterogeneous class by weighting.

The most suitable setting for a beneficial weighting is when $p_{22} \geq 1 - p_{21}$. In this case, given a small value of p_{21} (e.g. $p_{21} \leq 0.15$), the benefit is sure for almost all degrees of mismatch between data at hand and the target population.

The most unfavorable setting for a beneficial weighting is when the subclass probabilities are located on the same side of 0.5, thus when either $p_{22} < p_{21}$ or $p_{21} < p_{22} < 0.5$. Obviously, the best chances for a weighting benefit in the former situation are given when p_{22} approaches 0, while in the latter situation when p_{22} approaches 0.5, thus when the dissimilarity between the subclass error probabilities is maximized.

The improvement potential of weighted estimates is regarded in more detail on two special subcases. First is addressed as it has more application in the diagnostic practice; second is used to investigate what happens in extreme cases:

- **Subcase** 1: *unbalanced true subclasses ($\pi_{21} < \pi_{22}$), balanced observed subclasses ($\hat{\pi}_{21} = \hat{\pi}_{22} = 0.5$)*
- **Subcase** 2: *unbalanced true subclasses ($\pi_{21} < \pi_{22}$), at least diametrally opposite unbalanced observed subclasses ($\hat{\pi}_{22} \leq \pi_{21}$ and $\hat{\pi}_{21} \geq \pi_{22}$).* Equivalently, this subcase is characterized by the relationship $f_d = \frac{1}{bf_t}$, with $b \geq 1$ being the degree of diametral opposition between the true and observed subclass prevalence structures.

Further, some theoretical findings regarding the weighting benefits in the special situations introduced by these two subcases are presented.

Subcase 1 ($\hat{\pi}_{21} = \hat{\pi}_{22} = 0.5$).

- Assume that $p_{21} \leq 0.4$ and $p_{22} \in (1 - p_{21}, 1]$. Then according to Corollary 5.2.7(i) weighted estimates are beneficial for any degree of unbalance of the target subclass structure starting already from 24 observations in the heterogeneous class.
- Starting from a size $N_2 = 100$ of the heterogeneous class, the weighting benefit is sure given any subclass error probabilities which differ by at least 10 percentage points, i.e. $|p_{21} - p_{22}| > 0.1$ as it is stated in Corollary 5.2.7(ii).
- Weighted error estimates are superior to the unweighted ones if the variance $p_{21}(1 - p_{21})$ of the misclassification event in the less prevalent target subclass is at least three times larger than the variance $p_{22}(1 - p_{22})$ of the misclassification event in the preponderant target subclass. See Proposition 5.2.6(i).

Subcase 2 ($\hat{\pi}_{22} \leq \pi_{21}$).

- The smaller the observed prevalence $\hat{\pi}_{22}$ of the large target subclass in comparison with the true prevalence π_{21} of the small target subclass, the less beneficial the weighted estimates in comparison to the unweighted ones. This is shown both by Tables 5.8 and 5.9 and Proposition 5.2.10(ii).
- If $p_{21} \leq 0.15$ and $p_{22} \geq 1 - p_{21}$, given at least 100 observations in the heterogeneous class, the benefit is sure at least up to a degree of unbalance of the target subclass structure of $f_t = 20$ and a degree of diametral opposition between the true and observed subclass structures of $b = 9$. See Table 5.9.

Weighted estimates of class means. The weighted mean estimate of a candidate feature in the heterogeneous class is computed as weighted sum of the empirical subclass means with weights given by the true subclass prevalences.

Suppose a user knows the true subclass distributions. Then the inequality (5.3.1) can be easily evaluated and provides a concrete decision on whether to weight or not.
If the user has no a-priori knowledge about the true subclass distributions, he can approximate them by means of resampling techniques starting from the data set at hand.
Our investigations about the efficiency of weighted mean estimates are designated for a user with a rather vague idea about the true subclass distributions.

The true means of the preponderant and less prevalent target subclasses are denoted as μ_{22} and μ_{21}, respectively.
The advantage of the weighted mean estimates upon the unweighted ones depends strongly on the gap between the true subclass means $|\mu_{21} - \mu_{22}|$ and the relationship between the target subclass variances of the candidate feature. Generally, the higher the difference between the true subclass means and the lower the variance of the preponderant target subclass in comparison to the less prevalent target subclass, the higher the chances for a benefit by weighting.
A favorable setting for weighting is when the true mean μ_{22} of the preponderant target subclass represents no typical observation for the distribution of the feature candidate in the less prevalent target subclass. This is the case when μ_{22} belongs rather to the tails of this distribution or completely outside of its range.

In general, the weighting benefit is endangered by extreme values of the degree of mismatch between data at hand and target population. Thus, given an extremely suboptimal or a well proportioned data at hand with respect to the target the decision on weighting or not should be processed with care. In our special subcases the degree of unbalance of the target subclass structure alone is already an indicator of the degree of suboptimality of the data at hand. Therefore, in these cases, the weighting decision should be made with particular care if the target subclass structure is highly unbalanced or nearly balanced.

If the subclass structure in the target population is pronounced, then the chances for a weighting benefit increase rapidly. This happens if the relative distance between the true subclass means with respect to the variance in the small target subclass is large and especially if the variance in the large target subclass is small.

Chapter 6

Discussion

The early identification of complex diseases by means of biomarker combinations is a challenging field in the medical research of the last decades. Statistical methods provide support for achieving meaningful and highly accurate diagnostic rules based on biomarker combinations. However, diagnostic rules obtained by means of multivariate classification algorithms should not only fulfill the goal of a reliable assessment of the disease status but also find a large acceptance in the medical practice.

We address the suitability of classification rules for the diagnostic practice regarding the accomplishment of important quality standards like *simplicity*, *interpretability*, *high accuracy*. Simple and interpretable rules enable an easy assessment by medical professionals and a good comprehension of the connections between biomarker concentrations and the disease status. Accurate rules guarantee a low probability of erroneous assignments, thus a reliable diagnosis on the target population. However, due to the trade-off between the performance and the complexity of the classification models, simple and interpretable rules may be obtained in change of some performance loss.

Dangers and benefits associated to the accomplishment of these quality standards are highlighted using Logic Regression as example. This new tree-based classification method has a great theoretical potential to provide simple, interpretable and therefore highly accepted rules in the diagnostic practice.
However, a shortcoming of this method is that continuous predictors should be dichotomized first, since this method works only with binary predictors. Two dichotomization alternatives are used, one based on the empirical quantiles and one based on a best threshold which is optimized by Logistic Regression with respect to the classification task. The performance of Logic Regression is evaluated in comparison to that of established classification algorithms, like Regularized Discriminant Analysis and Random Forests, which are known to provide good rules in terms of diagnostic accuracy.

Simulations of various distributional conditions show that the potential of this method is strongly dependent on the underlying data structure. Logic Regression provides the simplest and therefore also the most attractive rules for the diagnostic practice, while it loses up to 5 percentage points diagnostic accuracy in comparison to the other methods. The dichotomization procedure based on an optimized threshold is recommended for the diagnostic practice.

According to the simulations, the quantiles-based Logic Regression has a larger bias and instability in comparison to the other algorithms in many contexts. Despite of these findings, the quantiles-based method provides the best results on the real data example from a study for the early identification of rheumatoid arthritis.

The advantage of simple and interpretable models can be enhanced in future by a faster R-implementation of this method as well as by the right choice of the dichotomization method, triggering the high acceptance of this method within the diagnostic frame.

However, the main focus of this thesis is to find appropriate ways to adapt a classification rule for the diagnosis in a heterogeneous target population, when the target subclass prevalences are known. This problem is studied both empirically and theoretically in Chapters 4 and 5, respectively.

As a diagnostic rule has to accurately predict the disease status on the target population, an issue of great importance is to be able to design reliable diagnostic rules also when suboptimal data is used for learning.

In the statistical literature the only available method to account for the known target subclass prevalence structure is the so-called post-weighting. The final misclassification error estimate is weighted by means of the true subclass prevalences first in the phase of rule validation, thus, after building the rule on the suboptimal learning data. Our idea is to account for the true subclass prevalences already in the phases of rule building and optimization.

We propose four weighting algorithms for learning rules in the context of a heterogeneous target population when the observed and the target subclass prevalences are different.

Three of them account for the target subclass prevalence structure not only in the validation phase, when the final estimate of the rule performance is corrected with respect to the target situation, but already in the phases of rule building and/or optimization.

They are based on weighted estimates of the misclassification error and of the distribution parameters of the heterogeneous class. These weighted estimates are computed by replacing the observed with the target subclass prevalences in the computation formula of the common plug-in estimates. Their statistical properties are investigated both empirically and theoretically. Throughout this survey a homogeneous disease class and a heterogeneous control class with two subclasses are considered.

A simulation study is carried out to compare the performance of the weighting algorithms over several distribution configurations in the target population. This empirical survey is performed with respect to four features. Two of them are assumed to be informative and the other two to be non-informative for the class and subclass discrimination. Also, the investigations are restricted to a situation of interest for the diagnostic practice, which arises especially in the context of case-control studies: the subclass structure is balanced in the data used for learning.
The algorithms which account for the target subclass prevalence structure both in the phase of rule building, optimization and validation are proved to be superior in terms of accuracy and stability of their results in comparison to algorithms which weight only in the optimization and/or validation phase. According to our simulations, weighted rules approach well the expected classification errors in the target population especially when the discrepancy between the observed and the target subclass prevalences is high. Also, simulations show that a large discrepancy between the subclass degrees of overlap with the homogenous class as well as a pronounced subclass structure in the target population may be favorable conditions for a beneficial weighting.
The algorithms are verified with the rheumatoid arthritis real data. Based on the simulation and real data results, the algorithms which weight in all phases of the rule development process are strongly recommended given any suboptimal learning data. Their application guarantees, if not always a high benefit, then at least safe results in case of data non-representativeness.

According to the empirical survey, weighted rules approach better than unweighted ones the true classification errors in the context of data non-representativeness. From this point of view, the use of weighted estimates for the misclassification error and for the distribution parameters of the heterogeneous class seems to be beneficial. However, the benefits associated to a bias reduction by weighting can be endangered by a simultaneous variance inflation due to up-weighting of the under-represented subclasses in the data used for learning. Suitable as well as potentially dangerous situations for the application of weighted estimates are ascertained by means of theoretical investigations.

In general, weighted error estimates have good chances to outperform the unweighted ones given a reasonable class size, a high discrepancy between the subclass error probabilities and a medium to high discrepancy between the true and observed subclass prevalences. Also, given a small error probability in the less prevalent target subclass, the most suitable situation for weighting is when the error probability in the preponderant target subclass is at least equal to the hit probability in the less prevalent target subclass.
Weighted mean estimates have good chances to outperform the unweighted ones given a reasonable class size, a relevant distance between the true subclass means and a lower variance of the candidate feature in the preponderant than in the less prevalent target subclass. The latter two aspects stand actually for a well contoured subclass structure in the heterogeneous class.

One should notice that the subclass error probabilities are theoretical analogs for the degrees of subclass overlaps with the homogenous class addressed in the empirical survey. Also, a medium to high discrepancy between the true and observed subclass prevalences corresponds to a medium to high degree of unbalance of the target subclass structure in the empirical survey, which assumes a balanced observed subclass structure. Using these observations it becomes clear that the theoretical results confirm essentially the conclusions drawn from the empirical study.

However, the theoretical survey indicates also that the weighting benefit is endangered by extreme values of mismatch between the data used for learning and the target population. Given extremely different or very similar observed and target subclass prevalences, the decision on whether to weight or not should be processed with particular care.

We compare theoretically only weighted and unweighted estimates of the misclassification error and of the class means in the context of one candidate feature. However, the weighting strategies involve also weighted estimates of the covariance matrices. Therefore, a future task is to evaluate their statistical efficiency in comparison to the unweighted covariance estimates, too. Also, future investigations may be carried out in the context of many features.

Besides, this work addresses empirically and theoretically only the simplified case when only one class is heterogenous and it has two subclasses. The envision of a proper way to generalize both the empirical as well as the theoretical results for the case when both classes are heterogeneous and/or multiple subclasses are available is left for future work.

The weighting algorithms are tailored for the use with the Regularized Discriminant Analysis. It should be however pointed out that the weighting ideas are general and they can be adapted in future to other classification methods, too.

Bibliography

Arnett, F.C., Edworthy, S.M., Block, D.A., McShane, D.J., Fries, J.F., & Cooper, N.S. 1988. The American Rheumatism Association 1987 Revised Criteria for the Classification of Rheumatoid Arthritis. *Arthritis & Rheumatism*, **31**, 315–324.

Baron, Anna E. 1991. Misclassification among Methods Used for Multiple Group Discrimination-the Effects of Distributional Properties. *Statistics in Medicine*, **10**, 757–766.

Bauer, E., & Kohavi, R. 1999. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. *Machine Learning*, **36**, 105–139.

Begg, C. B., & Greenes, R.A. 1983. Assessment of Diagnostic Tests when Disease is Subject to Selection Bias. *Biometrics*, **39**, 206–215.

Breiman, L. 1996. Heuristics of Instability and Stabilization in Model Selection. *Annals of Statistics*, **24**(6), 2350–2383.

Breiman, L. 1997. *Random Forests-Random Features*. Tech. rept. Statistics Department, University of California, Berkely, CA 94720.

Breiman, L. 2001. Random Forests. *Machine Learning*, **45**, 5–32.

Breiman, L., Friedman, J. H., Olshen, R.A., & Stone, C.J. 1984. *Classification and Regression Trees*. Chapman and Hall: Boca-Raton, Florida.

DeLong, Elizabeth R., DeLong, David M., & Clarke-Pearson, Daniel L. 1988. Comparing the Areas under Two or More Correlated Receiver Operating Characteristic Curves: a Nonparametric Approach. *Biostatistics*, **44**, 837–845.

Dietterich, T. G. 1998. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neuronal Computation*, **10**, 1895–1923.

Dodd, Lori E., & Pepe, Margaret S. 2003. Partial AUC estimation and regression. *UW Biostatistics Working Paper Series*, **Working Paper 181**.

Etzioni, R., Kooperberg, C., Pepe, M.S., & Smith, R. 2003. Combining biomarkers to detect disease with application to prostata cancer. *Biostatistics*, **4**(4), 523–538.

Eugster, M. J. A., Hothorn, T., & Leisch, F. 2008. *Exploratory and Inferential Analysis of Benchmark Experiments*. Tech. rept. Department of Statistics, University of Munich.

Fisher, R.A. 1936. The use of multiple measurments in taxonomic problems. *Annals of Eugenics*, **7**, 179–188.

Friedman, J. H. 1989. Regularized Discriminant Analysis. *Journal of the American Statistical Association*, **84**(405), 165–175.

Hand, D.J. 1992. Statistical methods in diagnosis. *Statistical Methods in Medical Research*, **1**(1), 49–67.

Hand, D.J., & Vinciotti, V. 2003. Local Versus Global Models for Classification Problems: Fitting Models Where it Matters. *The American Statistician*, **57**(2), 124–131.

Hanley, J.A., & Hajian-Tilaki, K.O. 1997. Sampling Variability of Nonparametric Estimates of the Areas under Receiver Operating Characteristic Curves: an Update. *Academic Radiology*, **4**, 49–58.

Hastie, T., & Tibshirani, R. 1996. Discriminant Analysis by Gaussian Mixtures. *Journal of the Royal Statistical Society, Series B*, **58**(2), 155–176.

Hastie, Trevor, Tibshirani, Robert, & Friedman, Jerome. 2001. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer Series in Statitics. Springer-Verlag: New York.

Hosmer, David W., & Lemeshow, S. 2000. *Applied Logistic Regression*. 2nd. edn. Wiley Series in Probability and Statitics. John Wiley and Sons, Inc.: New York.

Hothorn, T., Leisch, F., Zeileis, A., & Hornik, K. 2005. The Design and Analysis of Benchmark Experiments. *Journal of Computational and Graphical Statistics*, **14**(3), 675–699.

Irwig, L., Glaziou, Paul P., Geoffrey, B., Chock, C., Mock, P., & Simpson, Judy M. 1994. Efficient Study Designs to Assess the Accuracy of Screening Tests. *American Journal of Epidemiology*, **140**(8), 759–769.

Janes, H., & Pepe, M. 2006. The Optimal Ratio of Cases to Controls for Estimating the Classification Accuracy of a Biomarker. *Biostatistics*, **7**(3), 456–468.

Kohavi, R. 1995. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *International Joint Conference on Artificial Intelligence*.

Kooperberg, C., & Ruczinski, I. 2006. *LogicReg: Logic Regression*. R package version 1.4.3.

Liaw, A., & Wiener, M. 2002. Classification and Regression by randomForests. *R News*, **2**(3), 18–22.

McLachlan, G.J. 1992. *Discriminant Analysis and Statistical Pattern Recognition*. John Wiley and Sons Ltd: New York.

Obuchowski, N., & Zhou, X.-H. 2002. Prospective Studies of Diagnostic Test Accuracy when Disease Prevalence is Low. *Biostatistics*, **3**(4), 477–492.

Pepe, M. 2003. *The Statistical Evaluation of Medical Tests for Classification and Prediction*. Oxford University Press Inc.: New York.

Plutowsky, M.E.P. 1995. *Survey: Cross-Validation in Theory and in Practice*. Tech. rept. Department of Computational Science Research David Sarnoff Research Center, Princeton, New Jersey, USA.

Powels, B. 2001. *Diplomarbeit: Diskriminanzanalyse bei fast-singulären Kovarianzmatrizen*. TU-Dortmund Statistics Department, Dortmund.

Ruczinski, I. 2000. *Logic Regression*. Ph.D. thesis, University of Washington, Washington.

Ruczinski, I., Kooperberg, C., & LeBlanc, M. L. 2003. Logic Regression. *Journal of Computational and Graphical Statistics*, **12**(3), 475–511.

SAS-Institute. 2000. *SAS V8.2*. Cary, NC, USA.

SAS-Institute. 2005. *JMP Design of Experiments, Release 6*. Cary, NC, USA. ISBN 1-59047-816-9.

Schmitt, R.I. 2005. *Master Thesis: Logic Regression in Diagnostic Classification Problems*. TU-Dortmund, Statistics Department, Dortmund, in Collaboration with Roche Diagnostics GmbH, Penzberg.

Schwender, H., & Ickstadt, K. 2007. Identification of SNP interactions using logic regression. *Biostatistics*. Epub ahead of print.

Sukhatme, S., & Beam, C.A. 1994. Stratification in Nonparametric ROC Studies. *Biometrics*, **50**, 149–163.

Szepannek, G., & Luebke, K. 2004. *Different Subspace Classification*. Tech. rept. Department of Statistics, University of Dortmund.

Szepannek, G., & Weihs, C. 2005. *Variable Selection for Discrimination of More than Two Classes Where Data are Sparse*. Tech. rept. Department of Statistics, University of Dortmund.

Szepannek, G., & Weihs, C. 2006. *Local Modelling in Classification on Different Feature Subspaces*. Tech. rept. Department of Statistics, University of Dortmund.

Team, R Development Core. 2007. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Titterington, D. M., & Bowman, A. W. 1985. A Comparative Study of Smoothing Procedures for Ordered Categorical Data. *Journal of Statistical Computation and Simulation*, **21**, 291–312.

Tutz, G., & Binder, H. 2005. Localized Classification. *Statistics and Computing*, **15**, 155–166.

Vaid, T.P., Burl, M.C., & Cervis, N.S. 2001. Comparison of the Performance of Different Discriminant Algorithms in Analyte Discrimination Tasks using an Array of Carbon Black-Polymer Composite Vapor Detectors. *Analytical Chemistry*, **73**(2), 321–331.

van Laarhoven, P.J., & Aarts, E.H. 1987. *Simulated Annealing: Theory and Applications*. Kluwer Academic Publishers: Boston.

Weihs, C., & Jessenberger, J. 1999. *Statistische Methoden zur Qualitätssicherung und -optimierung in der Industrie*. Wiley-VCH Verlag: Weinheim.

Wild, N., Karl, J., Grunert, V. P., Schmitt, R. I., Garczarek, U., Krause, F., Hasler, F., Van-Riel, P. L. C. M., Bayer, P. M., Thun, M., Mattey, D. L., Sharif, M., & Zolg, W. 2008. Diagnosis of rheumatoid arthritis: multivariate analysis of biomarkers. *Biomarkers*, **13**(1), 88–105.

Wild, Norbert, Karl, J., & Grunert, V.P. 2003. *Multivariate Analysis in Rheumatoid Arthritis vs. Controls*. Roche Diagnostics GmbH, New Technologies.

Xu, Q.-S., Liang, Y.-Z., & Du, Y.-P. 2004. Monte Carlo Cross-Validation for Selecting the Model and Estimating the Prediction Error in Multivariate Calibration. *Journal of Chemometrics*, **18**, 112–120.

Yee, T. W. 2007. *VGAM: Vector Generalized Linear and Additive Models*. R package version 0.7-3.

Appendix A

Proofs

A.1 Proofs to Tables 5.4 and 5.5

First of all, the reader is reminded that by convention $f_t > 1$ and $f_d < f_t$, or equivalently, $\rho^* < 0$. Before starting with individual proofs for the cases presented in Tables 5.4 and 5.5, we make some helpful remarks on the behavior of function G.

An investigation of G with respect to r is easier than with respect to any other of its arguments. Here $r \neq 1$, since the case when $r = 1$ is analyzed separately and according to Proposition 5.2.1 it leads to counterproductive weighted error estimates.

G can be expressed in a simplified way as:

$$G(p_{21}, r, \rho^*, f_t) = \frac{\tilde{a}}{(r-1)^2}[\tilde{b} - \tilde{c}r(1 - rp_{21})] \tag{A.1.1}$$

with

$$\begin{aligned}
\tilde{a}(p_{21}, \rho^*, f_t) =& \left(\frac{1 - p_{21}}{p_{21}}\right)\left[\frac{1 + (1 + \rho^*)f_t}{\rho^* f_t}\right] < 0\\
\tilde{b}(\rho^*, f_t) =& (2 + \rho^*)f_t + 2 > 0\\
\tilde{c}(p_{21}, \rho^*, f_t) =& \left(\frac{1}{1 - p_{21}}\right)\left[\frac{2(1 + \rho^*)f_t + 2 + \rho^*}{(1 + \rho^*)}\right] > 0
\end{aligned}$$

and the constraint

$$p_{22} = rp_{21} \leq 1.$$

Remark A.1.1.

(i) Function $\tilde{a}$ is always negative and strictly monotonically increasing with respect to p_{21} and f_t, while it is strictly monotonically decreasing with respect to ρ^*.

(ii) Function $\tilde{b}$ is positive and strictly monotonically increasing with respect to both ρ^* and f_t.

(iii) Function $\tilde{c}$ is always positive, but has the same monotonicity behavior like function $\tilde{a}$.

(iv) $\forall p_{21} \in [0, 1]$, $\forall \rho^* < 0$, $\forall f_t > 1$, it holds $\tilde{b} < (1 - p_{21})\tilde{c}$.

The form of G from (A.1.1) given fixed values of $p_{21} > 0$, $\rho^* < 0$, and $f_t > 1$, shows actually the restriction of G in the argument r:

$$G_{|p_{21},\rho^*,f_t}(r) = \frac{\tilde{a}\tilde{c}p_{21}r^2 - \tilde{a}\tilde{c}r + \tilde{a}\tilde{b}}{(r-1)^2}.$$

This form of G is useful to investigate its behavior with respect to r. From Remark A.1.1 (iv) it clearly results that:

$$4p_{21}\tilde{b} < 4p_{21}(1 - p_{21})\tilde{c} \geq 4 \cdot 0.25\tilde{c} = \tilde{c}.$$

Consequently, $G_{|p_{21},\rho^*,f_t}(r)$, has exactly two zeros, namely at:

$$r_0(p_{21}, \rho^*, f_t) = \frac{1 - \sqrt{1 - 4p_{21}\tilde{b}\tilde{c}^{-1}}}{2p_{21}} < 1 \qquad r_1(p_{21}, \rho^*, f_t) = \frac{1 + \sqrt{1 - 4p_{21}\tilde{b}\tilde{c}^{-1}}}{2p_{21}} > 1. \quad \text{(A.1.2)}$$

Using again Remark A.1.1(iv), it is easy to prove that $r_0 \in (0, 1)$, while $r_1 > 1$. E.g. it holds:

$$1 - 4p_{21}\tilde{b}\tilde{c}^{-1} > 1 - 4p_{21}(1 - p_{21}) \Leftrightarrow 1 - 4p_{21}\tilde{b}\tilde{c}^{-1} > (1 - 2p_{21})^2,$$

and therefore, if $p_{21} < 0.5$:

$$\sqrt{1 - 4p_{21}\tilde{b}\tilde{c}^{-1}} > 1 - 2p_{21}.$$

If $p_{21} \geq 0.5$ this inequality holds anyway. Therefore,

$$r_0 - 1 = \frac{1 - 2p_{21} - \sqrt{1 - 4p_{21}\tilde{b}\tilde{c}^{-1}}}{2p_{21}} < 0 \Leftrightarrow r_0 < 1, \forall p_{21} \in (0, 1].$$

Remark A.1.2.

(i) Both r_0 and r_1 are strictly monotonically decreasing with respect to p_{21}.

(ii) The quotient $\tilde{b}\tilde{c}^{-1}$ is strictly monotonically increasing both in ρ^* and f_t; therefore, r_0 is monotonically increasing and r_1 monotonically decreasing with respect to these two arguments.

(iii) Both r_0 and r_1 satisfy the constraint $rp_{21} \leq 1$.

Proof.

(i) We prove here only the monotonicity of r_0. In a similar way results the monotonicity of r_1, too. We denote the part of $\tilde{c}$ which is independent of p_{21} by $\tilde{c}'$, thus $\tilde{c}' = \tilde{c}(1 - p_{21})$. From Remark A.1.1 (iv) it is clear that $\tilde{b}\tilde{c}'^{-1} < 1$.

The first derivative of r_0 with respect to p_{21} is:

$$\begin{aligned}\frac{\partial r_0}{\partial p_{21}} &= -\frac{1}{2p_{21}^2} - \frac{1}{2\sqrt{\frac{1}{4p_{21}^2} - \frac{\tilde{b}\tilde{c}'^{-1}(1-p_{21})}{p_{21}}}}\left(-\frac{2}{4p_{21}^3} + \frac{\tilde{b}\tilde{c}'^{-1}}{p_{21}^2}\right) \\ &= -\frac{1}{2p_{21}^2}\left[1 - \frac{1 - 2p_{21}\tilde{b}\tilde{c}'^{-1}}{\sqrt{1 - 4p_{21}(1 - p_{21})\tilde{b}\tilde{c}'^{-1}}}\right]\end{aligned} \tag{A.1.3}$$

We need to show that the term in brackets is positive, which is equivalent to proving that:

$$1 - 2p_{21}\tilde{b}\tilde{c}'^{-1} < \sqrt{1 - 4p_{21}(1 - p_{21})\tilde{b}\tilde{c}'^{-1}}.$$

If $2p_{21}\tilde{b}\tilde{c}'^{-1} > 1$, then the inequality is obviously accomplished. Otherwise, the inequality is equivalent to:

$$(1 - 2p_{21}\tilde{b}\tilde{c}'^{-1})^2 < 1 - 4p_{21}(1 - p_{21})\tilde{b}\tilde{c}'^{-1} \Leftrightarrow \tilde{b}\tilde{c}'^{-1} > (\tilde{b}\tilde{c}'^{-1})^2 \Leftrightarrow 0 < \tilde{b}\tilde{c}'^{-1} < 1$$

and the last double inequality is true.

(ii) It is enough to show that the function

$$\tilde{b}\tilde{c}'^{-1} = \frac{(1 + \rho^*)[2 + (2 + \rho^*)f_t]}{[2(1 + \rho^*)f_t + 2 + \rho^*]}$$

is monotonically increasing both with respect to ρ^* and f_t.

Its first derivative with respect to ρ^* is:

$$\frac{\partial\tilde{b}\tilde{c}'^{-1}}{\partial\rho^*} = \frac{[-2(1 + f_t)^2 + (f_t + 2f_t^2)\rho^* + (f_t + 2f_t^2)\rho^{*2}]}{[2(1 + \rho^*)f_t + 2 + \rho^*]^2},$$

It is easy to prove that the quadratic term in the numerator is positive for every f_t ($\Delta = -4(f_t + 2f_t^2)(2 + 3f_t) < 0,\ \forall f_t > 1$). Thus, this derivative is positive, from where the increasing monotonicity of $\tilde{b}\tilde{c}^{-1}$ with respect to ρ^* follows.

The first derivative of $\tilde{b}\tilde{c}'^{-1}$ with respect to f_t is:

$$\frac{\partial\tilde{b}\tilde{c}'^{-1}}{\partial f_t} = (1 + \rho^*)\frac{4\rho^{*2}}{[2(1 + \rho^*)f_t + 2 + \rho^*]^2} > 0,$$

which proves the increasing monotonicity of $\tilde{b}\tilde{c}^{-1}$ *with respect to* f_t*, too.*

(iii) Since $r_0 \in (0, r_1)$*, it is sufficient to prove that* $r_1 p_{21} \leq 1$*. This is equivalent to showing that the inequality:*

$$\sqrt{1 - 4p_{21}\tilde{b}\tilde{c}^{-1}} < 1$$

holds, which is trivial.

Further, using Remark A.1.2(iii), the definition interval for r_1 given some fixed value of $p_{21} \leq 1$ is $(1, p_{21}^{-1}]$, like specified in Table A.1.
According to condition (5.2.13), the weighting benefit is achieved only if

$$G < N_2, \tag{A.1.4}$$

Obviously, when G is negative, the benefit of weighted error estimates upon the unweighted ones is sure. Table A.1 indicates the sign of the first derivative of function G with respect to r, which is useful to understand the monotonicity of G with respect to r and therefore, its sign, too. Using the form of G from (A.1.1), its first derivative with respect to r is given by:

$$\frac{\partial G}{\partial r} = \left(-\frac{\tilde{a}}{(r-1)^3}\right)\left[2\tilde{b} - \tilde{c} - r(1 - 2p_{21})\tilde{c}\right], \tag{A.1.5}$$

where the term in parentheses is negative for $r < 1$, and positive for $r > 1$, since $\tilde{a} < 0$.

Consequently, like Table A.1 highlights, the sign of G is decided by the sign of the expression in brackets, thus of $2\tilde{b} - \tilde{c} - r(1 - 2p_{21})\tilde{c}$.
Now, the sign of this expression is clearly negative when $p_{21} \leq 0.5$ and $r > 1$, or when $p_{21} > 0.5$ and $r < 1$.
When $p_{21} \leq 0.5$ and $r > 1$, starting from Remark A.1.1(iv) this can be seen as it follows:

$$\tilde{b} < (1-p_{21})\tilde{c} \Leftrightarrow 2\tilde{b} - \tilde{c} - r(1-2p_{21})\tilde{c} < 2(1-p_{21})\tilde{c} - \tilde{c} - r(1-2p_{21})\tilde{c} = \underbrace{(1-r)}_{<0}\underbrace{(1-2p_{21})}_{\geq 0}\tilde{c} \leq 0,$$

since $\tilde{c} > 0$.

This means that the derivative of G with respect to r is negative for $p_{21} \leq 0.5$ and $r > 1$, and positive for $p_{21} > 0.5$ and $r < 1$, like it is specified in Table A.1 in the first two rows. This indicates the decreasing monotonicity of G with respect to r when $p_{21} \leq 0.5$ and $r > 1$, and its increasing monotonicity when $p_{21} > 0.5$ and $r < 1$.

The restriction of G to fixed values of p_{21}, ρ^*, and f_t was denoted as $G_{|p_{21},\rho^*,f_t}$. The r-intervals on which $G_{|p_{21},\rho^*,f_t}$ is positive or negative presented in the last row of Table A.1, are derived using its monotonicity (see the sign of its derivative with respect to r in Table A.1) and the observations:

$$G_{|p_{21},\rho^*,f_t}(r=0) = \tilde{a}\tilde{b} < 0,$$

$$\lim_{r\to 1} G_{|p_{21},\rho^*,f_t}(r) = \infty,$$

and

$$G_{|p_{21},\rho^*,f_t}(r = p_{21}^{-1}) = \tilde{a}\tilde{b}\left(\frac{p_{21}}{1-p_{21}}\right)^2 < 0.$$

The last row in Table A.1 indicates that, given fixed values of p_{21}, ρ^*, and f_t, the weighting benefit is sure for r values in $[0, r_0]$ and $[r_1, p_{21}^{-1})$, where $G_{|p_{21},\rho^*,f_t}$ has a negative sign. On these domains of r, the benefit condition (A.1.4) is accomplished independently of the size of the heterogeneous class N_2. Ideally, r_0 and r_1 would approach 1.

Only on an interval around 1, which depends on N_2, the condition (A.1.4) is not accomplished. The values of $G_{|p_{21},\rho^*,f_t}$ explode if r approaches 1, which means that the more similar the subclass error probabilities p_{21} and p_{22} become, the less probable is to obtain a benefit by weighting.

r	$[0$		r_0		$1)$	$(1$		r_1		$p_{21}^{-1}]$
$\text{sgn}(\frac{\partial G}{\partial r})$, if $p_{21} \le 0.5$	$(-)\cdot\text{sgn}[2\tilde{b} - \tilde{c} - r(1-2p_{21})\tilde{c}]$					$-$	$-$	$-$	$-$	$-$
$\text{sgn}(\frac{\partial G}{\partial r})$, if $p_{21} > 0.5$	$+$	$+$	$+$	$+$	$+$	$(+)\cdot\text{sgn}[2\tilde{b} - \tilde{c} - r(1-2p_{21})\tilde{c}]$				
$\text{sgn}(G)$	$\tilde{a}\tilde{b}$	$-$	0	$+$	∞	∞	$+$	0	$-$	$\tilde{a}\tilde{b}\left(\frac{p_{21}}{1-p_{21}}\right)^2$

Table A.1: Sign of the first derivative and monotonicity behavior of $G_{|p_{21},\rho^*,f_t}(r)$ with respect to r.

The next theoretical result derives the monotonicity behavior of G with respect to r on the domains on which this is not clarified by Table A.1. Thus, when $p_{21} \le 0.5$ and $r < 1$, and when $p_{21} > 0.5$ and $r > 1$. This result enables us to prove that the larger the size N_2 of the heterogenous class, the smaller the r-interval around 1, on which weighting provides no benefit (i.e. on which $G \ge N_2$).

Proposition A.1.1. Function $G_{|p_{21},\rho^*,f_t}$ is always strictly monotonically increasing on its positivity domain below 1 (i.e. on $[r_0, 1)$) and strictly monotonically decreasing on its positivity domain above 1 (i.e. on $(1, r_1]$).

Proof. *According to (A.1.5), the derivative of G with respect to r has a unique zero at* $r^* = \frac{2\tilde{b}\tilde{c}^{-1}-1}{1-2p_{21}}$. *We regard separately two cases:*

(1. Case) $p_{21} > 0.5$

In this case we need the position of r^* *with respect to* r_1 *(see the second row in Table A.1). Remark A.1.1(iv) yields:*

$$\tilde{b}\tilde{c}^{-1} < 1 - p_{21} < 0.5.$$

Further, it holds:

$$r^* > r_1$$

since

$$\frac{\overbrace{2\tilde{b}\tilde{c}^{-1} - 1}^{<0}}{\underbrace{1 - 2p_{21}}_{<0}} > \frac{1 + \sqrt{1 - 4p_{21}\tilde{b}\tilde{c}^{-1}}}{2p_{21}} \Leftrightarrow$$

$$4p_{21}\tilde{b}\tilde{c}^{-1} - 2p_{21} < 1 - 2p_{21} + (1 - 2p_{21})\sqrt{1 - 4p_{21}\tilde{b}\tilde{c}^{-1}} \Leftrightarrow \sqrt{1 - 4p_{21}\tilde{b}\tilde{c}^{-1}} > \underbrace{2p_{21} - 1}_{>0} \Leftrightarrow$$

$$\tilde{b}\tilde{c}^{-1} < 1 - p_{21}.$$

The last inequality is true as stated by Remark A.1.1(iv).
Since $p_{21} > 0.5$, *the term in brackets from (A.1.5) is negative for every* $r < r^*$. *Using Table A.1, it results that:*

$$sgn(\frac{\partial G}{\partial r}) = (+)[-] = -,$$

for every $r \in (1, r^*)$. *When* $p_{21} \leq 0.5$, *this derivative has the same sign on* $(1, r^*)$ *as the first row in Table A.1 shows. Thus, independently of* p_{21}, $G_{|p_{21},\rho^*,f_t}$ *is strictly monotonically decreasing at least on* $(1, r_1] \subset (1, r^*)$, *which represents its positivity domain above* 1 *(see the last row in Table A.1).*

(2. Case) $p_{21} \leq 0.5$

Recall that r^* *was* $\frac{2\tilde{b}\tilde{c}^{-1}-1}{1-2p_{21}}$. *In this case we need the position of* r^* *with respect to* r_0 *(see the first row in Table A.1). Obviously, if* $\tilde{b}\tilde{c}^{-1} \leq 0.5$, *then* $r^* < 0$, *and therefore* $r^* < r_0$, *too.*
If $\tilde{b}\tilde{c}^{-1} > 0.5$, *then* $r^* > 0$ *and it holds again:*

$$r^* < r_0,$$

since

$$\frac{2\tilde{b}\tilde{c}^{-1}-1}{1-2p_{21}} < \frac{1-\sqrt{1-4p_{21}\tilde{b}\tilde{c}^{-1}}}{2p_{21}} \Leftrightarrow 4p_{21}\tilde{b}\tilde{c}^{-1} - 2p_{21} < 1-2p_{21} - (1-2p_{21})\sqrt{1-4p_{21}\tilde{b}\tilde{c}^{-1}}$$

$$\Leftrightarrow 1-4p_{21}\tilde{b}\tilde{c}^{-1} > (1-2p_{21})\sqrt{1-4p_{21}\tilde{b}\tilde{c}^{-1}} \Leftrightarrow \sqrt{1-4p_{21}\tilde{b}\tilde{c}^{-1}} > \underbrace{1-2p_{21}}_{>0} \Leftrightarrow \tilde{b}\tilde{c}^{-1} < 1-p_{21},$$

and the last inequality is always accomplished as stated by Remark A.1.1(iv).
Therefore $[r_0, 1) \subset (r^*, 1)$ *always in this case.*
Since $p_{21} \leq 0.5$*, the term in brackets from (A.1.5) is negative for every* $r > r^*$*. Using Table A.1, it results that:*

$$sgn(\frac{\partial G}{\partial r}) = (-)[-] = +,$$

for every $r \in (r^*, 1)$*. When* $p_{21} > 0.5$*, this derivative has the same sign on* $(r^*, 1)$ *as the second row in Table A.1 shows. Thus, independently of* p_{21}*,* $G_{|p_{21},\rho^*,f_t}$ *is strictly monotonically increasing at least on* $[r_0, 1) \subset (r^*, 1)$*, which represents its positivity domain below* 1.

A precise description of the disadvantageous neighborhood of r is possible by solving the equation

$$G_{|p_{21},\rho^*,f_t}(r) = N_2.$$

Since $G_{|p_{21},\rho^*,f_t}$ is continuous to the left and right of 1 and using its limits when $r \to 1$ shown in the last row of Table A.1, this equation has exactly one solution within $(r_0, 1)$ and one solution within $(1, r_1)$. Explicitly, these solutions are:

$$r_2^{-,+} = \frac{2N_2 - \tilde{a}\tilde{c} \mp \sqrt{\Delta}}{2N_2 - (2p_{21})\tilde{a}\tilde{c}} \tag{A.1.6}$$

where

$$\Delta = \tilde{a}[\tilde{a}\tilde{c}(\tilde{c} - 4\tilde{b}p_{21}) + 4N_2(\tilde{b} - \tilde{c}(1-p_{21}))],$$

and the superscripts −, + indicate the middle operator ∓.

Note 3. Due to the monotonicity behavior of G asserted by Proposition A.1.1, the larger the size N_2 of the heterogeneous class is, the smaller the interval (r_2^-, r_2^+) gets, where weighting provides no benefit.

These observations prove again, that the safest situation for weighting is when the subclass misclassification probabilities are very different. However, the larger the sample size of the heterogenous class is, the less important this issue becomes.

Proof to Table 5.4

Proof. *When $r < 1$, Table A.1 indicates that function G is negative on $[0, r_0]$. Thus, given a fixed combination of p_{21}, ρ^* and f_t, the benefit condition (A.1.4) is accomplished, and the weighting benefit is sure for every r below $r_0(p_{21}, \rho^*, f_t)$.*

If the arguments p_{21}, ρ^* and f_t are varied over some range of values $\mathcal{R}$, then let r_0^{min} be the minimal value of r_0 over this range. It holds:

$$G_{|p_{21},\rho^*,f_t}(r) < 0,\ \forall r \leq r_0(p_{21}, \rho^*, f_t),\ \forall (p_{21}, \rho^*, f_t) \in \mathcal{R},$$

and therefore,

$$G_{|p_{21},\rho^*,f_t}(r) < 0,\ \forall r \leq r_0^{min} = \min_{\mathcal{R}} r_0(p_{21}, \rho^*, f_t).$$

Using the properties of decreasing monotonicity of r_0 with respect to p_{21}, and of increasing monotonicity with respect to ρ^* and f_t, stated by Remark A.1.2 (i)-(ii), the minimal r_0-values listed in Table A.2 are obtained, starting from different definition domains for (p_{21}, ρ^*, f_t). E.g., at row 2, $\mathcal{R} = (0, 0.4] \times [-0.7, -0.5) \times [1, 2)$, and therefore:

$$r_0^{min} = \min_{\mathcal{R}} r_0(p_{21}, \rho^*, f_t) = r_0(0.4, -0.7, 1) = 0.37.$$

This indicates that, given an error probability p_{21} between 0% and 40% in the less prevalent target subclass, a medium to high degree of suboptimality of the data at hand $\rho^* \in [-0.7, -0.5)$, and any degree of unbalance of the target subclass structure greater than 1, the weighting benefit is sure up to an error probability $p_{22} = 0.37 p_{21}$ in the preponderant target subclass.

Rows 1 – 12, **Table 5.4**	p_{21}	$\rho^* = \frac{f_d - f_t}{f_t}$	$f_t = \frac{\pi_{22}}{\pi_{21}}$	**minimal** $r_0(p_{21}, \rho^*, f_t)$
1	$(0, 0.5]$	$[-0.5, 0)$	$[1, 2)$	$r_0(0.5, -0.5, 1) = 0.45$
2	$(0, 0.4]$	$[-0.7, -0.5)$	$[1, 2)$	$r_0(0.4, -0.7, 1) = 0.37$
3	$(0, 0.5]$	$[-0.5, 0)$	$[2, 99)$	$r_0(0.5, -0.5, 2) = 0.47$
4	$(0, 0.4]$	$[-0.7, -0.5)$	$[2, 99)$	$r_0(0.4, -0.7, 2) = 0.39$
5	$(0, 0.5]$	$[-0.5, 0)$	$[99, \infty)$	$r_0(0.5, -0.5, 99) = 0.50$
6	$(0, 0.5]$	$[-0.7, -0.5)$	$[99, \infty)$	$r_0(0.5, -0.7, 99) = 0.40$
7	$(0.5, 0.6]$	$[-0.25, 0)$	$[1, 2)$	$r_0(0.6, -0.25, 1) = 0.49$
8	$(0.5, 0.6]$	$[-0.6, -0.5)$	$[1, 2)$	$r_0(0.6, -0.6, 1) = 0.30$
9	$(0.5, 0.6]$	$[-0.25, 0)$	$[2, 99)$	$r_0(0.6, -0.25, 2) = 0.49$
10	$(0.5, 0.55]$	$[-0.7, -0.5)$	$[2, 99)$	$r_0(0.55, -0.7, 2) = 0.30$
11	$(0.5, 0.6]$	$[-0.5, 0)$	$[99, \infty)$	$r_0(0.6, -0.5, 99) = 0.39$
12	$(0.5, 0.6]$	$[-0.7, -0.5)$	$[99, \infty)$	$r_0(0.6, -0.7, 99) = 0.32$

Table A.2: Values of r_0^{min} for specified domains of p_{21}, ρ^* and f_t. r_0^{min} *is also the maximal value of* $r < 1$ *for which* G *is negative; therefore, the benefit is sure under the given definition domain of the arguments.*

Proof to Table 5.5

Proof. *Using the form (A.1.1) of the G-function with the notations:*

$$a =: \tilde{a}\frac{p_{21}}{1 - p_{21}}$$

$$c =: \tilde{c}(1 - p_{21})$$

it results

$$G(p_{21}, r, \rho^*, f_t) = \frac{a}{(r-1)^2}\left(\frac{1 - p_{21}}{p_{21}}\right)\left[\tilde{b} - cr\left(\frac{1 - rp_{21}}{1 - p_{21}}\right)\right]. \qquad \text{(A.1.7)}$$

Here a, c *are preferred to* $\tilde{a}$ *and* $\tilde{c}$*, respectively, since they are independent of* p_{21}*. Further, one should recall that* $p_{22} = rp_{21}$ *and* $r \geq 2$*, in this case.*

The weighting benefit is sure, if

$$\tilde{b} - cr\left(\frac{1 - rp_{21}}{1 - p_{21}}\right) \geq 0 \Leftrightarrow \tilde{b} - cr^2\left(\frac{1 - p_{22}}{r - p_{22}}\right) \geq 0 \Leftrightarrow \frac{1 - p_{22}}{r - p_{22}} \leq \frac{\tilde{b}}{cr^2}$$

$$\Leftrightarrow p_{22} \geq p_{22}^0(r, \rho^*, f_t) =: \frac{r\left(\frac{\tilde{b}}{cr^2}\right) - 1}{\frac{\tilde{b}}{cr^2} - 1}.$$

The last inequality holds, since:

$$[\tilde{b} < (1 - p_{21})\tilde{c} = c] \wedge (r > 1) \Longrightarrow \frac{\tilde{b}}{cr^2} < 1.$$

Thus, for given values of r, ρ^* and f_t, weighting provides a sure benefit if p_{22} exceeds a certain threshold $p_{22}^0(r, \rho^*, f_t)$.

If the arguments r, ρ^* and f_t are varied over some range of values $\mathcal{R}'$, then let p_{22}^{max} be the maximal value of p_{22}^0 over $\mathcal{R}'$. It holds:

$$G < 0,\ \forall p_{22} \geq p_{22}^0(r, \rho^*, f_t),\ \forall (r, \rho^*, f_t) \in \mathcal{R}',$$

and therefore:

$$G < 0,\ \forall p_{22} \geq p_{22}^{max} = \max_{\mathcal{R}'} p_{22}^0(r, \rho^*, f_t).$$

Function p_{22}^0 is obviously monotonically increasing with respect to r. Also, according to Remark A.1.2(ii), $\tilde{b}\tilde{c}^{-1}$ and therefore, also $\frac{\tilde{b}}{cr^2}$, is strictly monotonically increasing with respect to ρ^* and f_t. Now:

$$p_{22}^0 = f(y)$$

where

$$f(y) = \frac{ry - 1}{y - 1}$$

is monotonically decreasing with respect to y, and

$$y = \frac{\tilde{b}}{cr^2}$$

is monotonically increasing with respect to ρ^* and f_t. Therefore, p_{22}^0 is monotonically decreasing with respect to ρ^* and f_t.

The maximal values of p_{22}^0 listed in Table A.3 are computed using the monotonicity of p_{22}^0 with respect to its three arguments. E.g. in the last row, the definition range for its arguments is

$\mathcal{R}' = [2,5] \times [-0.8,-0.5) \times [99,\infty)$. *Hence:*

$$p_{22}^{max} = \max_{\mathcal{R}'} p_{22}^0(r,\rho^*,f_t) = p_{22}^0(5,-0.8,99) = 0.90.$$

This indicates that, given a quotient r between the subclass error probabilities p_{22} and p_{21} within $[2,5]$, a medium to high degree of suboptimality of the data at hand $\rho^* \in [-0.8,-0.5)$, and an extremely high degree of unbalance of the target subclass structure $f_t \geq 99$, the weighting benefit is sure starting from an error probability of 90% in the preponderant target subclass.

Rows $1-6$, **Table 5.5**	r	$\rho^* = \frac{f_a - f_t}{f_t}$	$f_t = \frac{\pi_{22}}{\pi_{21}}$	**maximal** $p_{22}^0(r,\rho^*,f_t)$
1	$[2,5]$	$[-0.2,0)$	$[1,2)$	$p_{22}^0(5,-0.2,1) = 0.85$
2	$[2,4.5]$	$[-0.7,-0.5)$	$[1,2)$	$p_{22}^0(4.5,-0.7,1) = 0.91$
3	$[2,5]$	$[-0.2,0)$	$[2,99)$	$p_{22}^0(5,-0.2,2) = 0.85$
4	$[2,4.5]$	$[-0.7,-0.5)$	$[2,99)$	$p_{22}^0(4.5,-0.7,2) = 0.91$
5	$[2,5]$	$[-0.2,0)$	$[99,\infty)$	$p_{22}^0(5,-0.2,99) = 0.85$
6	$[2,5]$	$[-0.8,-0.5)$	$[99,\infty)$	$p_{22}^0(5,-0.8,99) = 0.90$

Table A.3: Values of p_{22}^0 for specified domains of r, ρ^* and f_t. p_{22}^{max} *is also the minimal value of $p_{22} > 0.5$ for which G is negative; therefore, the benefit is sure under the given definition domain of the arguments.*

Case: $r \geq 2$, $|\rho^*| > 0.5$, f_t arbitrary, and $p_{22} > 0.5$ (see rows 2, 4 and 6 from Table 5.5).
Statement: ***The closer the true error probability of the preponderant target subclass, p_{22}, to 1, the more probable the weighting benefit, too.***

Proof. *Consider $p_{22} \to 1$, or equivalently, $p_{21} \to r^{-1}$, which is easier to replace in the form of G from (A.1.7). Then it holds:*

$$\lim_{p_{22}\to 1} G = \frac{a\tilde{b}}{r-1} < 0.$$

Thus, when the misclassification probability of the under-represented subclass exceeds some high threshold and is larger than the misclassification probability of the over-represented subclass ($r > 1$, or equivalently, $p_{22} > p_{21}$), a benefit by weighting is guaranteed.

A.2 Other proofs

Proof (Remark 5.2.2)**.** *It holds:*

$$\begin{aligned}
\frac{\partial q_{min}}{\partial s} =& \frac{\partial}{\partial s}\left(\frac{\sqrt{1-4\sigma_1 s}-\sqrt{1-4\sigma_1}}{2\sqrt{\sigma_1}}\right)^2 \\
=& \frac{1}{4\sigma_1}\cdot 2(\sqrt{1-4\sigma_1 s}-\sqrt{1-4\sigma_1})\frac{1}{2\sqrt{1-4\sigma_1 s}}(-4\sigma_1) \\
=& -\frac{\sqrt{1-4\sigma_1 s}-\sqrt{1-4\sigma_1}}{\sqrt{1-4\sigma_1 s}} \\
=& -\left(1-\sqrt{\frac{1-4\sigma_1}{1-4\sigma_1 s}}\right).
\end{aligned}$$

When $s < 1$ *it results that:*

$$1-4\sigma_1 s > 1-4\sigma_1 > 0 \implies 1 > \sqrt{\frac{1-4\sigma_1}{1-4\sigma_1 s}} \implies \frac{\partial q_{min}}{\partial s} < 0.$$

When $s > 1$*, but* $s < (4\sigma_1)^{-1}$*, it holds:*

$$0 < 1-4\sigma_1 s < 1-4\sigma_1 \implies 1 < \sqrt{\frac{1-4\sigma_1}{1-4\sigma_1 s}} \implies \frac{\partial q_{min}}{\partial s} > 0.$$

This demonstrates the monotonicity behavior of q_{min} *with respect to* s *stated by Remark 5.2.2. Regarding* σ_1 *it holds:*

$$\begin{aligned}
\frac{\partial q_{min}}{\partial \sigma_1} =& \frac{\partial}{\partial \sigma_1}\left(\sqrt{\frac{1}{4\sigma_1}-s}-\sqrt{\frac{1}{4\sigma_1}-1}\right)^2 \\
=& 2\left(\sqrt{\frac{1}{4\sigma_1}-s}-\sqrt{\frac{1}{4\sigma_1}-1}\right)\cdot\frac{\partial}{\partial \sigma_1}\left(\sqrt{\frac{1}{4\sigma_1}-s}-\sqrt{\frac{1}{4\sigma_1}-1}\right) \\
=& -\frac{1}{4\sigma_1^2}\left(\sqrt{\frac{1}{4\sigma_1}-s}-\sqrt{\frac{1}{4\sigma_1}-1}\right)\cdot\left(\frac{1}{\sqrt{\frac{1}{4\sigma_1}-s}}-\frac{1}{\sqrt{\frac{1}{4\sigma_1}-1}}\right)
\end{aligned}$$

$$
\begin{aligned}
&= -\frac{1}{4\sigma_1^2}\left(\sqrt{\frac{1}{4\sigma_1}-s}-\sqrt{\frac{1}{4\sigma_1}-1}\right)\left(\frac{\sqrt{\frac{1}{4\sigma_1}-1}-\sqrt{\frac{1}{4\sigma_1}-s}}{\sqrt{\frac{1}{4\sigma_1}-1}\cdot\sqrt{\frac{1}{4\sigma_1}-s}}\right)\\
&=\frac{1}{4\sigma_1^2\sqrt{\frac{1}{4\sigma_1}-1}\cdot\sqrt{\frac{1}{4\sigma_1}-s}}\left(\sqrt{\frac{1}{4\sigma_1}-s}-\sqrt{\frac{1}{4\sigma_1}-1}\right)^2 > 0.
\end{aligned}
$$

This demonstrates that q_{min} is monotonically increasing with respect to σ_1.

Proof (Remark 5.2.3)**.** *It holds:*

$$
\begin{aligned}
\frac{\partial q_{max}}{\partial s} &=\frac{1}{4\sigma_1}2(\sqrt{1-4\sigma_1 s}+\sqrt{1-4\sigma_1})\frac{1}{2\sqrt{1-4\sigma_1 s}}(-4\sigma_1)\\
&=-(1+\sqrt{\frac{1-4\sigma_1}{1-4\sigma_1 s}}) < 0.
\end{aligned}
$$

Thus, q_{max} is a monotonically decreasing function of s. Also with respect to σ_1 it holds:

$$
\begin{aligned}
\frac{\partial q_{max}}{\partial \sigma_1} &=2\left(\sqrt{\frac{1}{4\sigma_1}-s}+\sqrt{\frac{1}{4\sigma_1}-1}\right)\cdot\frac{\partial}{\partial\sigma_1}\left(\sqrt{\frac{1}{4\sigma_1}-s}+\sqrt{\frac{1}{4\sigma_1}-1}\right)\\
&=-\frac{1}{4\sigma_1^2\sqrt{\frac{1}{4\sigma_1}-s}\sqrt{\frac{1}{4\sigma_1}-1}}\left(\sqrt{\frac{1}{4\sigma_1}-s}+\sqrt{\frac{1}{4\sigma_1}-1}\right)^2 < 0.
\end{aligned}
$$

This proves that q_{max} is monotonically decreasing with respect to σ_1, too.

Appendix B

Figures

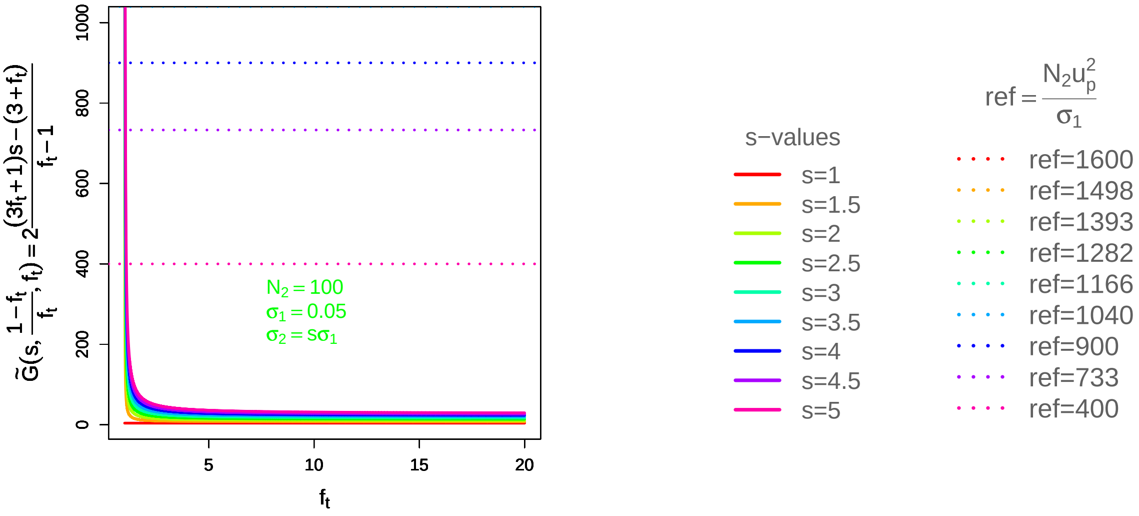


Figure B.1: Subcase 1: Weighted errors - relation between $\tilde{G}(s, \frac{1-f_t}{f_t}, f_t)$ and the *best case* reference when $s \geq 1$, $\sigma_1 = 0.05$, $N_2 = 100$, and $f_t \leq 20$.

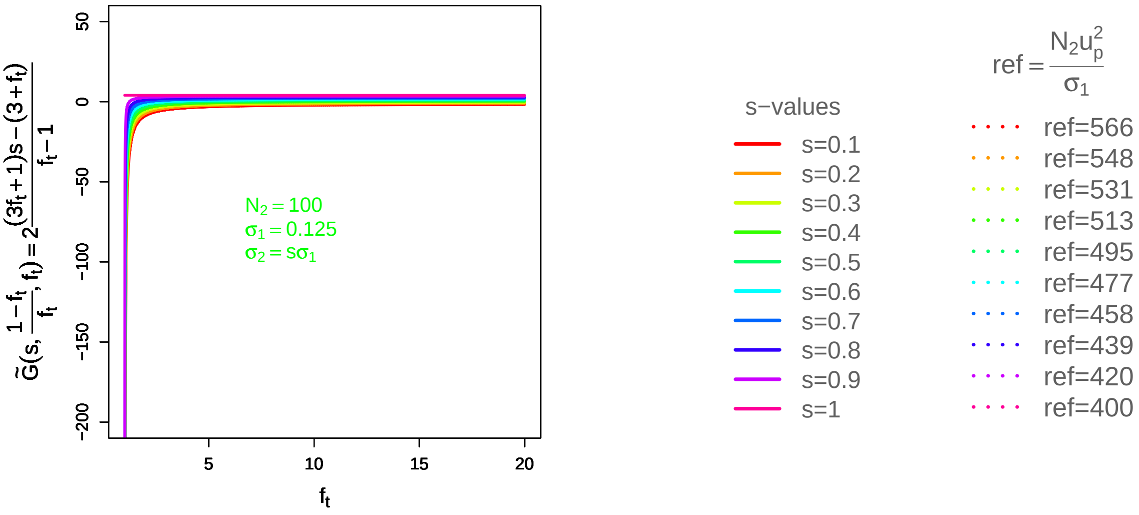


Figure B.2: Subcase 1: Weighted errors - relation between $\tilde{G}(s, \frac{1-f_t}{f_t}, f_t)$ and the *best case* reference when $s \leq 1$, $\sigma_1 = 0.125$, $N_2 = 100$, and $f_t \leq 20$.

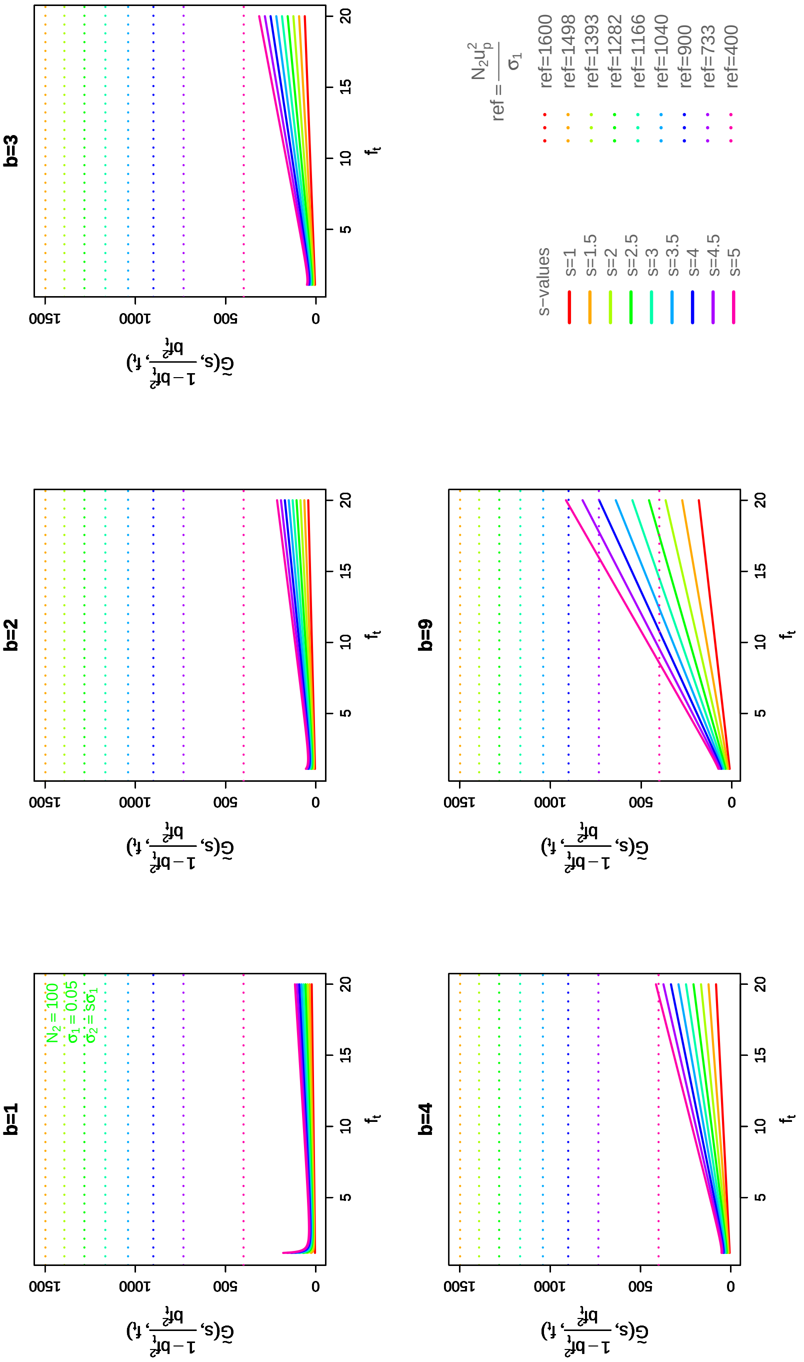


Figure B.3: Subcase 2: Weighted errors - relation between $\tilde{G}(s, \frac{1-bf_t^2}{bf_t^2}, f_t)$ and the *best case* reference when $s \geq 1$, $\sigma_1 = 0.05$, $N_2 = 100$, and $f_t \leq 20$. *When the error probability* p_{22} *of the preponderant target subclass is an element of* $(0.5, 95]$ *the weighting benefit is sure for any* $b \geq 4$ *and* $f_t \geq 20$.

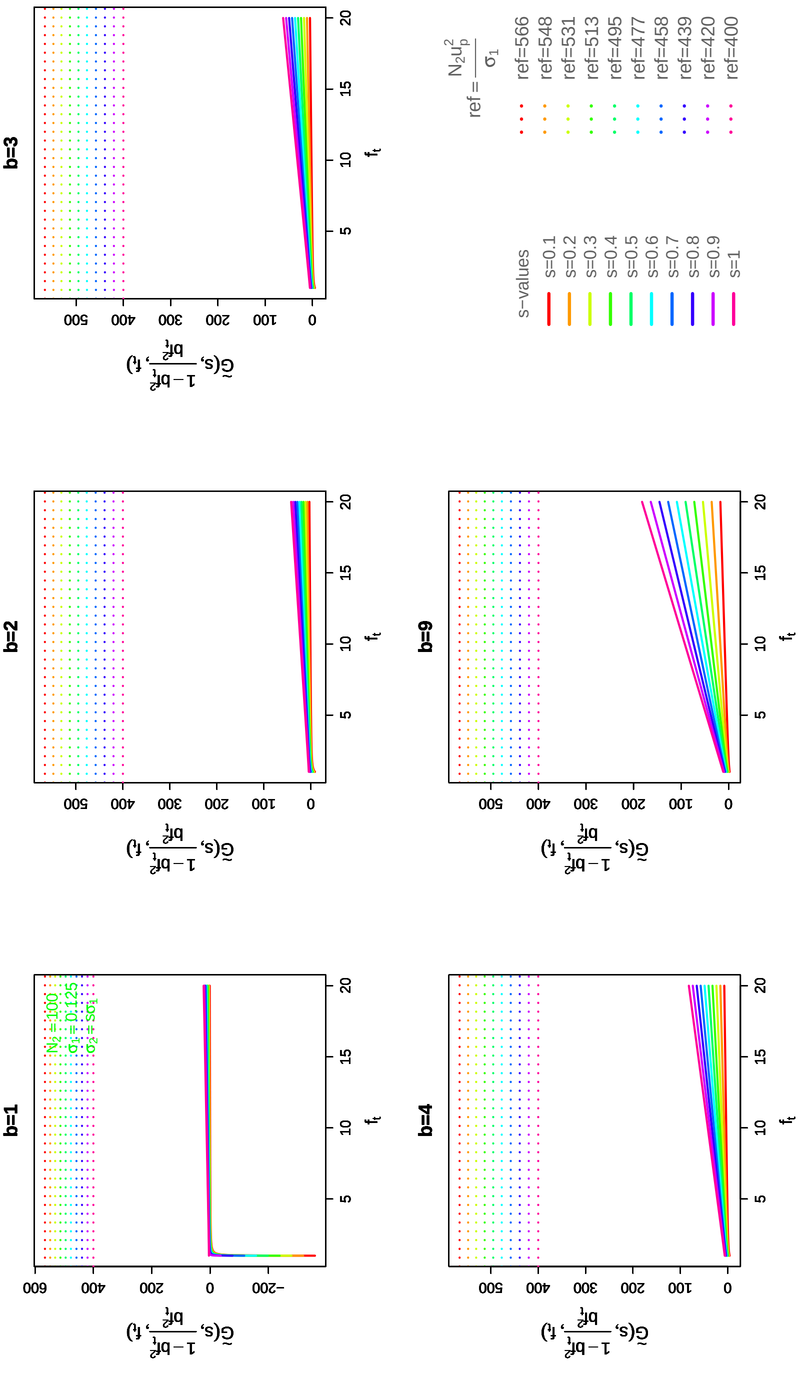


Figure B.4: Subcase 2: Weighted errors - relation between $\tilde{G}(s, \frac{1-bf_t^2}{bf_t^2}, f_t)$ and the *best case* reference when $s \leq 1$, $\sigma_1 = 0.125$, $N_2 = 100$, and $f_t \leq 20$. *When the error probability* p_{22} *of the preponderant target subclass is an element of* $[0.85, 1]$ *the weighting benefit is sure for any* $f_t \leq 20$ *and any* $b \geq 9$.