Ensemble Learning by Data Resampling

Michael Goebel

A thesis submitted in partial fulfilment of the requirements for
the degree of Doctor of Philosophy in Computer Science,
The University of Auckland, 2004

Abstract

We investigate ensemble learning methods that construct a classifier ensemble by repeatedly sampling the original training data and building a member classifier from each subsample. We find that the performance of standard Bagging can frequently be improved upon by simple variations of the sampling scheme, such as varying the sample size, or sampling without replacement instead of sampling with replacement.

For all methods tested, the ensemble performance is greatly dependent on properties of the problem domain and the data sample. We try to explain the observed performances of the various ensemble methods qualitatively and quantitatively, and find that current ensemble analysis methods such as margin distributions, κ-Error Diagrams, bias-variance decomposition etc. are not well suited for this task.

We postulate that the primary explanation for the performance of ensemble methods is to be found in their effects on accuracy of the ensemble members on one side and diversity among them on the other side, two contradictory goals which necessitate a compromise, or trade-off. This motivates the presentation of a precise yet general definition of what diversity is and how it is to be measured. This definition has the desirable property that it is applicable to all single-stage voting ensembles, under any given loss function.

We then study the mathematical relationships between ensemble loss, mean member loss, and diversity. For squared loss, we show that our definitions lead to the well known ensemble loss decomposition, and extend this decomposition to the case where the ensemble members, instead of a real number, return a probability distribution over $\mathcal{R}$.

For the case of 0-1 loss, we derive the exact mathematical relations between ensemble loss, mean member loss, and diversity. Studying those relations provides some valuable insights into ensembles behavior, and produces some unexpected hence interesting results. These results are also confirmed by the experimental observations.

Turning our attention back to the performance of Bagging variants, we show how the loss decomposition can be used to reduce the number of parameter settings which have to be tried out experimentally in order to find a well-performing ensemble method for a given particular problem.

Acknowledgments

I am greatly indebted to my supervisory committee, Pat Riddle, Mike Barley, and Hans Guesgen, for innumerable helpful comments, as well as for their general guidance and support.

Another big 'Thank you' to all those who provided valuable comments on earlier versions of this thesis, especially to Remco Bouckaert.

Special thanks also goes to the Department of Computer Science at the University of Auckland for their financial, technical, and administrative support.

Appendix A lists those who provided the datasets used in the experiments.

Credits are also due to all those who contribute their time and energy to produce all these wonderful free ('free' as in free speech) software packages used for the conduct of the research as well as for the production of this thesis – you are way too many to mention individually, but way too important not to mention at all.

Lastly, to my parents, Rotraut and Manfred, and to my fiancee, María Cristina, as well as to her parents, María del Carmen and José Carlos, for their extraordinary patience, love, and support.

This would not have been possible without you.

Contents

List of Tables

List of Figures

Abbreviations

BVD	Bias-Variance Decomposition
Bagging	Bootstrap Aggregating
Cragging	Cross-Validation Aggregating
ECOC	Error-Correcting Output Coding
i.i.d.	independently and identically distributed
iff	if and only if
MCMC	Markov Chain Monte Carlo
vs.	versus
w.r.t.	with respect to

Notation

$a = b$	Equality of a and b
$a := b$	Definition of a as b; or assignment of a to value of b
$\forall$	For all (universal quantifier)
$f : A \to B$	Function f from A to B
$I()$	Indicator function
$\{a_1, \ldots, a_n\}$	Finite set or multi-set consisting of n elements $a_1, \ldots, a_n$
$\{a \vert \mathcal{P}(a)\}$	Set containing all elements a for which $\mathcal{P}(a)$ is true
$\emptyset$	Empty set
$\vert A \vert$	Cardinality (number of elements in set or tuple A)
$\vert a \vert$	Absolute value of numeric variable a
$A \times B$	Cartesian product of two sets A and B
A^n	$A \times A \times \cdots \times A$ (n times) iff A is a set,
	$A * A * \cdots * A$ (n times) iff A is a number
$\mathcal{R}$	The set of all real numbers
$\mathcal{N}$	The set of all natural numbers, including 0
$\mathcal{N}^+$	The set of all positive natural numbers
∞	Infinity
$\mathbf{a} = \langle a_1, \ldots, a_n \rangle$	Ordered tuple $\mathbf{a}$ consisting of n elements $a_1, \ldots, a_n$
$p(a)$	Probability that a is true iff a is a predicate
	Probability that $A = a$ iff A is a random variable
$P(A)$	Probability distribution of random variable A
$P(A, B)$	Joint probability distribution of random variables A and B
$P(A \vert B)$	Probability distribution of A conditioned B
$P(A \vert b)$	Probability distribution of A, given that $B = b$
$\{P(A)\}$	Set containing all probability distributions $P(A)$
$\{P(\mathcal{R})\}$	Set containing all probability distributions over $\mathcal{R}$
$E_{a \in A}[V_a]$	Expectation of random variable V taken over set A
$E_{P(c)}[V_c]$	Expectation of random variable V according to $P(c)$
$\mathbf{X}$	Input space
Y	Outcome space
$\widehat{Y}$	Ensemble prediction space
$\widehat{Y}_c$	Member prediction space
$\mathbf{s} = \langle \mathbf{x}, y \rangle$	A given input/outcome - pair (instance)
$\mathbf{S} = \langle \mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_m \rangle$	Multi-set (sample) of instances
$P(\mathbf{X}, Y)$	Probability distribution over instances

$p(\mathbf{x}, y)$	Probability of encountering instance $\langle \mathbf{x}, y \rangle$
C	Classifier or ensemble $C : \mathbf{X} \rightarrow \widehat{Y}$
J	Classifier inducer (learner)
l	Loss function $l : \widehat{Y} \times Y \rightarrow \mathcal{R}$
$l(\widehat{y}, y)$	Loss of prediction $\widehat{y}$ relative to outcome y
$l_2(\widehat{y}, y)$	Squared loss of prediction $\widehat{y}$ relative to outcome y
$l_{01}(\widehat{y}, y)$	0-1 loss of prediction $\widehat{y}$ relative to outcome y
$l_{\|\|}(\widehat{y}, y)$	absolute loss of prediction $\widehat{y}$ relative to outcome y
k	number of classes in discrete outcome space
m	sample size for sample of instances
n	number of classifiers in ensemble
c, C_i	Member classifiers
V	Voting function
$\mathbf{c} = \langle c_1, \ldots, c_n \rangle$	Tuple of member classifiers
$\mathbf{w} = \langle w_1, \ldots, w_n \rangle$	Tuple of voting weights
$\widehat{y}_C(\mathbf{x})$	Classifier prediction
$\widehat{y}(\mathbf{x})$	Ensemble prediction
$\widehat{\mathbf{y}}(\mathbf{x})$	Tuple of ensemble members' predictions
$\widehat{y}_c(\mathbf{x})$	Ensemble member prediction
$\widehat{P}(Y\|x)$	Belief distribution of probabilistic classifier for input x
$R(\widehat{y}\|x)$	Conditional risk of predicting $\widehat{y}$ for input x
$L(\mathbf{x}, y)$	Loss of ensemble for instance $\langle \mathbf{x}, y \rangle$
$\overline{L}(\mathbf{x}, y)$	Mean member loss of ensemble for instance $\langle \mathbf{x}, y \rangle$
$\overline{D}(\mathbf{x})$	Diversity of ensemble members for input x
L	Expected loss of ensemble for domain
$\overline{L}$	Expected mean member loss of ensemble for domain
$\overline{D}$	Expected diversity of ensemble for domain
$\mathcal{B}(s; n)$	*Bagging*$(s; n)$ (using n runs and relative sample size s)
$C(f; n)$	*Cragging*$(f; n)$ (using n runs and f folds)

1. Introduction

1.1 Scope

In recent years, there has been considerable interest in learners that produce models of small expected loss by generating and aggregating multiple individual models. It has been shown that ensemble methods often outperform single classifiers significantly, and the increase in available computing power has made the application of ensemble methods feasible even for large datasets ([18, 59, 69]).

A necessary and sufficient condition for an ensemble of classifiers to be more accurate than any of its individual members is assumed to be if the individual classifiers are accurate and diverse ([39]).

In this thesis we consider ensemble methods that construct a set of diverse classifiers by applying the same base learning algorithm to different subsets of the original training set, and then classify previously unseen examples by taking a (possibly weighted) vote of the predictions of the ensemble members. We call those methods resampling ensemble learning methods.

For the experimental part, we constrict ourselves to resampling methods for which the learning process is parallelizable – that is, the training set subsamples and the member classifiers can be constructed in parallel. The most well known such method is Bagging ([8]). Used in conjunction with C4.5 ([66]) as the base learner, vast improvements in performance over that of a single classifiers have been demonstrated on a wide variety of application domains, e.g. in [16, 59, 66].

1.2 Motivation

A number of attempts have been made to explain the success of Bagging and to analyze its performance theoretically. The most prominent such explanations are the Margin explanation ([71]) and the Bias-Variance explanation ([24, 48, 51]).

While those analyses and others are proving useful to some extent in supplying qualitative explanations for the success of ensemble learning methods, they still fail to provide accurate, quantitative diagnostic techniques for practical predictive analysis of ensemble performance ([18, 24, 40]). As a result, the design of ensemble learning

algorithms is – much like the design methods for classifier learning algorithms in general – still rather ad-hoc in nature.

In particular, although Bagging is by far the most popular sampling scheme for the training of ensembles with equally weighted classifier votes, to our knowledge nobody has yet stated concisely if and/or under which exact conditions Bagging can be shown to be the optimal sampling scheme for a given base learner. From the no-free-lunch theorems ([78, 79]) follows directly that there must exist situations in which standard Bagging is outperformed by other sampling schemes in terms of predictive performance – the question is just whether or not these situations arise frequently in practice.

Indeed, our experiments will show that Bagging can in fact be outperformed frequently not just in theory but also in practice by some simple variations of the sampling scheme, such as varying the bag size, or sampling without replacement instead of sampling with replacement.

Such variations often have the additional advantage of reducing the time needed to learn an ensemble. The speed advantages come mainly in two forms: First, the number of examples that is passed to the base learner in each iteration may sometimes be substantially reduced while maintaining or even increasing ensemble performance, and most base learning algorithms have a linear time complexity in terms of the number of training examples. Second, the performance of the ensemble as new classifiers are added may increase faster for some sampling schemes than for others, therefore less classifiers may have to be learned in order to achieve the same or even better performance than standard Bagging.

This opens the question of whether certain sampling schemes are performing uniformly better than others in certain situations, and if so, how one can recognize these situations without actually running all the different ensemble learning algorithms. To this end, we examine ensemble analysis methods currently in use, and find them unsuited to the task of predictive ensemble performance analysis.

It is commonly assumed that a good classifier ensemble is one whose members are both accurate and diverse, i.e., that member accuracy and diversity constitute necessary and sufficient conditions for an ensemble to obtain a low expected loss (e.g. [3, 17, 60, 65, 70]). However, member accuracy and diversity are quite contradictory goals: In order for the member classifiers to be diverse, they have to make mistakes. And if the member models are all accurate, their predictions will agree with each other.

Clearly, there is a trade-off one has to make when learning an ensemble of models: Compared to learning a single model, we will have to sacrifice some of its accuracy and instead generate a set of models which are somewhat less accurate on average but are diverse.

This poses a dilemma when designing or choosing an ensemble method for a given problem at hand: If we have a choice between several ensemble generation methods, should we choose one that generates very accurate and consequently highly corre-

lated models, or should we instead go for the one generating less accurate but more diverse models? How much accuracy should we sacrifice in order to gain how much diversity?

This trade-off is not yet well understood. While there is an abundance of empirical studies comparing various ensemble generation mechanisms on numerous learning problems, theoretical research in this area has been rather scarce. One of the reasons for this is the lack of a well-defined, universal, and quantitative definition of what exactly it means for a set of models to be diverse. Different measures of diversity are used for different loss functions and by different authors, e.g. in [18, 53, 64]. Accordingly, no unifying theory for analyzing the exact dependencies of the ensemble performance on the accuracy and diversity of its members exists ([43, 69]).

Here, we propose a new definition for the diversity of ensemble members on a given input. This definition can be applied no matter which loss function is in use. This is similar in spirit to recent attempts by several authors to find a unifying definition for classifier bias and variance ([9, 24, 48]). However, that research is concerned with the analysis of the expected loss of a single classifier over different training samples, while we are interested in the expected performance of ensembles of classifiers.

1.3 Scientific Contributions

The main scientific contributions of this thesis can be summarized as follows:

1. We perform an empirical comparison of different resampling ensemble learning methods. Our experimental results include error curves as well as κ-Error Diagrams, bias-variance decompositions, and cumulative margin distributions. Several years of computing time and over ten million induced base classifiers make this, to our knowledge, the most extensive such study undertaken to-date.

2. The experimental results provide motivation for additional research into resampling ensemble learning methods, as well as into ensemble analysis methodology.

3. We introduce some novel resampling ensemble learning methods, which, in certain situations, may outperform "standard" Bagging both in terms of ensemble accuracy and in terms of running time.

4. We provide novel, unified definitions of what it means for a set of classifiers to be accurate and diverse. These definitions have the desirable property that they are applicable to a large class of ensembles and under any given loss function, thus providing a unifying framework for the analysis of ensemble learning methods.

5. For the case of squared loss, we generalize the standard loss decomposition for ensembles employing sum vote to the case where each classifier returns a probability distribution over $\mathcal{R}$.

6. For the case of 0-1 loss, we derive the exact quantitative relationships between expected ensemble loss, ensemble member accuracy, and diversity. The examination of those relationships allows valuable and sometimes surprising insights into ensemble behavior and performance.

7. We show that, under 0-1 loss, member accuracy and diversity are necessary conditions for an ensemble to outperform the base classifier but not – as generally thought – sufficient ones. This explains some unexpected experimental results which show that ensembles may under certain conditions perform worse than all of its members.

8. We demonstrate how the loss decomposition can be used to reduce the number of ensemble methods which have to be tried out experimentally by a practitioner who is faced with the problem of finding the best ensemble for a given particular prediction problem.

9. All of the above combine to further the general understanding of ensemble behavior and performance.

1.4 Organization

The rest of this thesis is organized as follows:

In Chapter 2 we formally present the problem and the ensemble learning methods under consideration, going from the more general to the more specific. We start by formalizing our notion of predictors, predictor inducers, and how their performance is measured (Section 2.1 and Section 2.2). We then provide a simple taxonomy of ensemble learning methods (Section 2.3) and present the algorithms for the resampling schemes Bagging and Cragging (Section 2.4). We also briefly examine the relationship of our work to research on Bayesian Model Averaging (Section 2.5).

Chapter 3 presents a discussion of the accuracy vs. diversity trade-off for ensembles, and ultimately provides the motivation for the introduction of our unified loss decomposition framework.

In Chapter 4 we compare the performance of Bagging and Cragging on a set of UCI benchmark datasets, and analyze the results using various conventional analysis methods. We conclude that those methods are not well suited for the task of predictive performance analysis.

Chapter 5 presents a theoretical analysis of ensemble performance in terms of member accuracy and diversity. We present our unified loss definitions of mean member loss and diversity, and propose a unified decomposition of the ensemble loss as a

function of mean member loss and ensemble diversity (Section 5.2). We show that the commonly used decomposition under squared loss is but a special case of our unified decomposition, and extend this decomposition to the case where the member classifiers return a probability distribution over $\mathcal{R}$ (Section 5.3). We then instantiate the unified decomposition for 0-1 loss, for two-class problems (Section 5.4.1) as well as for multi-class problems (Section 5.4.2). These instantiations give the exact quantitative relationships between the loss of the ensemble and the diversity and average loss of its individual members. We discuss these relationships and analyze some of their implications (Section 5.4.3). We also derive upper and lower bounds on the expected ensemble loss that hold for a wide class of loss functions (Section 5.5).

In Chapter 6, we turn our attention back to the narrower problem of analyzing the performance of Bagging and Cragging resampling schemes. We show how the loss decomposition helps in predicting ensemble performance, and experimentally verify some issues that arose from the theoretical analysis.

Chapter 7 concludes by summarizing our main results and discussing directions for future work.

2. Problem Description

We consider the standard problem of off-line ("batch"), supervised learning in noisy environments, also commonly referred to as classification or regression. We formally describe the problem setting in the following section.

2.1 Classification and Regression

A learner is given a finite sample $\mathbf{S} = \langle \mathbf{s}_1 = \langle \mathbf{x}_1, y_1 \rangle, \ldots, \mathbf{s}_m = \langle \mathbf{x}_m, y_m \rangle \rangle$ of randomly drawn instances. Each instance $\mathbf{s}_i$ is a pair $\langle \mathbf{x}_i, y_i \rangle$, where $\mathbf{x}_i$ is called the input, and y_i is called the outcome. The set of all possible inputs is called input space and denoted by $\mathbf{X}$, while the set of all possible outcomes is called outcome space and denoted by Y. The instances are generated according to a stationary joint probability distribution $P(\mathbf{X}, Y)$ on $\mathbf{X} \times Y$, the nature of which is unknown to the learner.

After examining the sample $\mathbf{S}$, the learner is required to produce a classifier C. A classifier is a function $C : \mathbf{X} \to \widehat{Y}$ that maps each input $\mathbf{x} \in \mathbf{X}$ to some prediction $\widehat{y} := C(\mathbf{x})$. This classifier produced by the learner will then receive further random instances from the same joint distribution $P(\mathbf{X}, Y)$. For each instance $\langle \mathbf{x}, y \rangle$, the classifier will be shown only the input $\mathbf{x}$, and will be asked to choose a prediction $\widehat{y}(\mathbf{x})$ from some set of possible predictions $\widehat{Y}$.

The process of producing a classifier from a sample is known as inducing or training, while the process of showing further instances to the induced classifier is called evaluation or testing. The pseudo-code procedures for training and evaluating classifiers are shown in Figure 2.1 and Figure 2.2. Note that training sample and test sample are not sets in the strict mathematical sense of the word, but rather multi-sets, since they can contain any given pair $\langle \mathbf{x}, y \rangle$ more than one time.

Throughout this thesis, we will use the shorthand notation $\widehat{y}_C(\mathbf{x})$ to denote the prediction $\widehat{y}$ produced by the classifier C for input $\mathbf{x}$, or just $\widehat{y}(\mathbf{x})$ if the classifier C is clear from the context. The outcomes y are assumed to be causally independent of the predictions $\widehat{y}(\mathbf{x})$. The set of all possible predictions $\widehat{y}$ that a given classifier C can make is called prediction space and denoted by $\widehat{Y}_C$, or just $\widehat{Y}$ if the classifier C is clear from the context.

In general, for the theoretical results in Chapter 3 and Chapter 5 to be applicable,

$\mathbf{X}$, Y, and $\widehat{Y}$ may be arbitrary sets. To simplify the discussion, however, we restrict ourselves to problems for which the inputs $\mathbf{x}_i \in \mathbf{X}$ can be represented as ordered vectors of attribute values, and the outcomes y_i are either discrete-valued ($Y = \{Y_1, Y_2, \ldots, Y_{|Y|}\}$) or real-valued ($Y \subseteq \mathcal{R}$). In principle, each attribute represents a measurement of some instance property common to all instances. Throughout this thesis, the inputs $\mathbf{x}_i \in \mathbf{X}$ are vectors of the form $\langle x_{i1}, x_{i2}, \ldots, x_{in} \rangle$ for some fixed $n \in \mathcal{N}^+$. The components $x_{ij} \in X_j$ are called the attribute values of x_i. Attributes are called discrete-valued iff their values are drawn from a finite set of symbols $X_j = \{X_{j1}, X_{j2}, \ldots, X_{j|X_j|}\}$. They are called real-valued iff $X_j \subseteq \mathcal{R}$.

We consider two common types of classifiers: value classifiers, for which the prediction space $\widehat{Y}$ is the same as the outcome space Y, and distribution classifiers, which output a probability distribution over the outcome space.

Definition 2.1. *A* value classifier *is a function that maps an input* $\mathbf{x} \in \mathbf{X}$ *to an outcome* $y \in Y$*.*

Definition 2.2. *A* distribution classifier *is a function that maps an input* $\mathbf{x} \in \mathbf{X}$ *to a probability distribution* $\widehat{P}(Y|\mathbf{x})$ *over the outcome space* Y*.*

For distribution classifiers, the prediction space $\widehat{Y}$ is the set of all probability distributions $\{\widehat{P}(Y)\}$ on the outcome space Y. For any $y \in Y$, $\widehat{p}(y|\mathbf{x})$ reflects the degree of the classifiers internal belief that the true outcome for input $\mathbf{x}$ is y. We will call the the classifiers' outputs $\widehat{p}(y|\mathbf{x})$ *beliefs*, and the $\widehat{P}(Y|\mathbf{x})$ *belief distributions* – this is to avoid confusion with the conditional probability distributions $P(Y|\mathbf{x})$ on the instance space, as well as to emphasize the fact that the $\widehat{p}(y|\mathbf{x})$ are subjective probability values and not necessarily frequentist ones.

2.2 Loss Functions

The performance of classifiers (and therefore implicitly the learners that produced those classifiers) is assessed using loss functions.

Definition 2.3. *A* loss function *is a function* $l : \widehat{Y} \times Y \to \mathcal{R}$ *that maps each pair* $\langle \widehat{y}, y \rangle \in \widehat{Y} \times Y$ *to a real number* $l(\widehat{y}, y) \in \mathcal{R}$*.*

The loss $l(\widehat{y}, y)$ is interpreted as the cost of making the prediction $\widehat{y}$ when the true value is y. The goal of a learner is to output a classifier with the smallest possible expected loss $L = E_{P(\mathbf{X},Y)}\left[l(\widehat{y}, y)\right]$ according to some given loss function, i.e., a model that minimizes the average loss $l(\widehat{y}(\mathbf{x}), y)$ over instances drawn independently from $\mathbf{X} \times Y$ according to $P(\mathbf{X}, Y)$. Commonly used loss functions for value classifiers include:

- square loss

$$l_2(\widehat{y}, y) := (\widehat{y} - y)^2 \tag{2.1}$$

PROCEDURE **LEARN** ($\mathbf{X}$, Y, P, m, z, l)
  INPUT:
    $\mathbf{X}$ is the input space
    Y is the output space
    $\widehat{Y}$ is the prediction space
    P is a stationary joint probability distribution over $\mathbf{X} \times Y$
    m is the training set size
    z is the test set size
    l is a loss function $l : \widehat{Y} \times Y \to \mathcal{R}$
    J is a classifier inducer $J : (\mathbf{X} \times Y)^{|\mathbf{S}|} \times \{l : \widehat{Y} \times Y \to \mathcal{R}\} \to \{C : \mathbf{X} \to \widehat{Y}\}$
  OUTPUT:
    classifier C is a classifier $C : \mathbf{X} \to \widehat{Y}$
    L is the average loss of C
  BEGIN
    training sample *train* := SAMPLE ($\mathbf{X}$, Y, P, m)
    classifier C := TRAIN (J, *train*, l)
    test sample *test* := SAMPLE ($\mathbf{X}$, Y, P, z)
    average loss L := EVALUATE (C, *test*, l)
  END
  RETURN $\langle C, L \rangle$

Figure 2.1: Pseudo-code procedure LEARN.

- absolute loss
$$l_{\|}(\widehat{y}, y) := |\widehat{y} - y| \tag{2.2}$$
- zero-one loss
$$l_{01}(\widehat{y}, y) := \begin{cases} 0 & \text{iff } \widehat{y} = y \\ 1 & \text{iff } \widehat{y} \neq y \end{cases} \tag{2.3}$$

Square loss or absolute loss are usually applied whenever $\widehat{Y} = Y = \mathcal{R}$, whereas zero-one loss is predominant on problems where Y is some finite set of discrete symbols ($Y := \{y_1, \ldots, y_k\}$), and $\widehat{Y} = Y$.

Although very common, the use of value classifiers has some serious drawbacks. For example, the decision maker has no indication of how much confidence the classifier has in its prediction. It does not even allow the classifier to refrain from making a prediction if it is not sure of the correct label. Also, some types of mistakes may be more costly than other types of mistakes. In database marketing, for example, the cost of mailing a letter to a non-respondent is very small, but the cost of not mailing to a respondent is the entire profit lost ([22]). While such non-uniform error costs can be measured using loss functions, the above definition does not allow them to be taken into account explicitly when making a prediction. Furthermore, there is no possibility of dealing with changing loss functions: should the loss function change – for example, if the average cost of mailing a letter changes with the total number of

PROCEDURE **SAMPLE** ($\mathbf{X}$, Y, P, s)
  INPUT:
    $\mathbf{X}$ is the input space
    Y is the output space
    P is the joint probability distribution of $\mathbf{X} \times Y$
    s is the sample size
  OUTPUT:
    i.i.d.sample $\mathbf{S}$
  BEGIN
    FOR EACH i in $\{1, \ldots, s\}$
      draw instance $\langle \mathbf{x}_i, y_i \rangle$ at random from $\mathbf{X} \times Y$ according to P
    END FOR
    $\mathbf{S} := \langle \langle \mathbf{x}_1, y_1 \rangle, \langle \mathbf{x}_2, y_2 \rangle, \ldots, \langle \mathbf{x}_s, y_s \rangle \rangle$
    RETURN $\mathbf{S}$
  END

PROCEDURE **TRAIN** (J, $\mathbf{S}$, l)
  INPUT:
    J is a classifier inducer $J : \{(\mathbf{X} \times Y)^{|S|}\} \times \{l : \widehat{Y} \times Y \to \mathcal{R}\} \to \{C : \mathbf{X} \to \widehat{Y}\}$
    $\mathbf{S}$ is the training sample
    l is a loss function $l : \widehat{Y} \times Y \to \mathcal{R}$
  OUTPUT: classifier C
  BEGIN
    $C := J(\mathbf{S}, l)$
    RETURN C
  END

PROCEDURE **EVALUATE** (C, $\mathbf{S}$, l)
  INPUT:
    C is a classifier $C : \mathbf{X} \to \widehat{Y}$
    $\mathbf{S}$ is the test sample
    l is a loss function $l : \widehat{Y} \times Y \to \mathcal{R}$
  OUTPUT: average loss $L \in \mathcal{R}$
  BEGIN
    $L := 0$
    FOR EACH $\langle \mathbf{x}, y \rangle \in \mathbf{S}$
      prediction $\widehat{y} := C(\mathbf{x})$
      $L := L + l(\widehat{y}, y)$
    END FOR
    $L := L / |\mathbf{S}|$
    RETURN L
  END

Figure 2.2: Pseudo-code procedures SAMPLE, TRAIN, and TEST.

letters sent, or the expected profit per customer changes over time – the only option for dealing with the change is to learn a new classier from scratch.

Distribution classifiers, rather than producing a prediction $\widehat{y}(\mathbf{x})$ directly, output a belief distribution $\widehat{P}(Y|\mathbf{x})$ over the outcome space instead. In principle, more sophisticated loss functions are possible for distribution classifiers, and the problem of finding an optimal distribution classier for a given domain is reducible to finding an optimal approximation $\widehat{P}(Y|\mathbf{X})$ for the probability distribution $P(Y|\mathbf{X})$ on the outcome space, conditioned on the input space.

In practice, however, the performance of distribution classifiers is usually evaluated as follows:

Given a belief distribution $\widehat{P}(Y|\mathbf{x})$ and assuming the expected error costs $l(\widehat{y}, y)$ are known, the Bayes optimal prediction to make is then the one that minimizes the conditional risk ([22]):

$$R(\widehat{y}|\mathbf{x}) := \int_{y \in Y} \widehat{p}(y|\mathbf{x}) l(\widehat{y}, y)\, dy \tag{2.4}$$

The conditional risk $R(\widehat{y}|\mathbf{x})$ is the expected loss of making the prediction $\widehat{y}$ for input $\mathbf{x}$. The Bayes optimal prediction is guaranteed to achieve the lowest possible overall expected loss over all possible instances $\langle \mathbf{x}, y \rangle$, weighted by their probabilities of occurrence $p(\langle \mathbf{x}, y \rangle)$.

Thus, finding a prediction $\widehat{y}$ when presented with a belief distribution $\widehat{P}(Y|\mathbf{x})$ is straightforward, and the loss functions applicable to value classifiers can also be used for distribution classifiers.

2.3 Ensembles

An ensemble of classifiers is a set of classifiers whose individual predictions are combined in some way to arrive at predictions for previously unseen instances ([18]). That means that all ensembles are also classifiers according to either Definition 2.1 or Definition 2.2.

Many techniques for constructing ensembles of classifiers have been developed. As shown in Figure 2.3, ensemble techniques can be divided into categories, according to how the predictions of the individual member classifiers are combined into the final ensemble prediction, and according to how the member classifiers are generated.

At the prediction level, we distinguish between single-stage combination methods and multi-stage combination methods. In single-stage methods, all the individual member predictions can (in principle) be computed in parallel from the input x, and are then combined into the final ensemble prediction in one single step.

Common examples of single-stage combination methods are weighted or unweighted majority voting (e.g. [3, 63, 76]), confidence averaging (e.g. [27, 64, 74]), rank aggregation ([15, 28, 62]), and gating networks (e.g. [44, 45, 47, 72]). Bagging ([8])

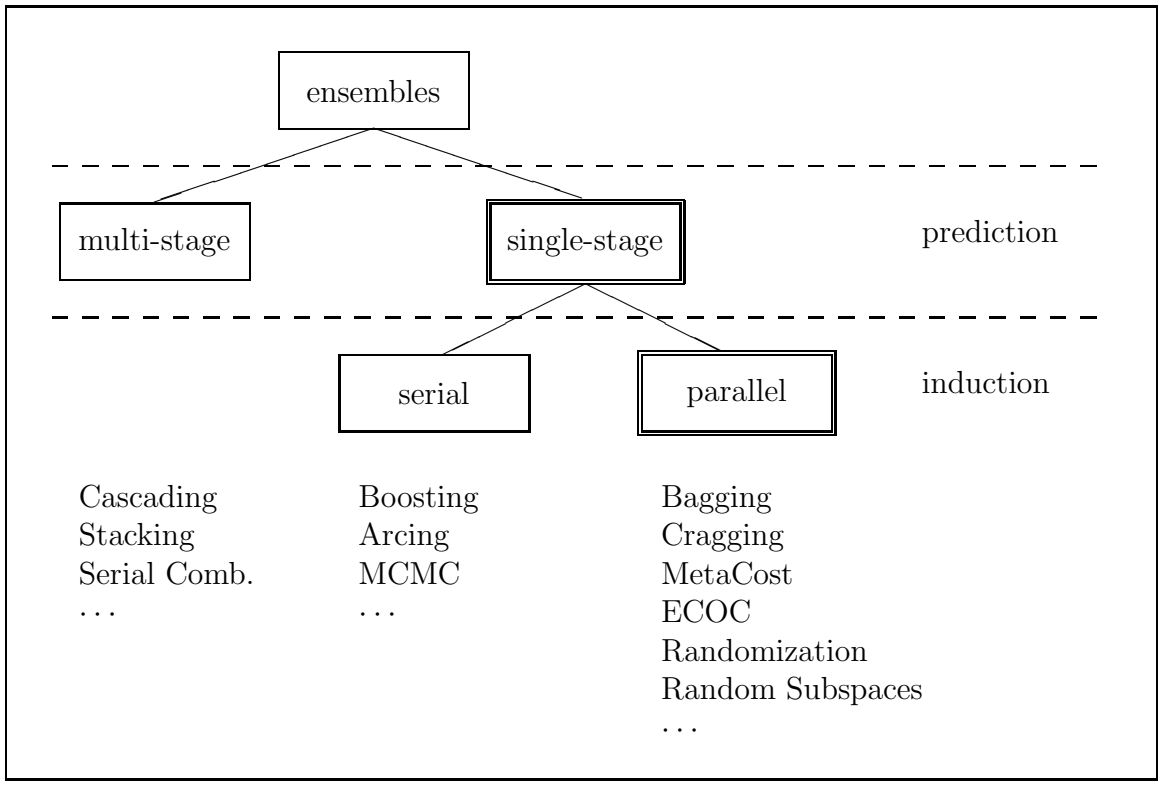

Figure 2.3: A taxonomy of ensemble learning methods.

and Boosting ([29, 72]) are examples of well-known ensemble learning methods using single-stage combination methods.

A single-stage voting ensemble can be fully characterized by specifying a set of member classifiers and a method of combining the (possibly weighted) individual member predictions into a final ensemble prediction.

Definition 2.4. *A (single stage) voting ensemble C with input space $\mathbf{X}$ and prediction space $\widehat{Y}$ is a tuple $C := \langle n, \mathbf{c}, \mathbf{w}, V \rangle$, where*

- *$n \in \mathcal{N}^+$ is the number of component classifiers,*
- *$\mathbf{c} := \langle c_1, c_2, \ldots, c_n \rangle$ is an ordered tuple of classifiers $c_i : \mathbf{X} \to \widehat{Y}_i$,*
- *$\mathbf{w} := \langle w_1, w_2, \ldots, w_n \rangle$ is a real-valued weight vector such that $\sum_{c=1}^{n} w_c = 1$, and*
- *V is a voting function $V : \widehat{Y}_1 \times \cdots \times \widehat{Y}_n \times \mathcal{R}_1 \times \cdots \times \mathcal{R}_n \to \widehat{Y}$ which maps tuples of member classifiers' predictions $\widehat{y}_c$ and classifiers' weights w_c into ensemble predictions $\widehat{y}$.*

Definition 2.5. *The ensemble prediction $\widehat{y}(\mathbf{x})$ of a (single-stage) voting ensemble $\langle n, \mathbf{c}, \mathbf{w}, V \rangle$ for input $\mathbf{x} \in \mathbf{X}$ is defined as*

$$\widehat{y}(\mathbf{x}) := V(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) \tag{2.5}$$

where $\widehat{\mathbf{y}}(\mathbf{x}) = \langle \widehat{y}_1(\mathbf{x}), \ldots, \widehat{y}_n(\mathbf{x}) \rangle$ is the vector of the individual member classifiers' predictions given input $\mathbf{x}$.

Given an input $\mathbf{x}$, first all the predictions of the member classifiers are computed, which can be done in parallel. Usually all the member classifiers operate on the same output space, which we will refer to as $\widehat{Y}_c$.

The requirement that the number of component classifiers be finite or even countable is not strictly necessary for the results in this thesis - it is just a concession in order to obtain a more readable notation. The same holds true for the requirement that the voting weights should always sum to 1.

Further on, we will frequently need to refer to the expected value of some variables, where the expectation is taken over the set of weighted individual member classifiers. To this end, we will use the notation $E_C[A_c]$ to refer to the expected value of variable A_c, taken over all member classifiers $c \in C$ according to the classifier weights w_c. For a finite ensemble C with n component classifiers this expectation is the weighted average

$$E_C[A_c] := \sum_{c=1}^{n} w_c A_c \tag{2.6}$$

In any case, the final ensemble prediction for single-stage voting ensembles is obtained simply by combining the (possibly weighted) individual member predictions via the voting function V.

Some examples of commonly used voting functions are:

- Sum Vote
  For $Y = \widehat{Y} = \widehat{Y}_c = \mathcal{R}$ and $l = l_2$:

$$V_{\text{sum}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) := E_C[\widehat{y}_c(\mathbf{x})] = \sum_{c=1}^{n} w_c \widehat{y}_c(\mathbf{x}) \tag{2.7}$$

  For $Y = \widehat{Y} = \mathcal{R}$, $\widehat{Y}_c = \{P(\mathcal{R})\}$, and $l = l_2$:

$$V_{\text{sum}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) := \sum_{c=1}^{n} w_c \int_{y' \in \mathcal{R}} \widehat{p}_c(y'|\mathbf{x}) y' \, dy' \tag{2.8}$$

- Majority Vote
  For $Y = \widehat{Y} = \widehat{Y}_c = \{Y_1, \ldots, Y_k\}$ and $l = l_{01}$:

$$V_{\text{maj}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) := \arg\max_{y' \in Y} \sum_{c=1}^{n} w_c I(\widehat{y}_c(\mathbf{x}) = y') \tag{2.9}$$

  For $Y = \widehat{Y} = \{Y_1, \ldots, Y_k\}$, $\widehat{Y}_c = \{P(Y)\}$, $l = l_{01}$:

$$V_{\text{maj}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) := \arg\max_{y' \in Y} \sum_{c=1}^{n} w_c I(\arg\max_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x}) = y') \tag{2.10}$$

- Probabilistic Vote
  For $Y = \widehat{Y} = \{Y_1, \ldots, Y_k\}$, $\widehat{Y}_c = \{P(Y)\}$ and $l = l_{01}$:

$$V_{\text{prob}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) := \arg\max_{y' \in Y} \sum_{c=1}^{n} w_c \widehat{p}_c(y'|\mathbf{x}) \tag{2.11}$$

  For $Y = \{Y_1, \ldots, Y_k\}$, $\widehat{Y} = \widehat{Y}_c = \{P(Y)\}$ and $l = l_{01}$:

$$V_{\text{prob}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) := \left\langle \widehat{P}(Y|\mathbf{x}) \right\rangle \text{ with } \widehat{p}(y|\mathbf{x}) := \sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x}) \tag{2.12}$$

The weight vector $\mathbf{w} = \langle w_1, \ldots, w_n \rangle$ is determined as part of the induction process. The function $I()$ denotes the the indicator function

$$I(a = b) := \begin{cases} 1 & \text{iff } a = b \\ 0 & \text{iff } a \neq b \end{cases} \tag{2.13}$$

Many other voting functions are conceivable and have been investigated, see e.g. [43], [63] or [73] for surveys.

In multi-stage methods, some of the member classifiers take as part of their input the predictions of other member classifiers. Therefore, the prediction process is not completely parallelizable. Popular multi-stage combination methods are for example Cascading ([46]), Stacking ([80]), or Serial Combination ([55]).

The nature of multi-stage combination methods usually calls for ensemble induction methods that also proceed in multiple stages: classifiers which take as part of their input the predictions of classifiers in lower stages must usually be trained after the construction of the earlier stage classifiers is already complete. For, single-stage combination methods, however, we can distinguish ensemble techniques further by whether or not the induction process can be parallelized. Parallelizable ensemble induction methods include for example Bagging ([8]), MetaCost ([22]), ECOC ([19, 50]), Randomization ([16]), and Random Subspaces ([42]). Some popular ensemble induction methods are not parallelizable: they proceed sequentially in iterations. In each iteration, some member classifiers are constructed, and the output (classifiers, weights, etc.) of previous iterations is needed in later iterations. Boosting ([29, 72]), Arcing ([9]), and MCMC ([2, 32]) are examples of popular ensemble methods where the voting process is parallelizable but the induction process is not.

Our theoretical results in Chapter 5 are independent of how the member classifiers were generated – they are applicable to all single-stage prediction methods as defined in Definition 2.4.

The experimental part of this thesis (Chapter 4 and Chapter 6) is concerned with ensemble methods that generate member classifiers by running a particular learning algorithm on different samples, which are generated from the original training sample by some fixed sampling method. All the ensemble methods in our experiments fall into the category of single-stage prediction, parallel induction.

```
PROCEDURE BAGGING (J, S, l, r, s)
  INPUT:
    J  is a classifier inducer
    S  is the training sample
    l  is a loss function l : Ŷ × Y → R
    r  is the desired number of iterations
    s  is the desired relative sample size
  OUTPUT:
    ensemble C with r member classifiers
  BEGIN
    FOR EACH i in {1, ..., r}
      S_i := Bootstrap sample containing s * |S| instances from S
             (i.i.d.sample with replacement)
      C_i := J(S_i, l)
    END FOR
    C := ⟨C_1, C_2, ..., C_r⟩
    RETURN C
  END
```

Figure 2.4: The Bagging algorithm.

2.4 Bagging and Cragging

There has recently been much interest in ensemble methods that generate multiple member classifiers by running some fixed base learning algorithm multiple times on different samples generated from the original training sample.

One of the most popular such ensemble methods is Bagging ([8]). Figure 2.4 shows the outline of the Bagging algorithm. Given a training sample $\mathbf{S}$ and a classifier inducer J, Bagging constructs an ensemble of r member classifiers by generating r bootstrap samples (hence the name *B*ootstrap *Agg*regating) and running the classifier inducer on each of the r samples. Each bootstrap sample is generated by sampling $s * |\mathbf{S}|$ instances from the original training sample with replacement, for some constant $s \in \mathcal{R}^+$. Note that in the original Bagging algorithm, each bootstrap sample $\mathbf{S}_i$ contains the same number of instances as the original training sample $\mathbf{S}$. In this thesis, the number of instances in each bootstrap sample is $s * |\mathbf{S}|$, where s is a parameter to the Bagging algorithm. The standard behavior of $|\mathbf{S}|$ instances in each bootstrap sample is obtained by setting $s := 1$.

The final ensemble classifier C is assembled from the r member classifiers by one of the voting functions given in Section 2.3, with the voting weights all being equal ($w_c := 1/r$ for all $c \in C$).

Significant performance improvements through Bagging have been demonstrated empirically by many authors (e.g. [3, 16, 18, 68, 59]). However, the theoretical findings related to Bagging (and other single-stage prediction ensembles, for that

PROCEDURE **CRAGGING** (J, $\mathbf{S}$, l, r, s)
INPUT:
J is a classifier inducer
$\mathbf{S}$ is the training sample
l is a loss function $l : \widehat{Y} \times Y \rightarrow \mathcal{R}$
r is the desired number of iterations
s is the desired number of cross-validation folds
OUTPUT:
ensemble $\mathbf{C}$ with $r * s$ member classifiers
BEGIN
FOR EACH i in $\{1, \ldots, r\}$
randomly divide $\mathbf{S}$ into s equal-sized partitions $\mathbf{S}'_{i,j}$
FOR EACH j in $\{1, \ldots, s\}$
$\mathbf{S}_{i,j} := \mathbf{S}'_{i,1} \cup \cdots \cup \mathbf{S}'_{i,j-1} \cup \mathbf{S}'_{i,j+1} \cup \cdots \cup \mathbf{S}'_{i,s}$
$C_{(i-1)*s+j} := J(\mathbf{S}_{i,j}, l)$
END FOR
END FOR
$\mathbf{C} := \{C_1, C_2, \ldots, C_{r*s}\}$
RETURN $\mathbf{C}$
END

Figure 2.5: The Cragging algorithm.

matter) are still not complete: There have been different and sometimes contradictory explanations as to when it works and why. In Chapter 4, we will investigate the more popular of those explanations in detail. Specifically, nobody has yet been able to show conclusively if, and under which precise conditions, for a given problem domain $\langle \mathbf{X}, Y, P \rangle$ and restrictions $\langle l, |\mathbf{S}|, n \rangle$, Bagging is the optimal way of combining n classifiers.

The relative sample size s of the bootstrap samples is one example – nobody has proven yet that always setting $s := 1$ is the optimal thing to do in terms of expected ensemble loss. This motivates the experimental studies carried out in Chapter 3 and Chapter 4.

Another performance-influencing factor that we considered worth a closer look is the actual sampling scheme. We could replace the sampling with replacement by a sampling scheme without replacement. This leads to an ensemble method called Cragging (as in *Cr*oss-validation *Agg*regating), which is shown in Figure 2.5. Cragging divides the original training sample $\mathbf{S}$ into s approximately equal-sized, mutually exclusive partitions or folds $\{\mathbf{S}'_1, \ldots, \mathbf{S}'_s\}$. The training sample for each member classifier is then constructed by copying the original sample $\mathbf{S}$, but leaving out one of the s partitions. This whole partitioning process can be repeated r times, leading to an ensemble consisting of a total of $r * s$ member classifiers.

As we will usually compare ensembles generated using the same base classifier in-

ducer J (namely, C4.5) and under the same loss function l, we will simply use $Bagging(s; n)$ or $\mathcal{B}(s; n)$ to denote Bagging with n iterations and relative sample size s (Bagging (J, $\mathbf{S}$, l, r, s) with $r := n$), and we will use $Cragging(f; n)$ or $C(f; n)$ to denote Cragging with n iterations and f partitions (Cragging (J, $\mathbf{S}$, l, r, s) with $r := n$ and $s := f$) operating on some sample $\mathbf{S}$ where J, l, and $\mathbf{S}$ are clear from the context. Note that $\mathcal{B}(s; n)$ generates an ensemble with n members, while $C(f; n)$ generate ensembles consisting of $n * f$ members.

Ensembles constructed by Cragging are also sometimes called *cross-validated committees* ([18]). There have been some studies of Cragging in the literature ([52, 61]), but those are inconclusive to say the least. Kohavi reports in [49], within the context of estimating classifier accuracy that, under 0-1 loss, cross-validation results in accuracy estimates with both lower bias and higher variance than accuracy estimates obtained using bootstrap samples. This leads us to suspect that Cragging may actually produce ensembles whose member are more accurate and diverse than those produced by Bagging – a hypothesis which we will investigate in Chapter 3.

2.5 Relationship to Bayesian Model Averaging

Bayesian Model Averaging (BMA) is a technique designed to help account for the uncertainty inherent in the model induction process. By averaging over many different classifiers, BMA can incorporate uncertainties about the underlying problem domain into the induction and prediction processes. As such, BMA can be interpreted as a single-stage voting ensemble technique, with the classifiers' weights w_c corresponding to the posterior model probability given the observed training sample S. Mathematically, the BMA rule can be written as

$$\widehat{p}(\langle \mathbf{x}, y \rangle | S) := \sum_{c \in C} p(\langle \mathbf{x}, y \rangle, S | c) \widehat{p}(c), \tag{2.14}$$

where $\widehat{p}(c)$ is the prior belief (prior to having observed any training data) that c is the true model for the problem domain, and $p(\langle \mathbf{x}, y \rangle, S|c)$ is the probability of observing the new instance $\langle \mathbf{x}, y \rangle$ and the training sample S, given that c actually is the true model ([57]).

It is well known that, when using a complete set of all candidate models, such that exactly one candidate model actually is the true model, the BMA rule combined with conditional risk minimization as in Equation 2.4 will lead to optimal predictors. That is, no other prediction method operating with the same prior knowledge (as manifested by the selection of candidate models c and prior model probabilities $\widehat{p}(c)$) can achieve lower expected loss on new instances.

Explicitly constructing such an ensemble of all candidate models (*"Optimal BMA"*) is usually computationally infeasible, due to the sheer number of candidate models. Optimal BMA is therefore commonly approximated by selecting the few candidate models with the highest posterior probability and combining them into an ensemble

(*"Common BMA"*). However, no equivalent optimality guarantees can be made for Common BMA.

In fact, other single-stage voting ensemble methods can be interpreted as alternative approximations of Optimal BMA, and have been shown to outperform Common BMA under certain circumstances ([20, 23]).

Both BMA and our accuracy vs. diversity trade-off analysis provide two alternative explanations for the performance of ensemble techniques. As is usually the case with multiple analyses of any phenomena, these two analyses highlight different aspects concerning the performance of ensemble techniques.

On one side, under the assumption that exactly one of the ensemble members constitutes the true model for the given problem domain, Equation 2.14 can be used to compute an optimal set of weights w_c for the ensemble members, and the optimality guarantee implies the existence of optimal trade-off points for the accuracy vs. diversity trade-off, which can even be quantitatively specified.

The above assumption is violated for all of the ensemble methods and prediction problems studied in the experimental part of this thesis. However, it may certainly be possible to use optimal BMA to arrive at meaningful conclusions about the location of optimal trade-off points for simple base learners on simple artificial problem domains.

One the other side, our work shows that Common BMA may not be the best approximation to Optimal BMA. This is especially true in situations where Common BMA selects models with high posterior probabilities which are very similar to each other. It is entirely conceivable that choosing models with lower posterior probabilities but higher diversity will result in better performing ensembles than those constructed using Common BMA.

Thus, our work provides an alternative explanation for the outperformance of Common BMA by simple ensemble methods (such as e.g. Bagging) to the explanations given in ([20, 57]).

3. The Accuracy-Diversity Trade-Off

Just as there can be no single best classifier in general ([21, 78, 79]), there can be no single best ensemble method. The correct question is therefore usually not which method is better, but rather under which conditions a certain method outperforms another one or vice versa.

The prevailing wisdom is that a good classifier ensemble is one whose members are both accurate and diverse (e.g. [3, 17, 60, 65, 70]). To see why accuracy and diversity are good, consider an ensemble $C := \langle n, \mathbf{c}, \mathbf{w}, V \rangle$, consisting of three equally-weighted classifiers ($n := 3$, $\mathbf{c} := \langle c_1, c_2, c_3 \rangle$, and $\mathbf{w} := \langle 1/3, 1/3, 1/3 \rangle$), and any test instance $\langle \mathbf{x}, y \rangle$. On being presented with input $\mathbf{x}$, the three models make the predictions $\widehat{\mathbf{y}}(\mathbf{x}) = \langle \widehat{y}_1(\mathbf{x}), \widehat{y}_2(\mathbf{x}), \widehat{y}_3(\mathbf{x}) \rangle$.

If the three models are identical (not diverse), then whenever $\widehat{y}_1(\mathbf{x})$ is incorrect, $\widehat{y}_2(\mathbf{x})$ and $\widehat{y}_3(\mathbf{x})$ will also be incorrect, as will be the prediction of the ensemble as a whole. If, however, the three models tend to make different predictions given input $\mathbf{x}$, then even though $\widehat{y}_1(\mathbf{x})$ is incorrect, $\widehat{y}_2(\mathbf{x})$ and $\widehat{y}_3(\mathbf{x})$ may still be correct, so that a majority vote of the three models c_1, c_2, and c_3 will correctly predict y. More precisely, if we had an ensemble consisting of $n = 2k + 1$ models whose probabilities of making a mistake would all be equal to some probability $p < 0.5$ and whose mistakes would be independent of each other, the probability that the majority vote of those classifiers would be wrong would follow the area under the binomial distribution $B_{2k+1}(k; p)$, which rapidly approaches 0 with growing k.

Unfortunately, in real life the matter is not quite that simple. The mistakes of the member models will never be completely independent of each other, unless the predictions themselves are completely random (in which case the probability of a mistake will not be less than 0.5 anymore). In fact, accuracy and diversity are quite contradictory goals: In order for the member models to be diverse, they have to make mistakes. And if the member models are all accurate, their predictions will agree with each other.

Clearly, there is a trade-off one has to make when learning an ensemble of models: Compared to learning a single model, we will have to sacrifice some of its accuracy and instead generate a set of models which are somewhat less accurate on average but are diverse.

Within the neural network community, this trade-off has been formalized and quantified for the case of square loss ([18, 39, 52]): Given an ensemble C consisting of n member classifiers $c_1, c_2, \ldots, c_n$, the mean member loss is the opposite of mean member accuracy, and is defined as

$$\overline{L}_2(\mathbf{x}, y) := E_C\left[(\widehat{y}_c(\mathbf{x}) - y)^2\right] = \frac{1}{n}\sum_{c=1}^{n}(\widehat{y}_c(\mathbf{x}) - y)^2. \tag{3.1}$$

The diversity among the ensemble members is measured by

$$\overline{D}_2(\mathbf{x}) := E_C\left[(\widehat{y}_c(\mathbf{x}) - \widehat{y}(\mathbf{x}))^2\right] = \frac{1}{n}\sum_{c=1}^{n}(\widehat{y}_c(\mathbf{x}) - \widehat{y}(\mathbf{x}))^2. \tag{3.2}$$

The ensemble loss, which is defined as

$$L_2(\mathbf{x}, y) := (\widehat{y}(\mathbf{x}) - y)^2, \tag{3.3}$$

can then be re-written as

$$L_2(\mathbf{x}, y) = \overline{L}_2(\mathbf{x}, y) - \overline{D}_2(\mathbf{x}), \tag{3.4}$$

which very nicely expresses the trade-off between member accuracy and diversity in a concise numerical form ([52]). From Equation 3.4 also follows that, under square loss, accuracy and diversity are necessary and sufficient conditions for an ensemble of classifiers to outperform any of its member classifiers.

It is commonly assumed that similar trade-offs apply to ensemble learning in general, and that the necessity and sufficiency of member accuracy and diversity extend to other loss functions as well ([3, 17, 65, 70]): For an ensemble of classifiers to be useful, its member classifiers must disagree on some instances. Clearly, if all the member predictions agree with each other on all instances, then there is no more information to be gained from the ensemble then there is from just a single classifier.

Figure C.1 in Appendix C (pages 86–87) shows the error curves for *Cragging*$(n; 1)$ for each of the datasets tested, together with the expected loss for a single classifier shown as horizontal lines (see Section 4.1 for a detailed description of the experimental methodology). Figure 3.1 on page 20 summarizes the results from Appendix C: it shows the error curve obtained by averaging all the error curves shown in Figure C.1.

Remember that *Cragging*$(n; 1)$ generates n classifiers by dividing the original training sample $\mathbf{S}$ into n disjoint partitions $\{\mathbf{S}'_1, \ldots, \mathbf{S}'_n\}$ and calling the base learner n times, each time with a new training sample $\mathbf{S}_i := \mathbf{S} - \mathbf{S}'_i$ that is generated from the original training sample by leaving out the instances in partition $\mathbf{S}'_i$. The 0-1 loss of the resulting ensemble is shown on the y-axis, for the ensembles generated with $n := \{2, \ldots, 30\}$.

Each of the member classifiers is trained on $|\mathbf{S}|(1 - 1/n)$ instances, and any two member classifiers of the same ensemble share all but $|\mathbf{S}|/n$ training instances. We

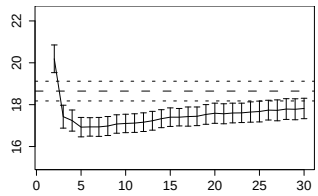

Figure 3.1: Averaged error curves for $Cragging(n; 1)$

can expect therefore, generally, that small settings of n will result in ensembles of low member accuracy (as the member classifiers get to see only a small fraction of the original training sample) and high diversity (as the member classifiers are trained with a higher fraction of disjoint instances), whereas higher settings of n will result in ensembles of higher member accuracy and lower diversity. Looking at Figure C.1 and Figure 3.1, one can see that, as we increase n, the ensemble loss tends to first decrease and then increase again, with some domain-dependant optimal setting for n resulting in a minimal loss in between. This supports the notion of some accuracy-diversity tradeoff taking place.

Similar observations can be made when comparing the performance of ensembles with the same number of member classifiers but generated using different sampling schemes, as shown in Figure 3.2 on pages 21–23. All ensembles shown consist of 30 member classifiers, with the loss of a single base classifier shown for comparison on the left. Recall that $Bagging(s; n)$ works by generating n member classifiers using n bootstrap samples of the original training sample $\mathbf{S}$, each bootstrap sample containing $s * |S|$ instances.

As we go from $Bagging(0.5; 30)$ over $Bagging(1; 30)$ on to $Bagging(2; 30)$, we expect the member classifiers to become more and more accurate (as the bootstrap samples they are trained on contain a higher and higher proportion of the original training instances) but less and less diverse (as the bootstrap samples for the different member classifiers share a higher and higher proportion of instances).

Similar effects occur when we move from $Cragging(2; 15)$ over $Cragging(3; 10)$ to $Cragging(30; 1)$: member classifiers will become more and more accurate but less and less diverse. Recall that $Cragging(f; n)$ generates n member classifiers in n/f iterations, during each iteration partitioning the original training sample $\mathbf{S}$ into f partitions and and then generating f member classifiers by leaving out one of the partitions at a time.

So, what Figure 3.2 shows is that changes to member accuracy and diversity using different methods but qualitatively in the same direction will result in similar qualitative changes to the ensemble loss, independent of how these changes in member accuracy and diversity were caused. Again, this supports the hypothesis of member accuracy and diversity being the main influencing factors with respect to expected ensemble loss.

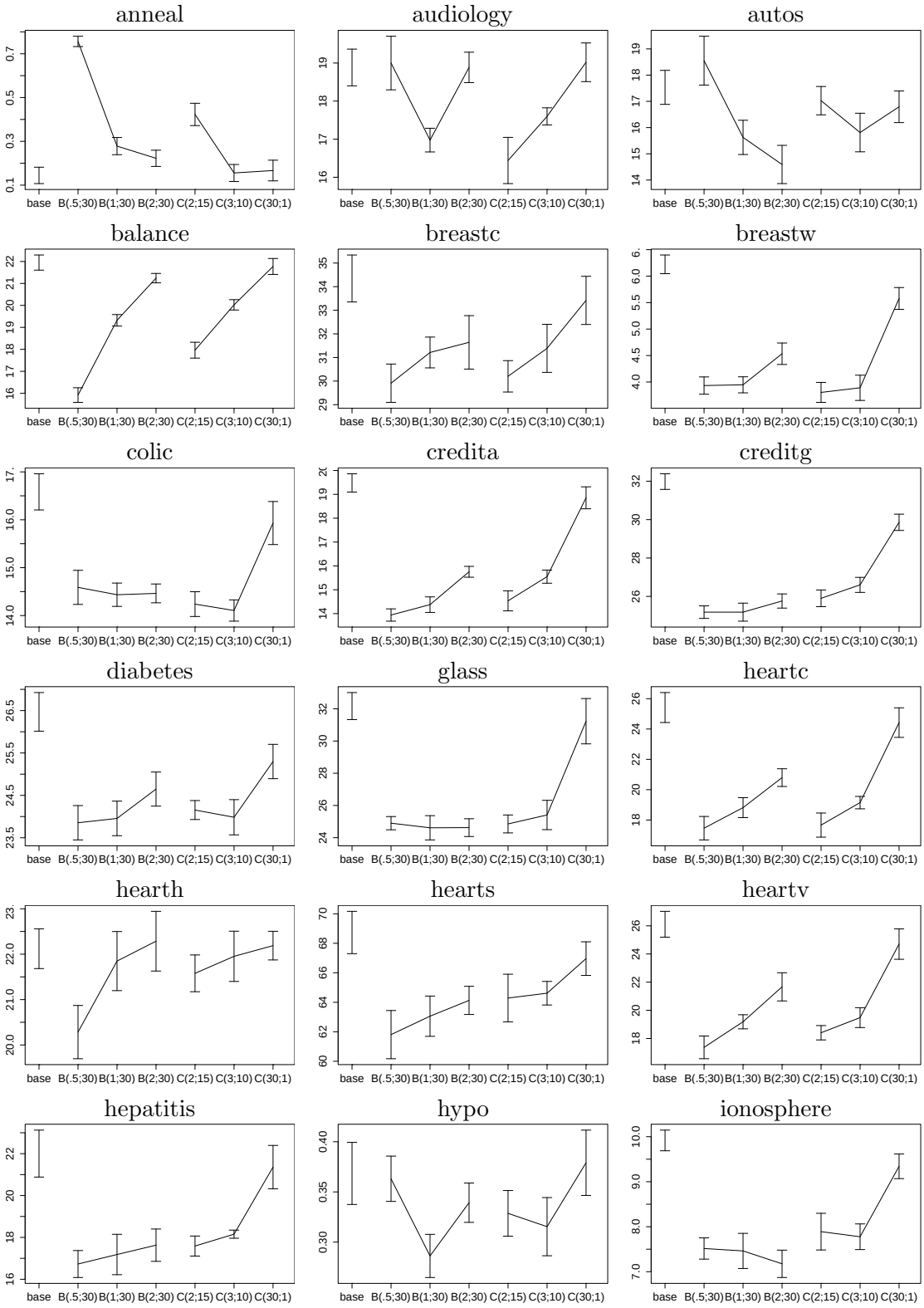

Figure 3.2: Loss comparison for ensembles with 30 classifiers.

(continued on next page)

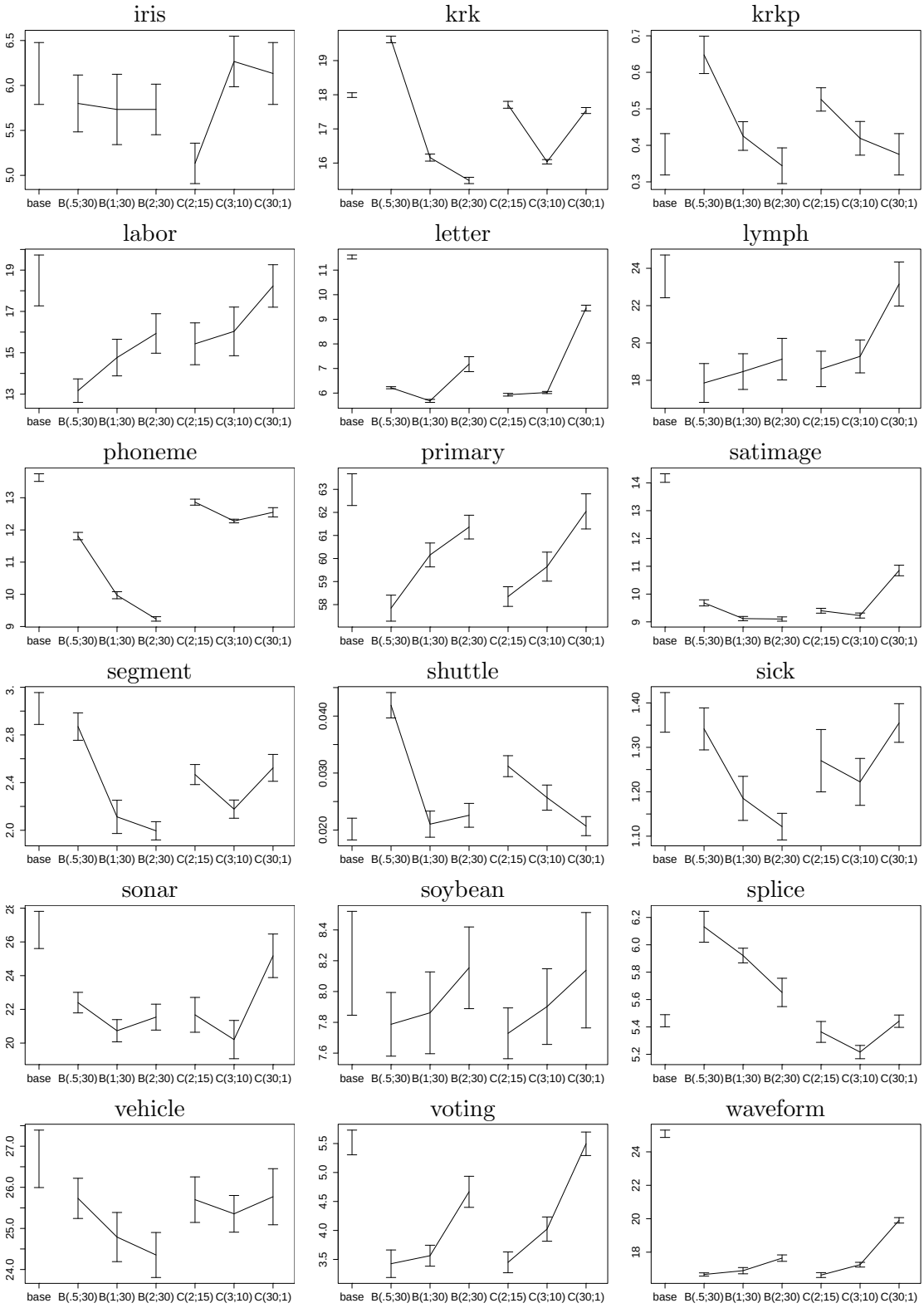

Figure 3.2: Loss comparison for ensembles with 30 classifiers.

Thus, in summary we find that the notion of a trade-off between member accuracy and diversity in order to achieve minimal ensemble loss is not only nice and intuitive, but also seems to hold up in practice. However, so far this is only guessing, and a formal, quantitative analysis of the accuracy vs. diversity trade-off has been missing so far for loss functions other than square loss. In Chapter 5 we will show that, for the case of 0-1 loss, accuracy and diversity are indeed necessary conditions but by themselves not sufficient ones.

4. Current Ensemble Analysis Methods

In Chapter 3 we presented a preliminary intuitive explanation for why ensembles work. We have also seen (Figure 3.2) that "standard" Bagging can indeed frequently be improved upon by varying the instance sampling scheme.

In this chapter we are going to examine some additional/alternative ensemble analysis methods together with their respective explanations of why and how ensembles work.

4.1 Experimental Methodology

To evaluate the performance of the various sampling schemes, we selected 36 datasets from the UCI dataset repository ([5]). The datasets where chosen without regard to the result of the evaluation. Rather, we selected them based on their previous usage in similar studies – for easier comparability of results – and on their easy availability. The chosen datasets actually form a superset of the UCI datasets employed for the comparison studies in [3, 16, 49, 59], and [66]. Table 4.1 on page 25 gives the characteristics of the datasets. No other domains have been tested as part of this study.

All the experimental results were obtained using 10 runs of 10-fold cross-validation, resulting in a total of 100 runs for each dataset, ensemble method, and number of component classifiers. The same random seed was used for each experiment, ensemble method, and number of member classifiers, resulting in the same 100 training data samples to be passed to each ensemble method in each experiment, thus ensuring direct comparability of results with respect to the different ensemble methods and across experiments.

As base classifier inducer, we used J.48 in the Weka machine learning software version 3.1.5 ([77]), which is the Weka implementation of C4.5 Release 8 ([66]) to generate unpruned decision trees ("`weka.classifiers.j48.J48 -- -U -B`"). Multiple studies ([3, 8, 16]) have come to the conclusion that ensemble methods based on data resampling work best with unpruned decision trees. This is attributed generally to the fact that unpruned decision trees are more "unstable", i.e., they exhibit

Dataset	Description	Instances	Features		Classes
			cont.	discr.	
anneal	Annealing	898	6	32	5
audiology	Audiology	226	-	69	24
autos	Auto Imports	205	15	10	6
balance	Balance Scale	625	4	-	3
breastc	Breast-Cancer	286	0	9	2
breastw	Breast-Cancer Wisconsin	699	9	-	2
colic	Horse-Colic	368	7	15	2
credita	Australian Credit	690	6	9	2
creditg	German Credit	1,000	7	13	2
diabetes	Pima Diabetes	768	8	-	2
glass	Glass Identification	214	9	-	6
heartc	Cleveland Heart-Disease	303	6	7	2
hearth	Hungarian Heart-Disease	294	6	7	2
hearts	Switzerland Heart-Disease	123	6	7	5
heartv	Virginia Heart-Disease	270	6	7	2
hepatitis	Hepatitis	155	6	13	2
hypo	Hypothyroid	3,772	7	22	4
ionosphere	Ionosphere	351	34	-	2
iris	Iris	150	4	-	3
krk	King+Rook vs. King	28,050	-	6	18
krkp	King+Rook vs. King+Pawn	3,196	-	36	2
labor	Labor Contracts	57	8	8	2
letter	Letter Recognition	20,000	16	-	26
lymph	Lymphography	148	3	15	4
phoneme	Phonemes	5,404	5	-	2
primary	Primary Tumor	339	-	17	22
satimage	Satellite Images	6,435	36	-	6
segment	Image Segmentation	2,310	19	-	7
shuttle	Shuttle	58,000	9	-	7
sick	Thyroid Disease	3,772	7	22	2
sonar	Sonar Signals	208	60	-	2
soybean	Soybean	683	-	35	19
splice	Splice-Junctions (DNA)	3,190	-	61	3
vehicle	Vehicle Silhouettes	846	18	-	4
voting	House Votes	435	-	16	2
waveform	Waveform-5000	5,000	40	-	3

Table 4.1: Datasets used in the experiments.

a greater statistical variance than pruned ones ([8, 51], see also Section 4.5).

Where error bars are shown or standard deviations are reported, these reflect one standard deviation taken over the ten runs, after averaging the ten cross-validation results for each run separately – as recommended e.g. in [7, 49]. Ensembles consist of up to 30 member classifiers, as several studies (e.g. [59, 66]) have shown that most of the performance gains occur with combining the first few classifiers (see also Section 4.2).

Several of the experiments described in the remainder of this chapter produce estimates of the expected ensemble performance by repeatedly measuring the ensemble loss on some data subsamples. To ensure that all the experimental results reported are indeed correct and consistent with each other, we compare the ensemble loss estimates obtained from different experiments with each other in Appendix J.

4.2 Error Curves

One of the simplest approaches to analyzing ensemble performance is to actually check how the ensemble loss behaves as more and more member classifiers are added to the ensemble. Figure 4.1 on pages 26–31 shows the error curves for Bagging with different relative sample sizes, namely $\mathit{Bagging}(0.5; n)$, $\mathit{Bagging}(1; n)$ ("normal" Bagging), and $\mathit{Bagging}(2; n)$, while Figure 4.2 on pages 32–36 shows the error curves for $\mathit{Cragging}(2; n)$ and $\mathit{Cragging}(3; n)$ versus $\mathit{Bagging}(1; n)$. The horizontal lines show the mean and standard deviation of the 0-1 loss for a single base classifier.

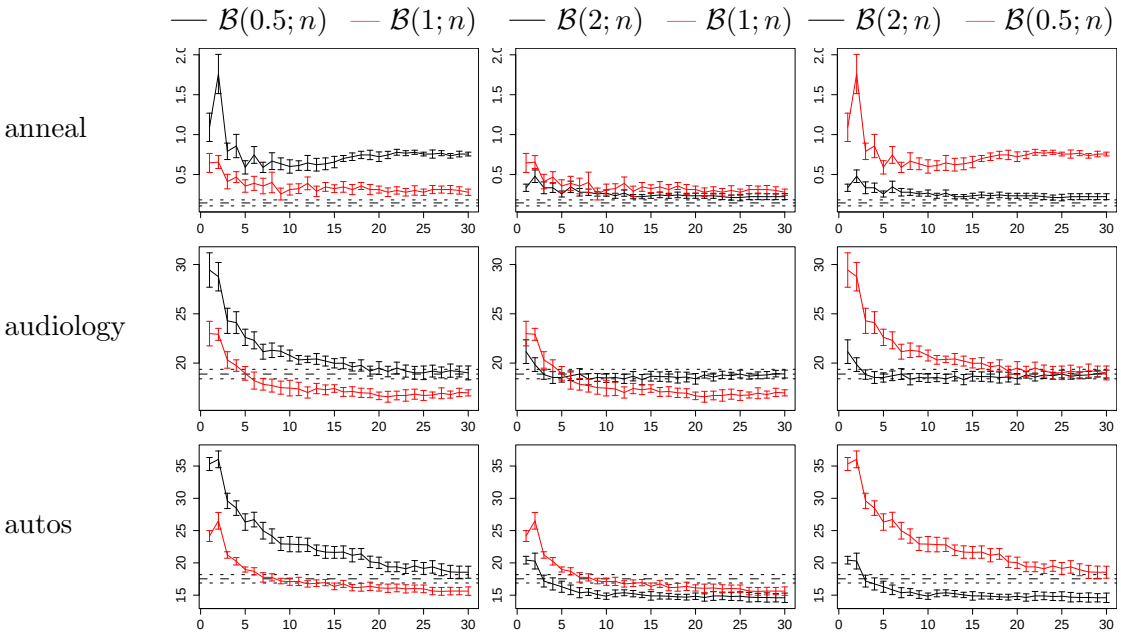

Figure 4.1: Error curves for $\mathcal{B}(0.5; n)$ versus $\mathcal{B}(1; n)$, $\mathcal{B}(2; n)$ versus $\mathcal{B}(1; n)$, and $\mathcal{B}(2; n)$ versus $\mathcal{B}(0.5; n)$.

(continued on next page)

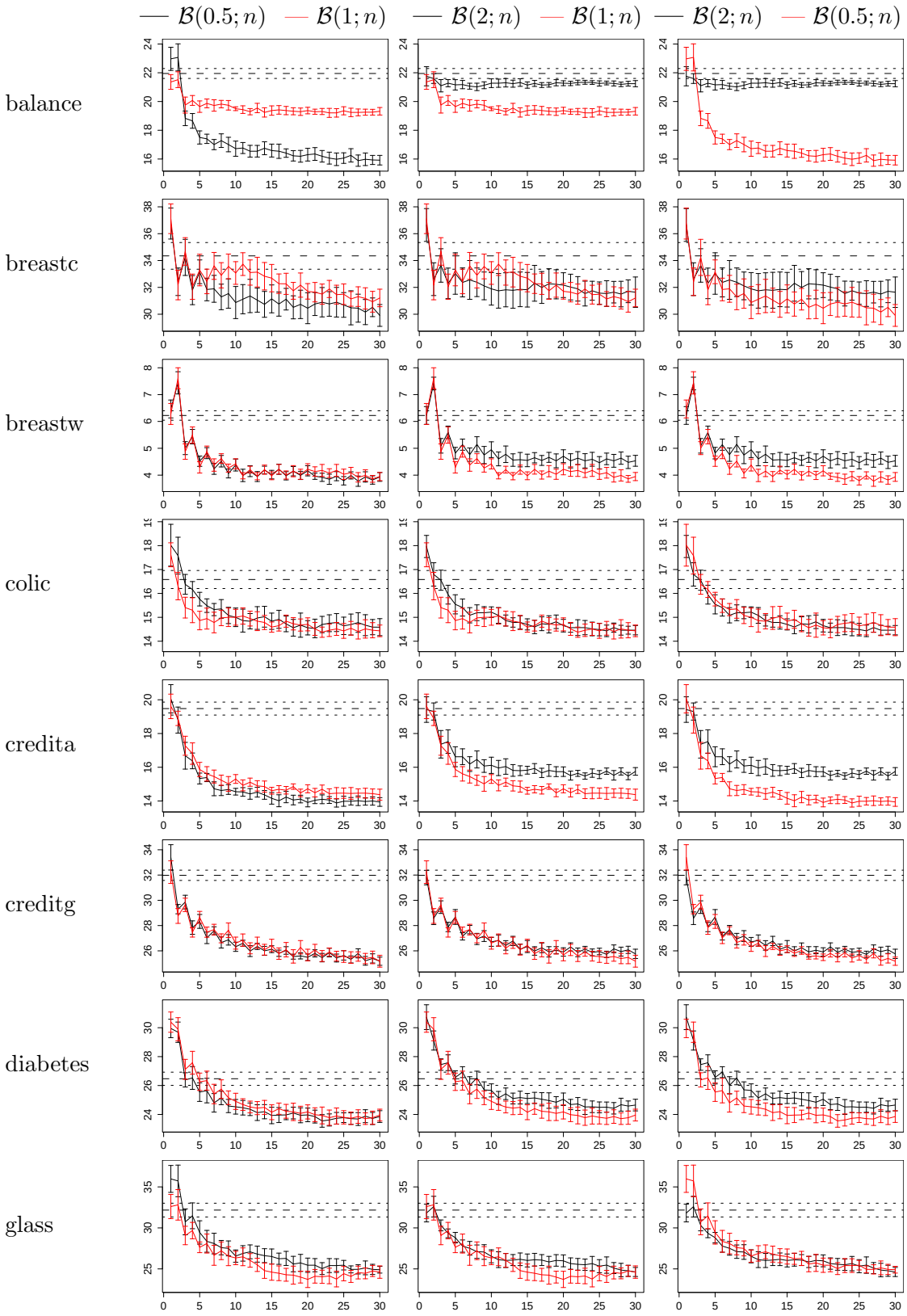

Figure 4.1: Error curves for $\mathcal{B}(0.5;n)$ versus $\mathcal{B}(1;n)$, $\mathcal{B}(2;n)$ versus $\mathcal{B}(1;n)$, and $\mathcal{B}(2;n)$ versus $\mathcal{B}(0.5;n)$.

(continued on next page)

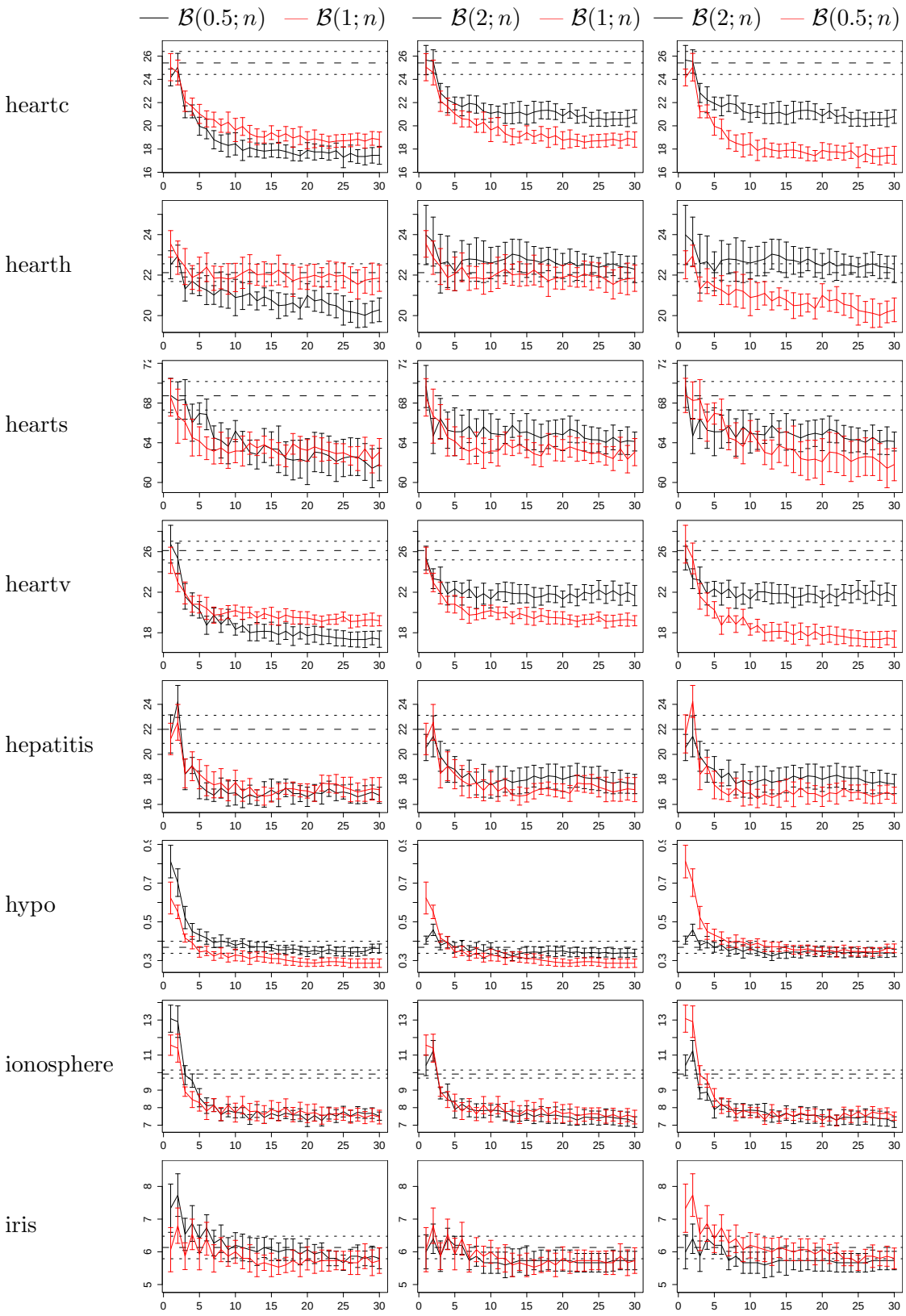

Figure 4.1: Error curves for $\mathcal{B}(0.5;n)$ versus $\mathcal{B}(1;n)$, $\mathcal{B}(2;n)$ versus $\mathcal{B}(1;n)$, and $\mathcal{B}(2;n)$ versus $\mathcal{B}(0.5;n)$.

(continued on next page)

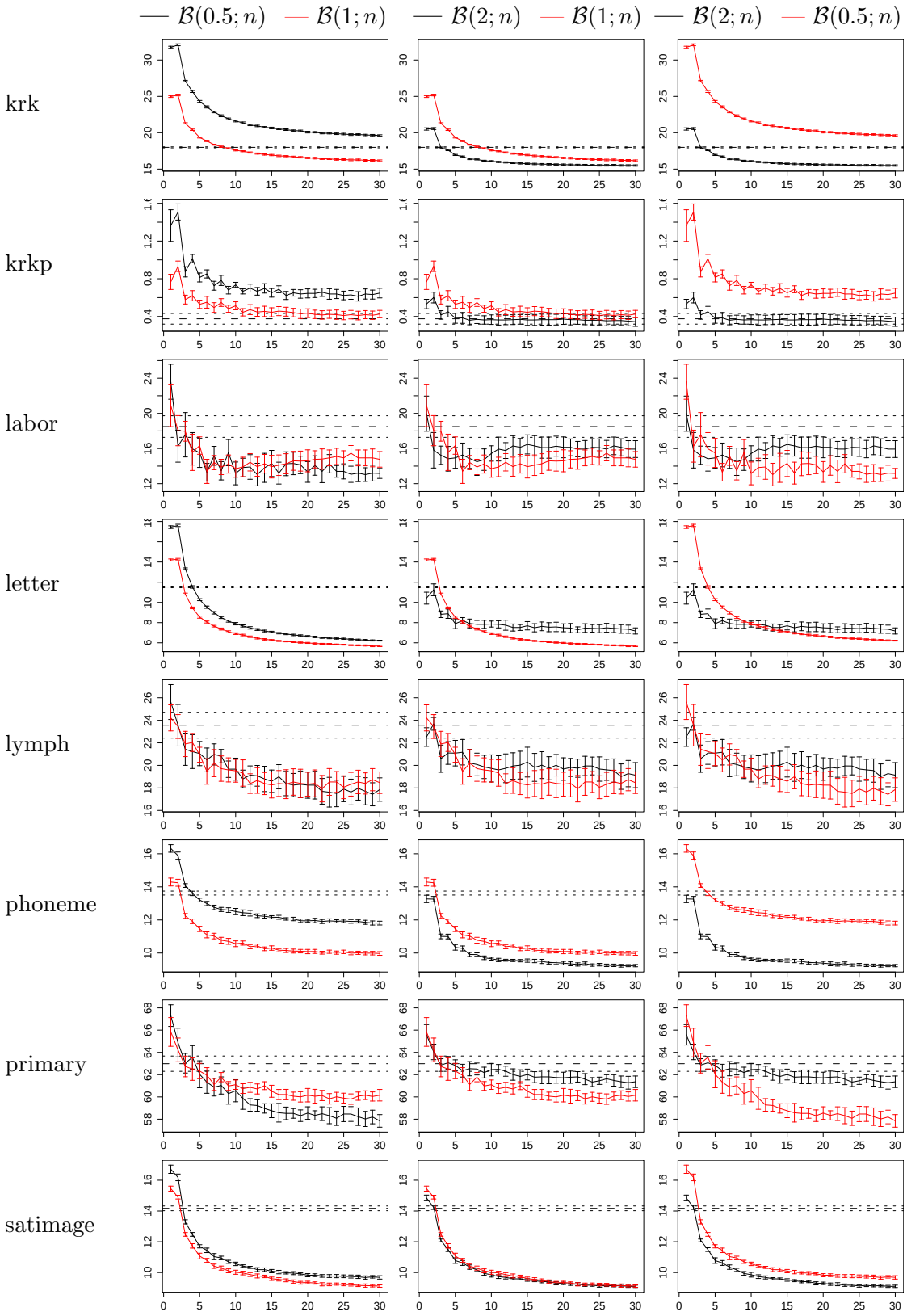

Figure 4.1: Error curves for $\mathcal{B}(0.5;n)$ versus $\mathcal{B}(1;n)$, $\mathcal{B}(2;n)$ versus $\mathcal{B}(1;n)$, and $\mathcal{B}(2;n)$ versus $\mathcal{B}(0.5;n)$.

(continued on next page)

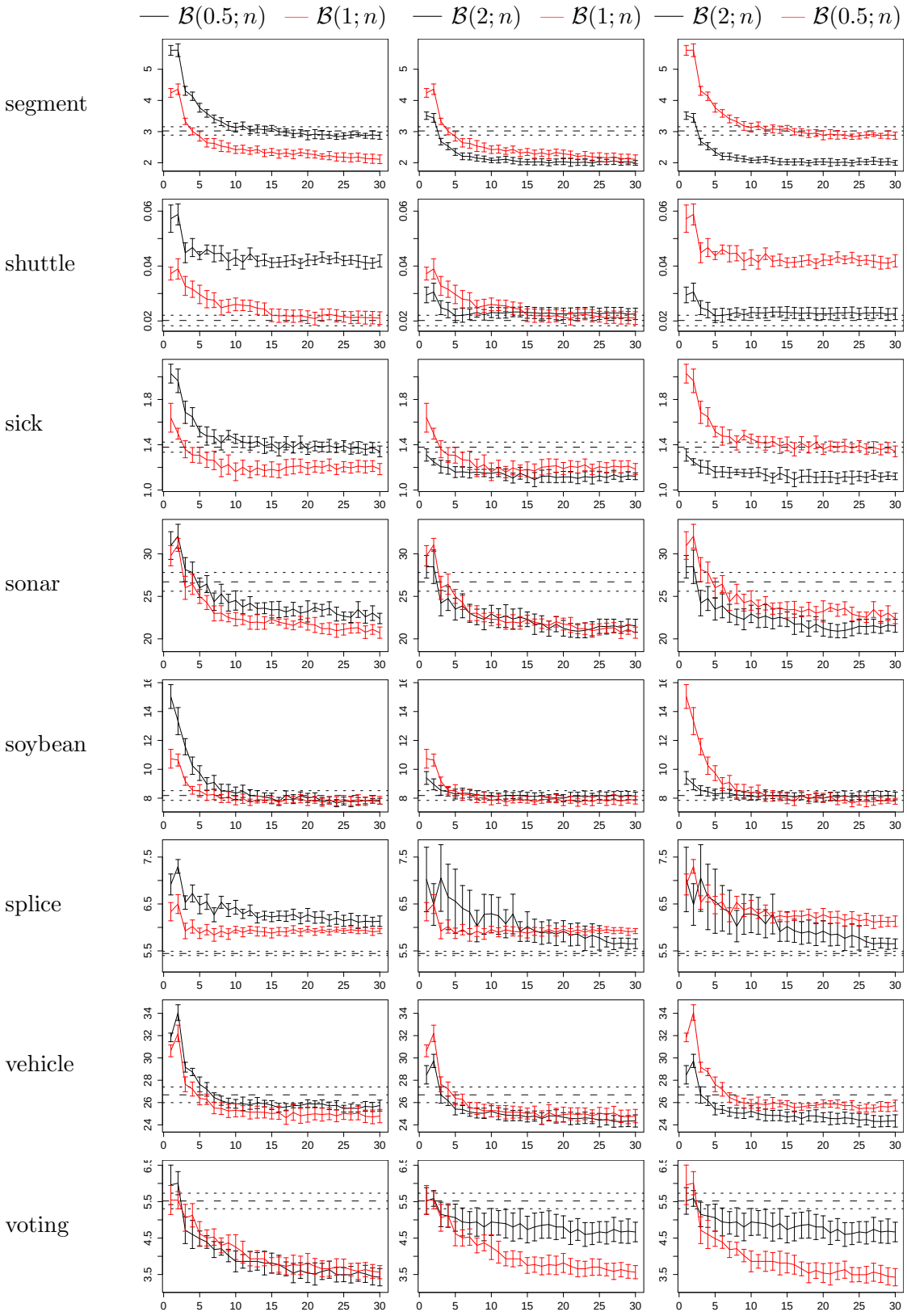

Figure 4.1: Error curves for $\mathcal{B}(0.5;n)$ versus $\mathcal{B}(1;n)$, $\mathcal{B}(2;n)$ versus $\mathcal{B}(1;n)$, and $\mathcal{B}(2;n)$ versus $\mathcal{B}(0.5;n)$.

(continued on next page)

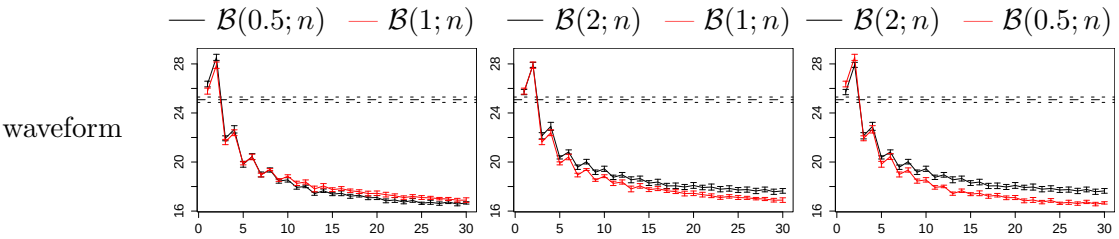

Figure 4.1: Error curves for $\mathcal{B}(0.5;n)$ versus $\mathcal{B}(1;n)$, $\mathcal{B}(2;n)$ versus $\mathcal{B}(1;n)$, and $\mathcal{B}(2;n)$ versus $\mathcal{B}(0.5;n)$.

From a "well-behaved" ensemble one would expect the ensemble loss to decrease as the ensemble size increases – we have already seen in Figure C.1 on pages 86–87 that *Cragging*$(n;1)$ is not a "well-behaved" ensemble method in this sense. This can be attributed to the fact that, with increasing n, the expected accuracy of the member classifiers will increase but the diversity will decrease, and the increase in member accuracy may not be enough to make up for the decrease in diversity. All the other ensemble methods described here are "well-behaved" in the sense that the expected ensemble loss tends to continue to decrease with increasing number of member classifiers. Intuitively and unlike for *Cragging*$(n;1)$, for both *Bagging*$(s;n)$ and *Cragging*$(f;n)$, member accuracy and diversity are dependent only on the relative size s or the number of sample partitions f, but not the number of iterations n.

Figure 4.1 and Figure 4.2 also show that the loss limit is usually reached with only a relatively small number – sometimes as few as 5 – member classifiers. This confirms previous results from e.g. [59] and [66].

Contrary to common beliefs, "normal" Bagging (*Bagging*$(1;n)$) does not always perform better than a single base classifier: it is sometimes clearly outperformed by a single decision tree, e.g. on the datasets anneal, krkp (although not significantly), and splice in Figure 4.1 – this can also be seen in Figure 3.2 on pages 21–23.

It can also be seen in Figure 4.1 and Figure 4.2 that *Bagging*$(1;n)$, the "normal" Bagging, is indeed frequently outperformed by either *Bagging*$(0.5;n)$ (balance, breastc, credita, heartc, hearth, heartv, labor, primary), *Bagging*$(2;n)$ (anneal, autos, breastc, krk, krkp, phoneme, segment, shuttle, sick, splice, vehicle), *Cragging*$(2;n)$ (balance, breastc, credita, heartc, hearth, heartv, iris, labor, primary, splice, voting), or *Cragging*$(3;n)$ (anneal, autos, colic, krk, krkp, primary, segment, sonar, splice). Either the outperforming ensemble method has a lower ensemble loss limit as the number of component classifiers grows, or the loss limit is the same as that of *Bagging*$(1;n)$ but the outperforming method reaches that limit faster, that is, with a smaller number of component classifiers (see also Figure 3.2 on pages 21–23, which shows a direct loss comparison of ensembles with 30 member classifiers).

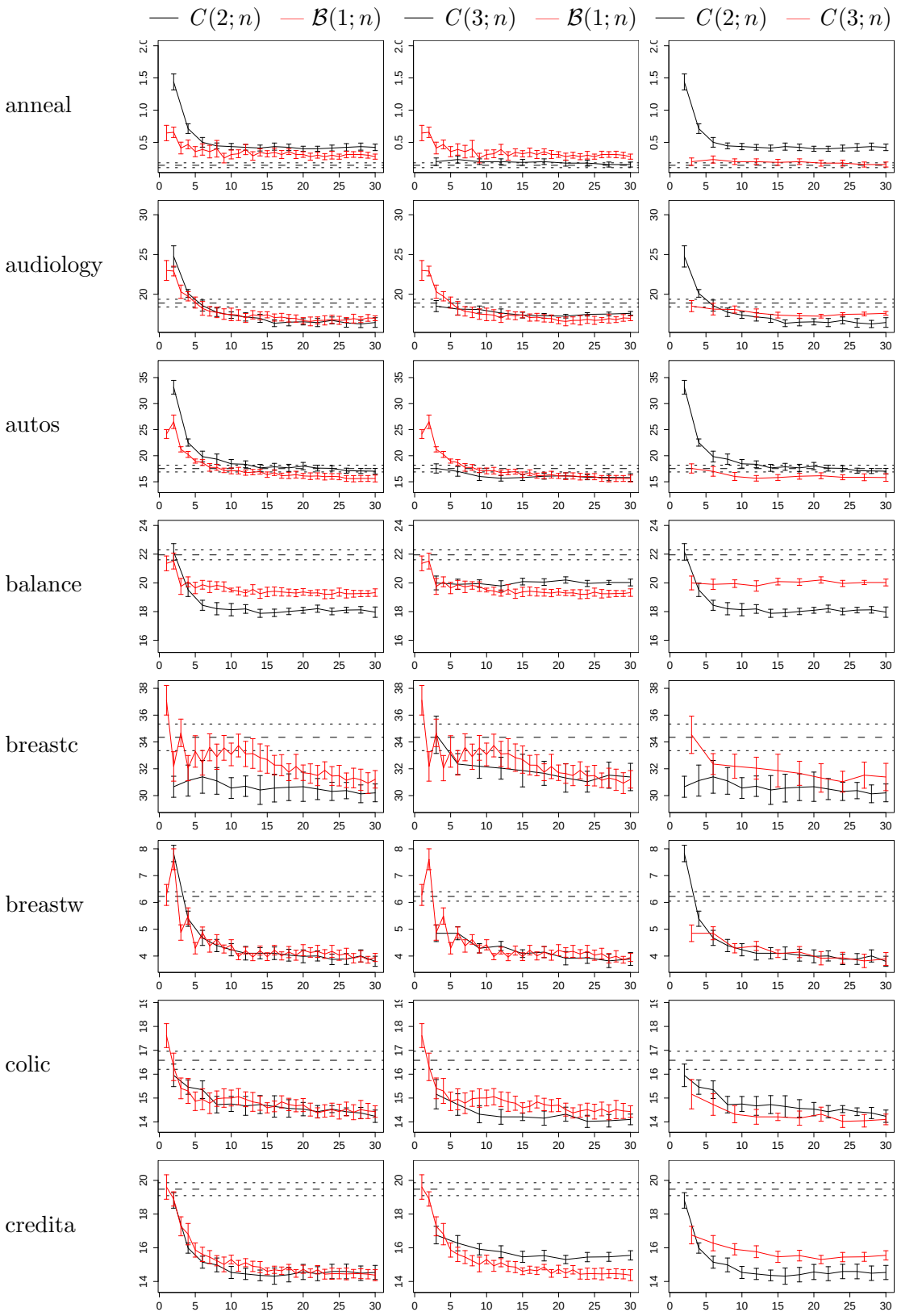

Figure 4.2: Error curves for $C(2;n)$ and $C(3;n)$ versus $\mathcal{B}(1;n)$.

(continued on next page)

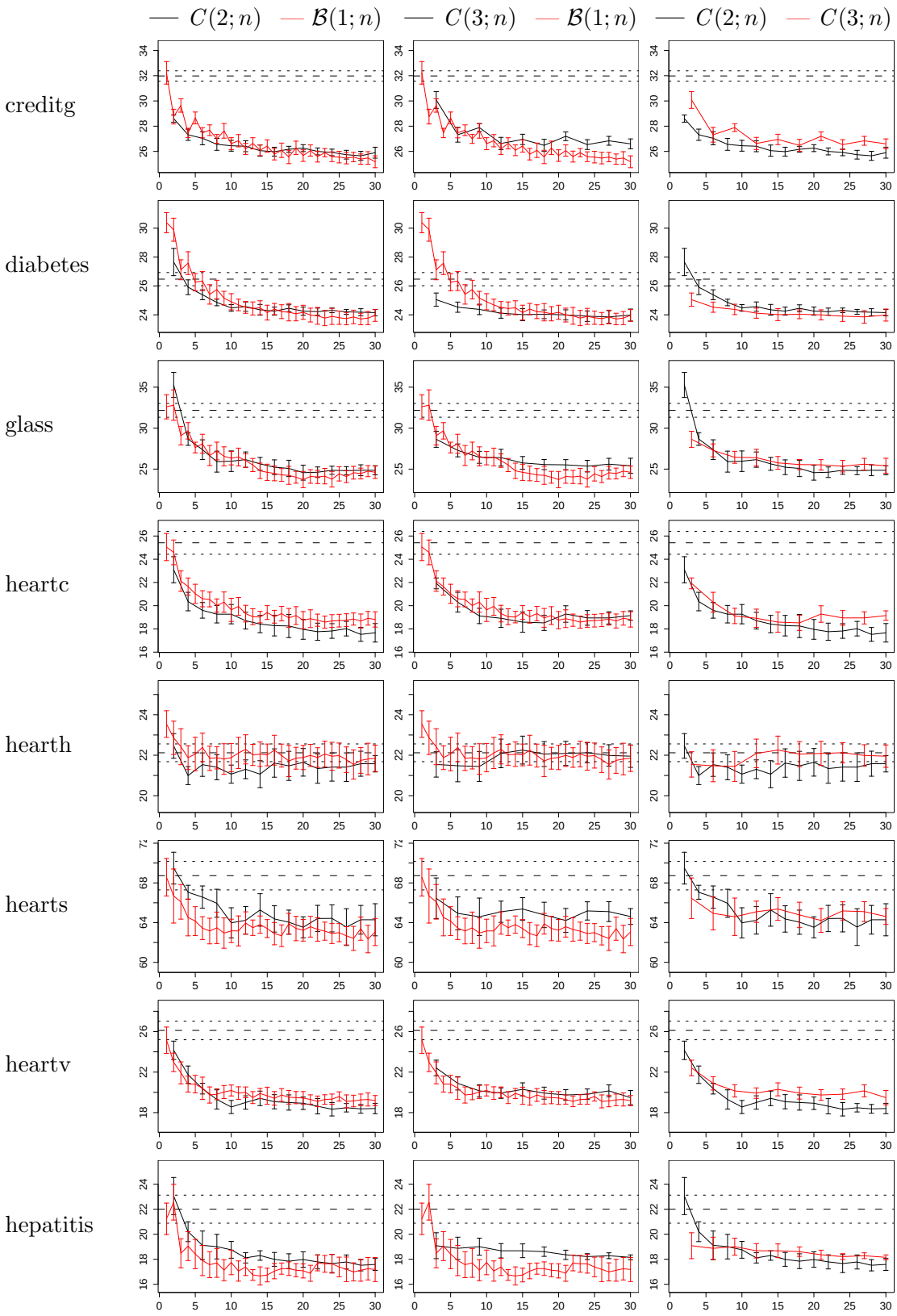

Figure 4.2: Error curves for $C(2;n)$ and $C(3;n)$ versus $\mathcal{B}(1;n)$.

(continued on next page)

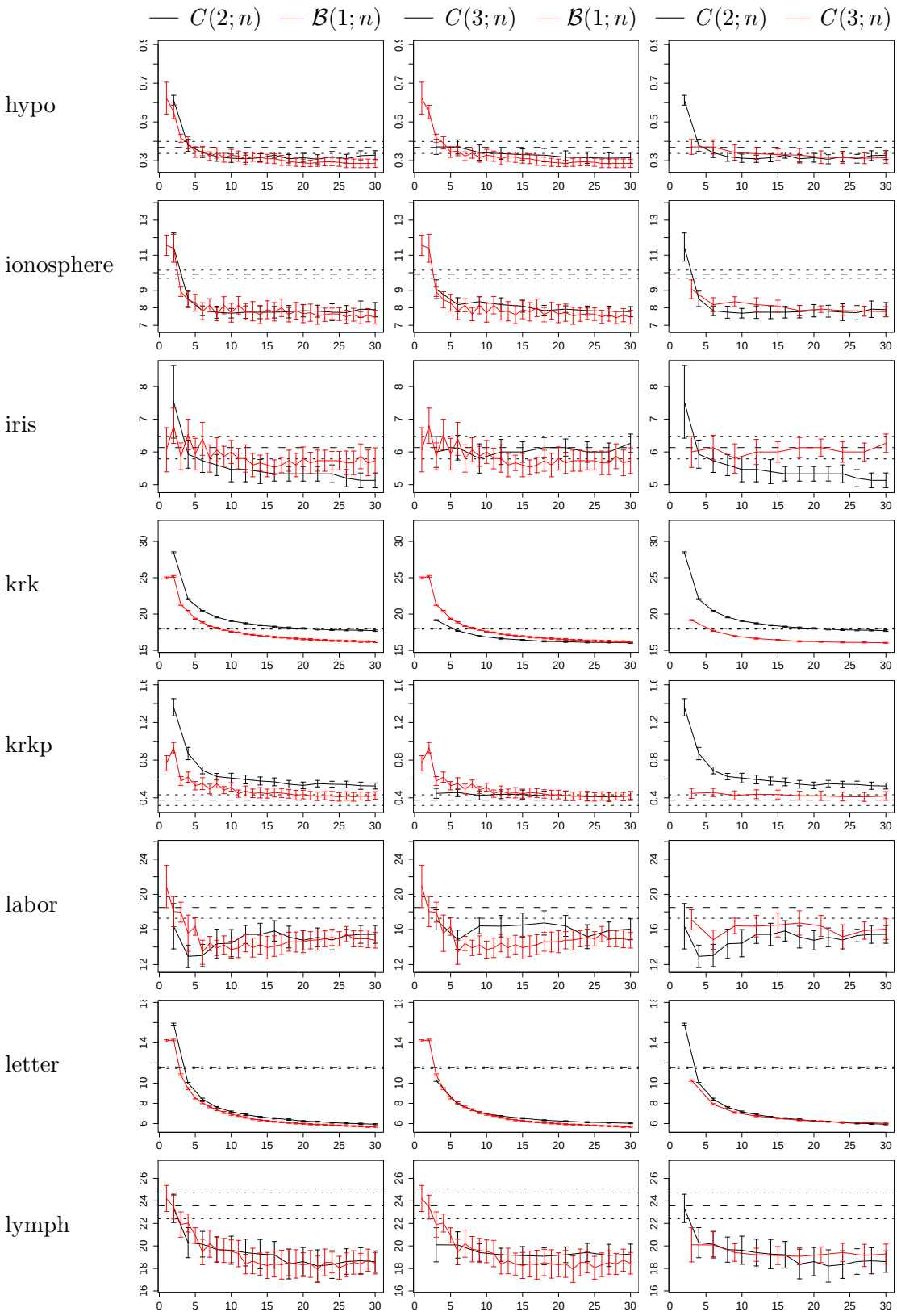

Figure 4.2: Error curves for $C(2;n)$ and $C(3;n)$ versus $\mathcal{B}(1;n)$.

(continued on next page)

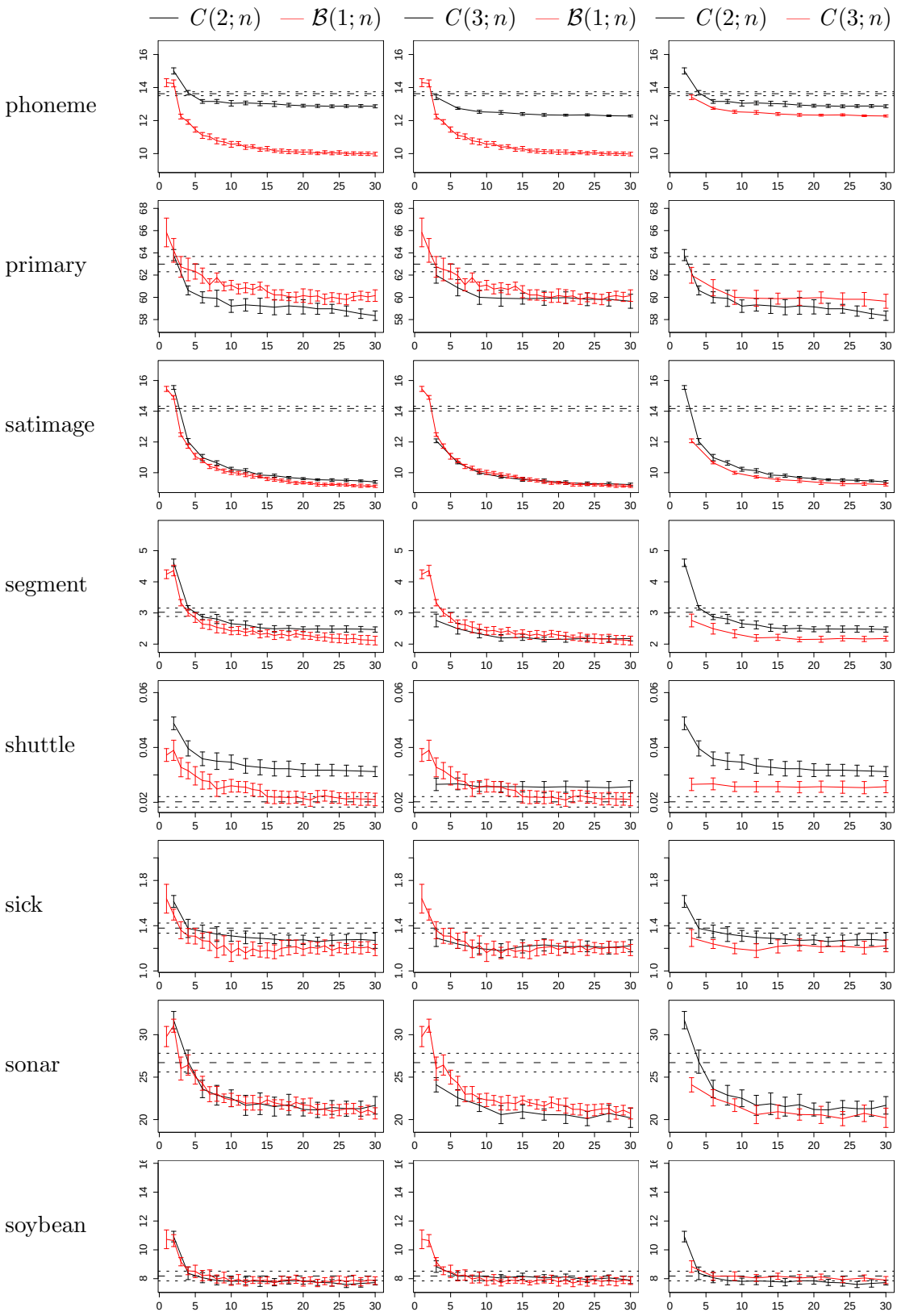

Figure 4.2: Error curves for $C(2;n)$ and $C(3;n)$ versus $\mathcal{B}(1;n)$.

(continued on next page)

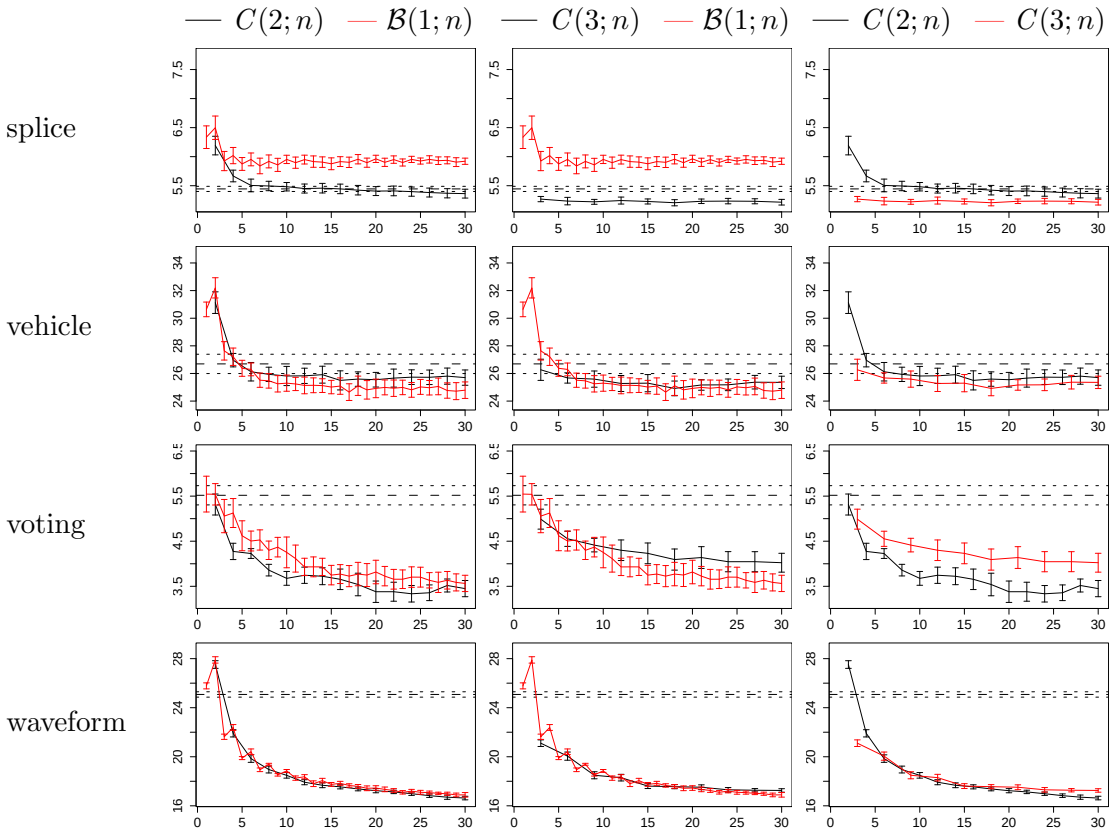

Figure 4.2: Error curves for $C(2;n)$ and $C(3;n)$ versus $\mathcal{B}(1;n)$.

These observations from Figure 4.1 and Figure 4.2 are summarized in Table 4.2. For each dataset, a "Y" is shown in those columns where the corresponding ensemble learning method has a lower loss limit than that of "normal" Bagging ($Bagging(1;n)$), and a "y" is shown where the corresponding ensemble learning method has approximately the same loss limit as "normal" Bagging, but reaches that limit using a smaller number of component classifiers. Overall, $Bagging(1;n)$ is outperformed by another sampling method on 23 out of the 36 datasets tested. Hence, it may be beneficial to also try sampling methods other than "standard" Bagging when constructing classifier ensembles.

Another interesting observation from Table 4.2 is that the relative ensemble performances seem to be correlated to each other. $Cragging(2;n/2)$ tends to outperform $Bagging(1;n)$ whenever $Bagging(0.5;n)$ outperforms $Bagging(1;n)$, and vice versa. Similarly, although less pronounced, $Cragging(3;n/3)$ tends to outperform $Bagging(1;n)$ whenever $Bagging(2;n)$ outperforms $Bagging(1;n)$, and vice versa. Furthermore, both $Cragging(2;n/2)$ and $Bagging(0.5;n)$ tend to outperform $Bagging(1;n)$ whenever $Cragging(3;n/3)$ and $Bagging(2;n)$ do not outperform $Bagging(1;n)$, and vice versa. Intuitively, both $Bagging(0.5;n)$ and $Cragging(2;n/2)$ produce ensembles with smaller member accuracy and higher diversity than those produced by $Bagging(1;n)$, while both $Bagging(2;n)$ and $Cragging(3;n/3)$ produce

Dataset	$\mathcal{B}(0.5;n)$	$\mathcal{B}(2;n)$	$C(2;n/2)$	$C(3;n/3)$
anneal		Y		Y
audiology				
autos		Y		y
balance	Y		Y	
breastc	Y	y	Y	
breastw				
colic				Y
credita	Y		y	
creditg				
diabetes				
glass				
heartc	Y		Y	
hearth	Y		y	
hearts				
heartv	Y		Y	
hepatitis				
hypo				
ionosphere				
iris			Y	
krk		Y		y
krkp		Y		y
labor	Y		y	
letter				
lymph				
phoneme		Y		
primary	Y		Y	y
satimage				
segment		y		y
shuttle		y		
sick		Y		
sonar				y
soybean				
splice		Y	Y	Y
vehicle		y		
voting	Y		y	
waveform	Y			

Table 4.2: Ensemble loss comparison summary. A "Y" is shown in those columns where the corresponding ensemble learning method has a lower loss limit than that of "normal" Bagging ($Bagging(1;n)$). A "y" is shown wherever the corresponding ensemble learning method has approximately the same loss limit as "normal" Bagging, but reaches that limit using a smaller number of component classifiers.

ensembles with higher member accuracy but lower diversity than those produced by $Bagging(1; n)$. Added to the results presented in Chapter 3, this lends further credibility to the hypothesis that mean member accuracy and diversity are the major influencing factors with regard to expected ensemble loss.

Even if two ensemble methods perform the same in terms of expected ensemble loss, it may still be advantageous to choose one over the other for computational reasons. For example, the data subsamples obtained using $Cragging(2; n/2)$ or $Bagging(0.5; n)$ contain only half the number of instances compared to that of $Bagging(1; n)$. Since most base classifiers have a time complexity at least linear in terms of the number of training instances, smaller subsamples will result in shorter run times for the ensemble learning process. When using decision tree algorithms as base learners, an additional benefit of using smaller data subsamples is that the resulting member classifiers will be decision trees with fewer nodes ([58]), which may result in better understandability of the learned ensemble, as well as faster classification of new instances.

4.3 κ-Error Diagrams

One method frequently used to analyze ensemble behavior are κ-Error Diagrams ([56]). The κ-Error Diagram is a scatterplot where each point corresponds to a pair of member classifiers. The x-coordinate of the point is the value of the kappa-statistic (κ), which is a statistical measure of agreement between two value classifiers. The y-coordinate of the point is the average 0-1 loss (error) for the two classifiers making up the pair. Hence, the points at the lower right of the κ-Error Diagram represent pairs of classifiers that are very accurate but also very similar to each other. The points at the upper left represent pairs of classifiers that are less accurate, but also less similar to each other. Points near the origin represent pairs of classifiers which exhibit the desirable properties of accuracy and diversity. Thus, κ-Error Diagrams can help visualize the accuracy and diversity of the individual classifiers making up an ensemble.

The κ statistic is computed as follows ([14, 56]). Given a discrete-valued outcome space Y with k possible outcomes, a data sample $\mathbf{S} = \langle \langle \mathbf{x}_1, y_1 \rangle, \langle \mathbf{x}_2, y_2 \rangle, \ldots, \langle \mathbf{x}_m, y_m \rangle \rangle$ with $y_i \in Y$ for $i \in \{1, \ldots, m\}$, and two value classifiers c_1 and c_2 whose prediction spaces are equal to the outcome space ($\widehat{Y}_{c1} = \widehat{Y}_{c2} = Y = \{Y_1, Y_2, \ldots, Y_k\}$), let K be a $k \times k$ matrix such that K_{ij} contains the number of instances in $\mathbf{S}$ that are classified as y_i by the first classifier and as y_j by the second classifier ($K_{ij} := |\{\langle \mathbf{x}, y \rangle \in \mathbf{S} | \widehat{y}_{c1}(\mathbf{x}) = y_i \wedge \widehat{y}_{c2}(\mathbf{x}) = y_j\}|$). Let

$$\Omega_1 := \frac{\sum_{i=1}^{k} K_{ii}}{m} \tag{4.1}$$

and

$$\Omega_2 := \sum_{i=1}^{k} \left(\sum_{j=1}^{k} \frac{K_{ij}}{m} \sum_{j=1}^{k} \frac{K_{ji}}{m} \right) \tag{4.2}$$

Then, the κ statistic is defined as

$$\kappa := \frac{\Omega_1 - \Omega_2}{1 - \Omega_2} \tag{4.3}$$

Ω_1 simply is an estimate of the probability $p(\widehat{y}_{c1}(\mathbf{x}) = \widehat{y}_{c2}(\mathbf{x}))$ that the two classifiers c_1 and c_2 agree with each other, while Ω_2 estimates the probability that c_1 and c_2 agree by chance. In problem domains with some outcomes occurring much more often than others, all reasonable classifiers would tend to agree with each other, thus all pairs of classifiers would obtain high values of Ω_1. Therefore, rather than using Ω_1 as a measure of agreement directly, the κ statistic corrects for this by also taking into account the probability that the two classifiers agree with each other simply by chance, given the observed counts K_{ij}.

The kappa statistic $\kappa = 1$ when the two classifiers agree on every example, and $\kappa = 0$ when the rate of agreement equals that expected by chance. Negative values of κ signify systematic disagreement between c_1 and c_2, that is, less agreement than that which would be expected by chance.

Appendix D contains the κ-Error Diagrams for ensembles with 30 member classifiers. Figure D.1 on pages 88–95 shows the κ-Error Diagrams for *Bagging*$(0.5; 30)$, *Bagging*$(1; 30)$, and *Bagging*$(2; 30)$. Figure D.2 on pages 95–102 shows the κ-Error Diagrams for *Cragging*$(2; 15)$, *Cragging*$(3; 10)$, and *Cragging*$(30; 1)$.

When comparing the κ-Error Diagrams in Figure D.1 for any given dataset, one can see that the pairwise average loss of the member classifiers decreases when going from *Bagging*$(0.5; 30)$ over *Bagging*$(1; 30)$ to *Bagging*$(2; 30)$, while the classifier agreement – as measured by the κ statistic – increases. Similarly, one can see from Figure D.2 that pairwise average member loss decreases while classifier agreement increases when going from *Cragging*$(2; 15)$ over *Cragging*$(3; 10)$ to *Cragging*$(30; 1)$. These results confirm our expectations from Chapter 3 regarding the behavior of member accuracy and diversity with respect to the different ensemble learning methods.

One can also see that component classifiers of the *Cragging*$(30; 1)$ ensembles are always more accurate but also less diverse that those of any other ensemble method. This does not come as a surprise either, as any two member classifiers share more than 96 percent of their training samples (see also Chapter 3).

While κ-Error Diagrams are useful for visualizing a given classifier ensemble, they have some handicaps which seriously limit their applicability to the task of predictive ensemble performance analysis as outlined in Section 1.2, that is, to be able to tell which ensemble learning method will perform well on a given problem domain, without actually having to go through the whole ensemble training process.

First, the κ statistic is defined only for value classifiers (that is, for classifiers whose

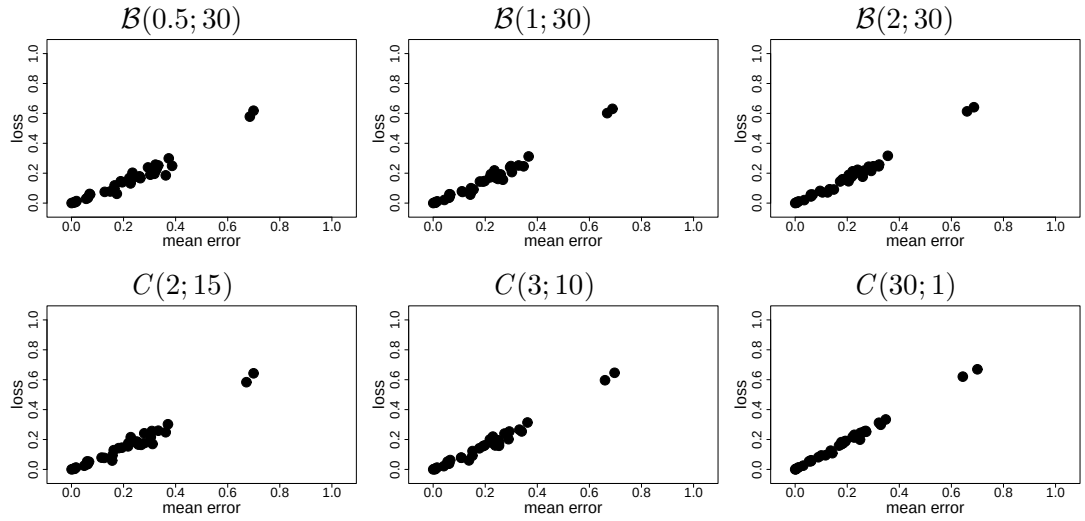

Figure 4.3: κ-Error Diagram summary – mean pairwise errors versus ensemble losses.

prediction space equals the outcome space) and as such is not applicable e.g. to voting functions other than democratic voting, or when the classifiers output a probability distribution over the outcome space. Moreover, application of κ-Error Diagrams is limited to problem domains where the outcome space is discrete-valued, as the κ statistic is defined only for such outcome spaces. Even if we were to restrict ourselves to domains with discrete-valued outcomes and to ensembles with democratic voting only, the use of κ-Error Diagrams for ensemble performance analysis only makes sense under 0-1 loss as a loss function. This is a rather serious limitation, as we frequently encounter problems where some mistakes are much more costly than others ([22]).

Second, κ-Error Diagrams provide little understanding of how the ensemble performance and the problem domain interact ([4]). For example, they don't provide answers to questions such as "On which types of examples does the ensemble perform well?", "On which types of examples are the member classifiers most/least accurate/diverse?", or "On which type of problem can I expect my ensemble learning method to perform well?". This restriction stems from the fact that the κ statistic and the average pairwise error are summary measures across the whole data sample, and is perhaps an inherent one when visualizing an ensemble of classifiers in a single graphic.

Finally, and perhaps most importantly given our goal of choosing a sampling method when presented with a problem domain or data sample, the κ value is a statistical measure of how significant the agreement between two classifiers is given the observed prediction counts, and as such inherently dependent on the distribution of outcomes in the data sample S. This makes comparison of κ-Error Diagrams across domains very difficult if not impossible.

To see this, consider Figure 4.3, Figure 4.4, and Figure 4.5, which we devised to sum-

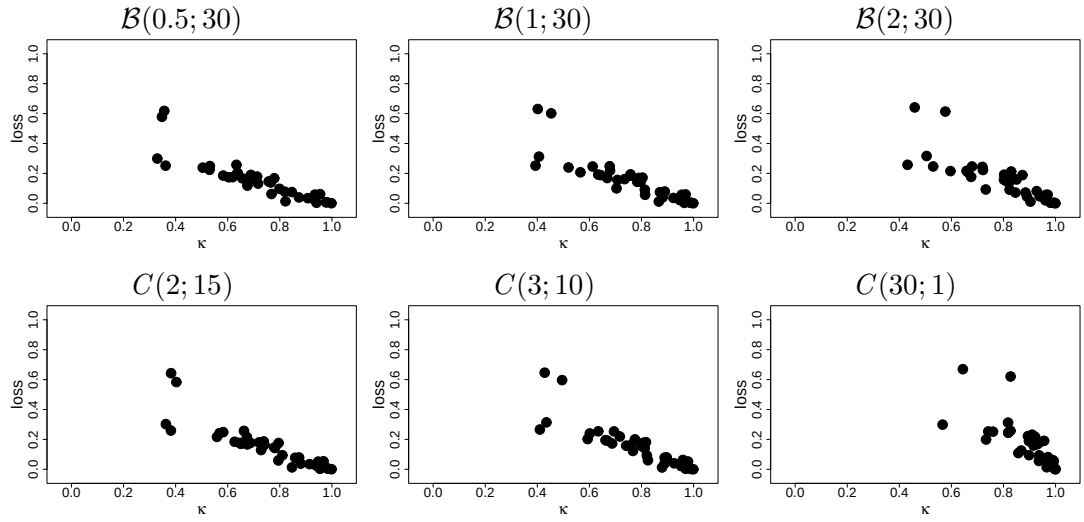

Figure 4.4: κ-Error Diagram summary – kappa values versus ensemble losses.

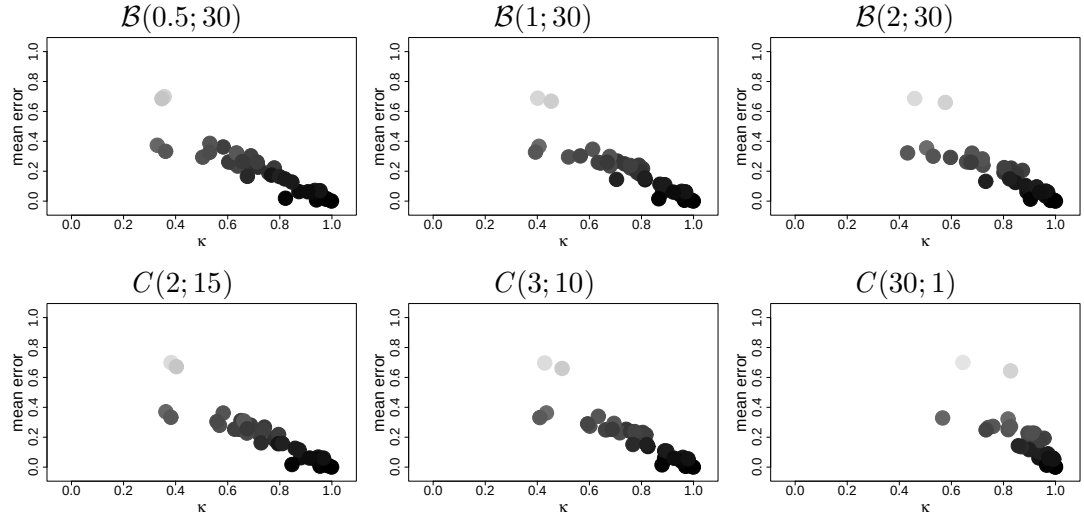

Figure 4.5: κ-Error Diagram summary. Darker points have lower ensemble loss.

marize the data from the κ-Error Diagrams shown in Figure D.1 and Figure D.2. All three figures show one scatterplot for each ensemble learning method. In the scatterplots, one point is shown for each dataset. The coordinates for each point $\langle x, y \rangle$ are obtained by averaging the κ values and the pairwise mean errors, respectively, over all possible pairs of member classifiers from the corresponding κ-Error Diagram (in Appendix D, one κ-Error Diagram is shown for each dataset and ensemble learning method).

Figure 4.3 shows the mean pairwise error of the component classifiers (x-axis) versus the 0-1 loss of the ensemble as a whole (y-axis). Similarly, Figure 4.4 shows the average κ values (x-axis) for the pairs of component classifiers versus the 0-1 loss of the ensemble (y-axis). Figure 4.5 shows the average κ values on the x-axis and the mean pairwise errors on the y-axis, while the gray-scale of the points represents the ensemble loss: darker points have lower ensemble loss than lighter points.

While there is a clear correlation between the ensemble loss and member accuracy as measured by the mean pairwise error of the component classifiers (Figure 4.3), a correlation between ensemble loss and member diversity is not nearly as evident (Figure 4.4). In fact, there even seems to be a negative correlation between ensemble loss and ensemble member diversity (remember that high κ values signify high agreement rates among member classifiers whereas small κ values signify high diversity). This can also be seen from Figure 4.5: The ensemble loss tends to decrease with increasing κ values (points get darker), even across problem domains where the mean pairwise error stays almost the same.

This is unintuitive and contrary to all previous results. We attribute this apparent discrepancy to the correction made by the κ statistic for the classifier agreement expected by chance. Intuitively, as far as the quantitative influence of member agreement on ensemble loss is concerned, it should not matter whether the agreement is systematic or "random", i.e., equal to that expected by chance. We therfore have to conclude that the κ statistic, while useful as a statistical measure of the significance of agreement, does not constitute an adequate measure of ensemble member diversity for our purposes.

4.4 Margin Distributions

In [71], Schapire et. al. presented a theoretical analysis framework of ensemble learning methods under 0-1 loss, applicable to single-stage voting ensembles of classifiers, such as e.g. in Bagging, Boosting, Arcing, ECOC, etc. They introduced a measure of an ensembles confidence into its prediction called the *margin*. For a given example $\langle \mathbf{x}, y \rangle$, the margin is the difference between the weight of votes correctly predicting the class label and the maximum weight of votes assigned to any single incorrect label. Specifically,

$$M_C(\langle \mathbf{x}, y \rangle) := \sum_{c \in C | \widehat{y}_c(\mathbf{x}) = y} w_c - \max_{y' \in Y, y' \neq y} \sum_{c \in C | \widehat{y}_c(\mathbf{x}) = y'} w_c \tag{4.4}$$

if C is an ensemble of value classifiers ($\widehat{Y}_c = Y \ \forall c \in C$), and

$$M_C(\langle \mathbf{x}, y \rangle) := \sum_{c \in C} w_c \widehat{p}_c(y|\mathbf{x}) - \max_{y' \in Y, y' \neq y} \sum_{c \in C} w_c \widehat{p}_c(y'|\mathbf{x}) \tag{4.5}$$

if C is an ensemble of distribution classifiers ($\widehat{Y}_c = \{\widehat{P}(Y)\} \ \forall c \in C$). The margin can also be defined for a single distribution classifier, namely as

$$M_C(\langle \mathbf{x}, y \rangle) := \widehat{p}_C(y|\mathbf{x}) - \max_{y' \in Y, y' \neq y} \widehat{p}_C(y'|x) \tag{4.6}$$

In the case where the ensemble is a distribution classifier ($\widehat{Y} = \{\widehat{P}(Y)\}$) whose members are distribution classifiers as well ($\widehat{Y}_c = \{\widehat{P}(Y)\}$), the definitions of the margin according to Equation 4.5 and Equation 4.6 coincide if the ensemble decides on its prediction by probabilistic voting (Equation 2.11). They don't necessarily coincide if other voting schemes (such as e.g. majority voting as in Equation 2.9) are employed.

The margin is a number in the range $[-1, 1]$, and an instance $\langle \mathbf{x}, y \rangle$ is classified correctly if and only if its margin $M_C(\langle \mathbf{x}, y \rangle)$ is positive. Schapire et. al. showed in [71] that an ensemble's generalization error (that is, the expected 0-1 loss on test instances) can be bound in terms of the distribution of margins on the training sample, the sample size, and the VC dimension of the base learner. In particular, the lower the probability of a small margin on the training sample, the lower the bound on expected ensemble loss.

Schapire et. al. also state, however, that the bounds presented in [71] are too loose to allow practical quantitative predictions of ensemble loss. Nevertheless, they propose the use of margin distributions to explain the success of ensemble learning algorithms such as Bagging and Boosting, and argue that ensembles primarily work because they minimize the probability of low margins on the training sample. Apart from the fact that application of margin analysis is sensible only for problems with discrete-valued outcomes, under 0-1 loss, and with certain voting functions, this proposed explanation has also been subsequently criticized in the literature for being inaccurate or incomplete.

For example, J.R. Quinlan demonstrates in [67] that classifiers which have the same margin distributions on the training sample may nevertheless exhibit significantly different 0-1 loss on the test sample. L. Breiman introduced an alternative analysis framework based on a function he called the *edge*, and notes that his framework "gives results which are the opposite of what we would expect given Schapire et al.'s explanation of why arcing works" ([10]). Within the context of linear classifiers, Herbrich and Graepel note that the margin is too coarse a measure to give sufficiently tight bounds on generalization error ([41]). In [37], A. Grove and D. Schuurmans present modified boosting algorithms that achieve larger minimum margins on the training sample than Adaboost, yet generally fail to yield better performance on test data ([37]). And in [40], M. Harries induced ensembles whose members all fit the training data perfectly – that is, the ensemble has a minimum margin of 1 – and

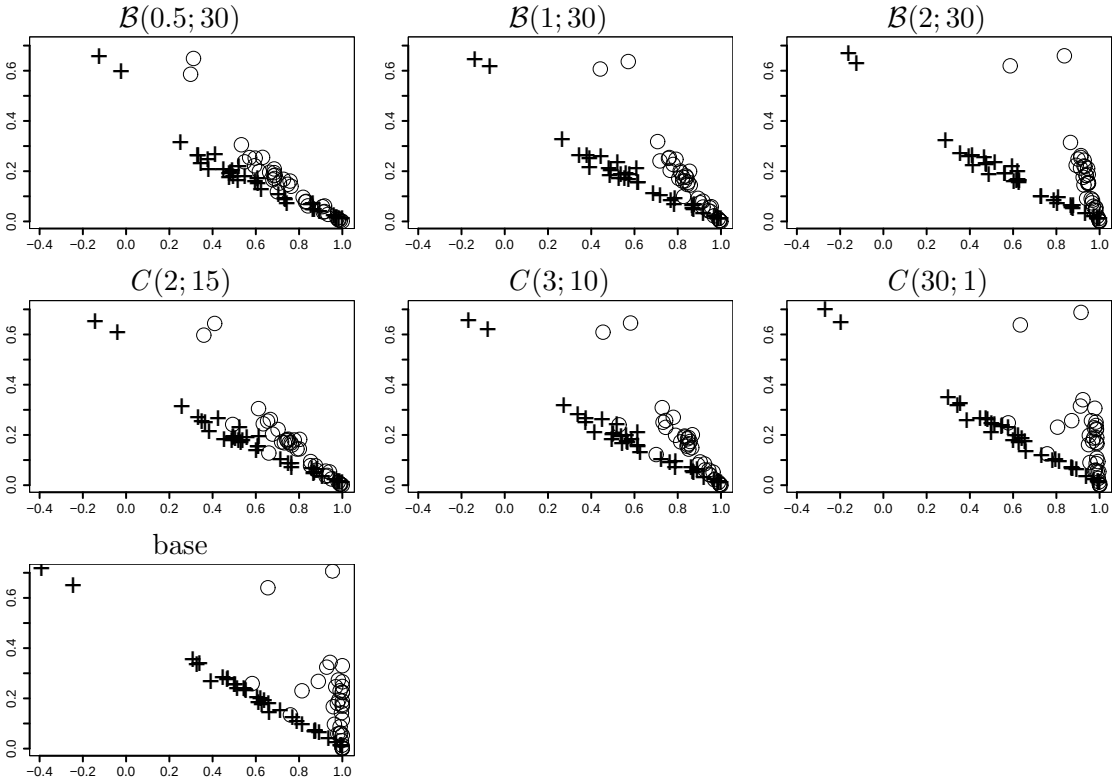

Figure 4.6: Average margins on training (o) and test (+) data (x-axis) versus average loss on test data (y-axis).

showed that generalization performance can still be improved after the maximum value for the minimum margin is already obtained.

Appendix E (pages 103–123) shows the cumulative margin distributions on both training set sample and test sample for ensembles containing 30 member classifiers, for the ensemble learning methods considered in the experimental part of this thesis: *Bagging*$(0.5;30)$, *Bagging*$(1;30)$, *Bagging*$(2;30)$, *Cragging*$(2;15)$, *Cragging*$(3;10)$, and *Cragging*$(30;1)$. Figure E.7 on pages 121–123 shows the cumulative margin distributions for a single base classifier. The margins are computed according to Equation 4.6, and the curves shown are obtained by averaging the 100 distributions obtained from 10 runs of 10-fold cross-validation.

We devised Figure 4.6 to summarize those results from Appendix E. For each of the ensembles as well as the base classifier, Figure 4.6 shows the average margins on the training ("o" points) and test ("+" points) samples on the x-axis versus the expected 0-1 loss on the test samples (y-axis) for each of the datasets from Table 4.1. The average margin for classifier C on data sample $\mathbf{S}$ is simply

$$\mathrm{MAVG}_C(\mathbf{S}) := \frac{1}{|\mathbf{S}|} \sum_{\langle \mathbf{x}, y \rangle \in \mathbf{S}} M_C(\langle \mathbf{x}, y \rangle) \tag{4.7}$$

According to [71], we should get lower ensemble losses with higher margins, and vice

versa. While this relationship holds at least approximately for the margins on the test samples, it does not hold for the training sample margins - and therefore training sample margins can not be used to predict ensemble behaviour on unseen test data. Quite often a classifier C_1 has both lower (higher) training sample margins and lower (higher) loss than another classifier C_2. We also note that the *Cragging*$(30; 1)$ ensembles seem to have the highest average training sample margins (which is not surprising as the *Cragging*$(30; 1)$ member classifiers can be expected to fit the original training sample most perfectly), but from Figure 3.2 we already know that *Cragging*$(30; 1)$ is actually the worst performing ensemble learning method out of the ones shown in Figure 4.6.

This is contradictory to the results we would expect from the analysis framework in [71], and shows that margin analysis alone can not be used to analytically explain ensemble behavior.

4.5 Bias-Variance Decomposition

The bias-variance decomposition is a widely-used theoretical tool in machine learning not only with respect to ensemble learning, but for developing and understanding prediction algorithms in general. It was originally developed in [33] for the case of squared loss, and recently several authors (e.g. [9, 31, 50, 48, 75]) have proposed bias-variance decompositions for 0-1 loss. Subsequently, Domingos ([24, 25, 26]) presented a unified theoretical framework for consistent application of bias-variance decompositions under different loss functions.

Let $\widehat{y}_{\text{opt}}(\mathbf{x}) := \text{argmin}_{\widehat{y}\in Y} E_{P(Y|\mathbf{x})}\left[l(\widehat{y}, y)\right]$ be the optimal prediction given input $\mathbf{x}$, i.e., the prediction with minimal expected loss out of all possible predictions $\widehat{y} \in \widehat{Y}$.

Imagine that the process of inducing and testing classifiers is repeated many times, each time using the same classifier inducer J but different training samples $\mathbf{S}$, which are obtained by sampling from $P(\mathbf{X}, Y)$. Then let

$$\widehat{y}_{\text{m}}(\mathbf{x}) := \text{argmin}_{\widehat{y}\in Y} E_{P(\mathbf{S})}\left[l(\widehat{y}, \widehat{y}_{J(\mathbf{S})}(\mathbf{x}))\right] \tag{4.8}$$

be the main prediction for input $\mathbf{x}$, that is, the prediction with minimal expected loss with respect to the predictions made by classifiers induced by J from the training samples $\mathbf{S}$. The main prediction $\widehat{y}_{\text{m}}(\mathbf{x})$ can also be interpreted as the prediction made by some (fictive) "average" predictor which in turn would be induced by J from an "average" training sample.

The expected loss $E_{P(\mathbf{S})}\left[L(J, \mathbf{x})\right] = E_{P(\mathbf{S}),P(Y|\mathbf{x})}\left[l(\widehat{y}_{J(\mathbf{S})}(\mathbf{x}), y)\right]$ for input $\mathbf{x}$ of classifiers learned by classifier inducer J is decomposed into three terms ([24]):

- The *statistical bias* $B(J, \mathbf{x}) := l(\widehat{y}_{\text{m}}(\mathbf{x}), \widehat{y}_{\text{opt}}(\mathbf{x}))$ is the loss of the main prediction $\widehat{y}_{\text{m}}(\mathbf{x})$ relative to the optimal prediction $\widehat{y}_{\text{opt}}(\mathbf{x})$. This is a sensible performance measure because it expresses what we can expect the generalization error to be on "average".

- The *statistical variance* $V(J, \mathbf{x}) := E_{P(S)}\left[l(\widehat{y}_{J(\mathbf{S})}(\mathbf{x}), \widehat{y}_{\mathrm{m}}(\mathbf{x}))\right]$ measures how the predictions of the classifiers induced by J vary around the main prediction, i.e., how the choice of the training sample effects the generalization error.

- The *noise* $N(\mathbf{x}) := E_{P(Y|\mathbf{x})}\left[l(\widehat{y}_{\mathrm{opt}}(\mathbf{x}), y)\right]$ is the expected loss of the optimal prediction $\widehat{y}_{\mathrm{opt}}(\mathbf{x})$ relative to the true outcomes y. This is the unavoidable component of the loss and is incurred independently of the classifier inducer J.

The underlying fundamental insight is that reducing either one of bias or variance without increasing the other will result in a reduced expected loss. A classifier inducer, when presented with a training sample S, has to choose one "best" classifier out of a set of possible candidate classifiers. The set of candidate classifiers is usually called *hypothesis space* or *model space* and is a property inherent to the inducer. When presented with different training samples, simple inducers (i.e., inducers with small model spaces) will tend to produce classifiers whose predictions are similar (low variance) but less than optimal (high bias). In contrast, more powerful inducers employing larger spaces of candidate classifiers will produce classifiers whose predictions, on "average", will be closer to the optimal prediction (low bias), but will fluctuate around this "average" (high variance).

Specifically, in [24] it is shown that, for the case of 0-1 loss,

$$E_{P(\mathbf{S})}\left[L(\mathbf{x})\right] = E_{P(\mathbf{S}),P(Y|\mathbf{x})}\left[l(\widehat{y}_{J(\mathbf{S})}(\mathbf{x}), y)\right] = c_1 N(\mathbf{x}) + B(J, \mathbf{x}) + c_2 V(J, \mathbf{x}), \quad (4.9)$$

where

$$\begin{aligned} c_1 = \;& p_{P(\mathbf{S})}(\widehat{y}_{J(\mathbf{S})}(\mathbf{x}) = \widehat{y}_{\mathrm{opt}}(\mathbf{x})) \\ & -p_{P(\mathbf{S})}(\widehat{y}_{J(\mathbf{S})}(\mathbf{x}) \neq \widehat{y}_{\mathrm{opt}}(\mathbf{x}))p_{P(\mathbf{S}),P(Y|\mathbf{x})}(y = \widehat{y}_{J(\mathbf{S})}(\mathbf{x})|y \neq \widehat{y}_{\mathrm{opt}}(\mathbf{x})) \end{aligned} \quad (4.10)$$

and

$$c_2 = \begin{cases} 1 & \text{iff } \widehat{y}_{\mathrm{m}}(\mathbf{x}) = \widehat{y}_{\mathrm{opt}}(\mathbf{x}) \\ -p_{P(\mathbf{S})}(\widehat{y}_{J(\mathbf{S})}(\mathbf{x}) = \widehat{y}_{\mathrm{opt}}(\mathbf{x})|\widehat{y}_{J(\mathbf{S})}(\mathbf{x}) \neq \widehat{y}_{\mathrm{m}}(\mathbf{x})) & \text{otherwise} \end{cases} \quad (4.11)$$

The concepts of bias and variance have been used to explain the empirical success of ensemble methods such as Bagging and Cragging. In [9], Breiman claims that Bagging, as well as other ensemble algorithms, reduces the variance portion of the loss while keeping the bias (almost) unchanged. Intuitively, while allowing a more intensive search for a single classifier is liable to increase variance, averaging the predictions of multiple classifiers is likely to reduce it ([24]). However, Breiman himself acknowledged that this is only part of the story ([9]). For example, if the base classifier is very "stable" (i.e, has a low variance), the reduction in variance will we negligible. Other authors have also criticized this explanation for being insufficient. For example, theoretical work of Buja and Stuetzle ([12, 13]) provides examples for which Bagging is proven to increase both bias and variance. Schapire and Singer note that a large variance of the base classifier inducer is not a requirement for

ensemble learning to be effective. Moreover, ensemble methods can increase the variance while still reducing the generalization error ([72]). Grandvalet ([34, 35, 36]) showed through simple experiments that Bagging may either reduce or increase the variance for one and the same problem domain, depending on the base classifier inducer. And in [1], simulations with artificial data show that drawing a new training sample for training each classifier – a process that ensemble methods such as Bagging supposedly imitate in order to achieve their improved performance – may not yield the same reductions of ensemble loss as other ensemble learning methods.

Appendix F shows the bias-variance decomposition results for each of the ensemble learning methods considered here, as well as for the base classifier inducer. For each ensemble learning method, there are two tables. The first table shows the absolute values of the bias and variance measurements whereas the second table shows the ratios relative to a single base classifier.

Following [24], we also show the values for the contribution to variance from unbiased examples

$$V_U(J) := E_{P(\mathbf{X})}\left[(1 - B(J, \mathbf{x}))V(J, \mathbf{x})\right] \tag{4.12}$$

and the contribution to variance from biased examples

$$V_B(J) = E_{P(\mathbf{X})}\left[c_3 B(J, \mathbf{x})V(J, \mathbf{x})\right] \tag{4.13}$$

where

$$c_3 = \begin{cases} 1 & \text{iff } \widehat{y}_{\mathrm{m}}(\mathbf{x}) = \widehat{y}_{\mathrm{opt}}(\mathbf{x}) \\ p_{P(\mathbf{S})}(\widehat{y}_{J(\mathbf{S})}(\mathbf{x}) = \widehat{y}_{\mathrm{opt}}(\mathbf{x}) | \widehat{y}_{J(\mathbf{S})}(\mathbf{x}) \neq \widehat{y}_{\mathrm{m}}(\mathbf{x})) & \text{otherwise} \end{cases} \tag{4.14}$$

The net variance $V(J)$ is the difference $V_U(J) - V_B(J)$.

A practical difficulty with measuring bias and variance is that, unlike in theory, we have only one training sample S. Following other authors ([48, 24]), the multiple training sets are therefore simulated using bootstrap resampling. There are, however, two problems with this:

First, the bootstrap sampling process will change the distribution of instances presented to the classifier inducer compared to the distribution of instances in the original training sample. Thus, classifiers are induced using one instance distribution but evaluated using another. Our experiments show that this will result in loss estimates which are generally too large (see Appendix J for detailed results). The effects of the bootstrap sampling procedure on the estimates of bias and variance are unknown.

Second, as the bootstrap samples are not representative of the real underlying population but only of the one original data sample, the bias-variance decomposition statistics can only be approximations to the true values. The accuracy of those approximations is unknown. This by itself is not an uncommon occurrence in Statistics and Machine Learning. Usually we can find better approximations by drawing larger data samples. Here, however, changing the size of the data sample would change the

Method	*Loss*	*Bias*	*Var*	Var_U	Var_B
Base Classifier	20.24	15.64	4.60	8.14	3.54
Bagging$(0.5; 30)$	17.37	15.20	2.18	4.68	2.50
Bagging$(1; 30)$	17.49	15.30	2.19	4.86	2.67
Bagging$(2; 30)$	17.98	15.39	2.59	5.41	2.81
Cragging$(2; 15)$	17.43	15.15	2.27	4.80	2.53
Cragging$(3; 10)$	17.76	15.21	2.55	5.20	2.65
Cragging$(30; 1)$	19.58	15.70	3.87	7.25	3.37

Table 4.3: Averages from bias-variance decomposition (absolute values).

Method	*Loss*	*Bias*	*Var*	Var_U	Var_B
Base Classifier	1.00	1.00	1.00	1.00	1.00
Bagging$(0.5; 30)$	0.88	1.12	0.48	0.59	0.91
Bagging$(1; 30)$	0.86	1.00	0.54	0.63	0.80
Bagging$(2; 30)$	0.88	0.97	0.71	0.73	0.81
Cragging$(2; 15)$	0.86	1.03	0.50	0.60	0.78
Cragging$(3; 10)$	0.87	0.99	0.58	0.66	0.79
Cragging$(30; 1)$	0.97	0.99	0.87	0.91	0.93

Table 4.4: Averages from bias-variance decomposition (ratios relative to the base classifier).

actual variables we want to measure, namely bias and variance of the learner. This dilemma only can be solved by using data sources where the underlying population parameters themselves are known and arbitrarily many samples can be generated, i.e., artificial data.

Thus, while we value the bias-variance decomposition as an important tool to analyze general behaviour of learners based on artificial data, we believe its applicability to real-world problems by practitioners to be rather limited.

For each dataset, we executed 10 runs of 10-fold cross-validation to obtain 100 estimates of bias, variance, etc. The mean and standard deviations over those 100 estimates are shown in Appendix F. To obtain each one of those 100 estimates, we induced 30 classifiers – or 30 classifier ensembles, each consisting of 30 base classifiers – from 30 training samples, which in turn where obtained by standard bootstrap resampling the training partition of the corresponding cross-validation fold. We then measured bias and variance of the classifiers on the test partition of the corresponding cross-validation fold. As the noise level $N(x)$ is very difficult to estimate, we assume $N(\mathbf{x}) = 0$ as in [48, 24], and let the noise be part of the bias estimates $B_J(\mathbf{x})$.

Table 4.3 and Table 4.4 summarize the results from Appendix F by averaging over

the datasets, allowing for a comparison of the average behavior of the ensemble methods across the whole range of datasets. Table 4.3 shows the average absolute values of the bias-variance decomposition statistics, while Table 4.4 show the average ratios relative to the base classifier inducer.

While it is true that Bagging usually reduces variance, this is not always the case – e.g. audiology dataset in Table F.2 on 125, which shows the bias-variance decomposition results of "standard" Bagging with 30 member classifiers ($Bagging(1;30)$) divided by those of the base classifier inducer. Also, the amount of the variance reduction varies greatly with the dataset. The bias is increased for some datasets and decreased for others. As a result, while the loss usually decreases (due to the variance reduction being larger than the increase in bias), the loss may also increase (e.g. shuttle dataset in Table F.2).

Different sampling schemes also have different impact on bias and variance – the increase/decrease of bias and variance may vary greatly with the sampling scheme. It is unclear how this variation can be related to dataset characteristics.

There are also some unresolved theoretical issues which affect the practical application of bias-variance decompositions. For example, the decomposition in Equations 4.9 through 4.11 is rather complex. Unfortunately, it has been shown that a nice and simple additive decomposition – such as the one for squared loss – does not exist for the case of 0-1 loss ([38]). Also, given some component classifiers, different ensembles can be constructed from those component classifiers via different voting functions. These ensembles will usually exhibit different ensemble loss (e.g. in [3]), something that currently can not be analyzed or explained with bias-variance decompositions. Finally, bias-variance decompositions still leave unanswered the question of how the success of ensemble learning methods can be related to domain characteristics.

The bias-variance decomposition is undoubtedly an invaluable theoretical tool for understanding classification and regression algorithms and has rightfully become a cornerstone of machine learning. Nevertheless, its practical applicability to the task of deciding which ensemble method to choose for which kind of problem domain is rather limited.

5. Loss Decomposition

In Chapter 3, we discussed the accuracy-diversity trade-off as a general, intuitive explanation for why ensembles outperform single classifiers: It is widely accepted [18, 39, 52] that a good ensemble is one whose members are both accurate and diverse.

Given a test example $\langle \mathbf{x}, y \rangle$ drawn from $\mathbf{X} \times Y$ according to $P(\mathbf{X}, Y)$, the error of the member classifiers – as a measure of accuracy – can be easily quantified, it is simply their expected loss. Here, the expectation is taken over the set of member classifiers: $\overline{L}(\mathbf{x}, y) = E_{c \in C}\left[L_c(\langle \mathbf{x}, y \rangle)\right] = \frac{1}{n}\sum_{c=1}^{n} l(\widehat{y}_c(\mathbf{x}), y)$, where y is the true value and $\widehat{y}_c(x)$ is the prediction made by the member classifier c for the given test example. Given the error, determining the accuracy is straightforward. This definition easily extends to multiple test examples. Currently, however, there is no universally agreed upon way to measure the diversity of the member models. Instead, different measures of diversity are used for different loss functions and by different authors, e.g. [18, 53, 64]. Accordingly, no unifying theory for analyzing the exact dependencies of the ensemble performance on the accuracy and diversity of its members exists [43, 69].

In this chapter we present a general definition of ensemble diversity that can be applied under any given loss function. We also propose a unified form for decomposing the loss of a classifier ensemble into the mean loss of the individual ensemble members and a term $\overline{D}(\mathbf{x})$ which is a direct measure of the diversity of the ensemble members. We show that the well-known decomposition under squared loss is a special case of this unified decomposition. We then instantiate the unified decomposition for the 0-1 loss function and derive formulas for the ensemble loss of democratically voting ensembles as a function of mean member loss and diversity, which provide insights into ensemble behavior.

5.1 Voting Schemes

In what follows, we will formalize the notion of diversity among the predictions of the ensemble members, in order to allow a formal, quantitative analysis of the accuracy-diversity trade-off. We would like this analysis framework to cover as many different ensemble learning algorithms as possible. However, as with any model, we have to make a compromise between generality and complexity of the analysis framework.

We chose to make the following two assumptions (which translate to certain restrictions on the ensemble learning methods covered by the framework).

First, we require all the member classifiers to be distribution classifiers. That is, we require each member classifier c, given input x, to output a belief distribution $\widehat{P}_c(Y|\mathbf{x})$ over the outcome space Y. This is not too strong a limitation, as value classifiers can be transformed easily into distribution classifiers using $\widehat{p}_c(y|\mathbf{x}) := 1$ if $y = \widehat{y}_c$ and $\widehat{p}_c(y|\mathbf{x}) := 0$ otherwise, for all $y \in Y$. Other types of classifiers besides value classifiers and distribution classifiers also exist (e.g, ranking classifiers), but either their use is currently rather limited, or their output could in principle be computed from an (internal) intermediate belief distribution anyway.

Second, in order to make the quantitative analysis feasible, we place a restriction on the voting schemes that can be used. We do so by defining the following class of voting functions:

Definition 5.1. *Let $l : \widehat{Y} \times Y \to \mathcal{R}$ be a loss function, let $\mathbf{c} = \langle c_1, \ldots, c_n \rangle$ be an ordered tuple of distribution classifiers, and let $\mathbf{w} = \langle w_1, \ldots, w_n \rangle$ be a weight vector such that $\sum_{c=1}^{n} w_c = 1$. Then, the* democratic *voting function is defined as*

$$V_{\text{dem}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) := \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x}) l(y', y'')\, dy''. \tag{5.1}$$

A single-stage voting ensemble $\langle n, \mathbf{c}, \mathbf{w}, V \rangle$ whose members are all distribution classifiers is called a democratic *ensemble if and only if its voting function V is equivalent to the democratic voting function V_{dem}.*

Definition 5.1 instantiates the requirement for an ensemble to output a prediction $\widehat{y}(\mathbf{x})$ which constitutes a compromise among the predictions $\widehat{P}_c(Y|\mathbf{x})$ of the individual ensemble members. Among all the possible predictions an ensemble could make for a given input, the ensemble prediction is the one that minimizes the expected loss of the ensemble prediction, according to the weighted predictions made by each of the individual members for the given input. Note that Equation 5.1 can also be written as

$$V_{\text{dem}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) := \arg\min_{y' \in Y} E_C \left[E_{\widehat{P}_c(Y''|\mathbf{x})} \left[l(y', y'') \right] \right]. \tag{5.2}$$

For the case that all the weights are equal, and the member classifiers as well as the ensemble itself are value classifiers ($\widehat{Y}_c = \widehat{Y} = Y$), Definition 5.1 specializes to the standard voting functions under all the commonly used loss functions. For example, under zero-one loss, the ensemble prediction $\widehat{y}(\mathbf{x})$ is the value most frequently predicted by the member classifiers. Under squared loss, it is the mean; and under absolute loss it is the median of all the member predictions $\widehat{y}_c \in \widehat{\mathbf{y}}(\mathbf{x})$.

All the ensemble methods considered in the experimental sections are *uniform* voting ensembles, that is, all the member classifiers carry the same weight $w_i := 1/|C|$. However, this is not a necessary condition, and all the results in this chapter apply equally to non-uniform voting ensembles.

If we define the ensemble's belief distribution to be the weighted average of the individual ensemble members' belief distributions, that is

$$\widehat{p}(y|\mathbf{x}) := E_C\left[\widehat{p}_c(y|\mathbf{x})\right] = \sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x}), \tag{5.3}$$

then the ensemble prediction of a democratic ensemble is also the Bayes optimal prediction, i.e., the one that minimizes the conditional risk $R(\widehat{y}|\mathbf{x})$ as given in Equation 2.4 on page 10:

$$\begin{aligned} V_{\text{dem}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) &= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x}) l(y', y'')\, dy'' \\ &= \arg\min_{y' \in Y} \int_{y'' \in Y} l(y', y'') \sum_{c=1}^{n} w_c \widehat{p}_c(y''|\mathbf{x})\, dy'' \\ &= \arg\min_{y' \in Y} \int_{y'' \in Y} l(y', y'') \widehat{p}(y''|\mathbf{x})\, dy'' \\ &= \arg\min_{y' \in Y} R(y'|\mathbf{x}). \end{aligned}$$

Most of the voting function presented in Section 2.3 (Equation 2.7 through Equation 2.12) can be re-written in the form specified by Definition 5.1.

- Sum Vote:
  For $Y = \widehat{Y} = \widehat{Y}_c = \mathcal{R}$, $\widehat{p}_c(y|\mathbf{x}) := I(y = \widehat{y}_c(\mathbf{x}))$, and $l = l_2$:

$$\begin{aligned} V_{\text{dem}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) &= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x}) l_2(y', y'')\, dy'' && (5.4) \\ &= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} I(y'' = \widehat{y}_c(\mathbf{x}))(y' - y'')^2\, dy'' && (5.5) \\ &= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c (y' - \widehat{y}_c(\mathbf{x}))^2 && (5.6) \\ &= \sum_{c=1}^{n} w_c \widehat{y}_c(\mathbf{x}) && (5.7) \\ &= V_{\text{sum}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) && (5.8) \end{aligned}$$

  For $Y = \widehat{Y} = \mathcal{R}$, $\widehat{Y}_c = \{P(\mathcal{R})\}$, and $l = l_2$:

$$\begin{aligned} V_{\text{dem}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) &= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x}) l_2(y', y'')\, dy'' && (5.9) \\ &= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x})(y' - y'')^2\, dy'' && (5.10) \\ &= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) y'\, dy' && (5.11) \\ &= V_{\text{sum}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) && (5.12) \end{aligned}$$

- Majority Vote:
  For $Y = \widehat{Y} = \widehat{Y}_c = \{y_1, \ldots, y_k\}$, $\widehat{p}_c(y|\mathbf{x}) := I(y = \widehat{y}_c(\mathbf{x}))$, and $l = l_{01}$:

$$\begin{aligned}
V_{\text{dem}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) &= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x}) l_{01}(y', y'')\, dy'' && (5.13)\\
&= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} I(y'' = \widehat{y}_c(\mathbf{x})) I(y' \neq y'')\, dy'' && (5.14)\\
&= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c I(\widehat{y}_c(\mathbf{x}) \neq y') && (5.15)\\
&= \arg\max_{y' \in Y} \sum_{c=1}^{n} w_c I(\widehat{y}_c(\mathbf{x}) = y') && (5.16)\\
&= V_{\text{maj}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) && (5.17)
\end{aligned}$$

- Probabilistic Voting:
  For $Y = \widehat{Y} = \{y_1, \ldots, y_k\}$, $\widehat{Y}_c = \{P(Y)\}$, and $l = l_{01}$:

$$\begin{aligned}
V_{\text{dem}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) &= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x}) l_{01}(y', y'')\, dy'' && (5.18)\\
&= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c \int_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x}) I(y' \neq y'')\, dy'' && (5.19)\\
&= \arg\min_{y' \in Y} \sum_{c=1}^{n} w_c (1 - \widehat{p}_c(y'|\mathbf{x})) && (5.20)\\
&= \arg\max_{y' \in Y} \sum_{c=1}^{n} w_c \widehat{p}_c(y'|\mathbf{x}) && (5.21)\\
&= \arg\max_{y' \in Y} \sum_{c=1}^{n} w_c I(\arg\max_{y'' \in Y} \widehat{p}_c(y''|\mathbf{x}) = y') && (5.22)\\
&= V_{\text{prob}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w}) && (5.23)
\end{aligned}$$

There exist also voting functions which are not democratic according to Definition 5.1. Examples are the majority vote with distribution classifiers (Equation 2.10), aristocratic voting schemes (dictatorships), or progressive voting schemes such as the progressive quadratic vote or the progressive exponential vote ([6]). Definition 5.1 could be extended to cover those voting schemes as well, by introducing intermediate functions $v(\widehat{p}_c(y|\mathbf{x}))$ which modify the ensemble members' belief distributions $\widehat{p}_c(y|\mathbf{x})$ prior to applying the current Definition 5.1.

5.2 Decomposition

The generalization error or loss of an ensemble for a given instance $\langle \mathbf{x}, y \rangle$ will, by definition, always equal the loss of the ensemble prediction relative to the true value. To avoid confusion, we will denote this ensemble loss by $L(\mathbf{x}, y)$ – as opposed

to $\overline{L}(\mathbf{x}, y)$, which will be called mean member loss and used to denote the average expected loss of the individual ensemble members. We then propose to measure the diversity among the predictions of the member classifiers by measuring their loss relative to the ensemble prediction.

Definition 5.2. *The* loss *of ensemble* $C = \langle n, \mathbf{c}, \mathbf{w}, V \rangle$ *on instance* $\langle \mathbf{x}, y \rangle$ *under loss function* l *is given by*

$$L(\mathbf{x}, y) := l(\widehat{y}, y). \tag{5.24}$$

This simply states that the loss of an ensemble is the difference between the ensemble prediction (computed through the voting function V as defined in Definition 2.5) and the true value, as measured by the loss function.

From the standard definitions of accuracy and error for different loss functions used e.g. in [16, 39, 52] we derived the following generalized definition:

Definition 5.3. *The* mean member loss *of ensemble* $C = \langle n, \mathbf{c}, \mathbf{w}, V \rangle$ *on instance* $\langle \mathbf{x}, y \rangle$ *under loss function* l *is given by*

$$\overline{L}(\mathbf{x}, y) := E_C \left[E_{\widehat{P}_c(Y|\mathbf{x})} \left[l(\widehat{y}_c(\mathbf{x}), y) \right] \right] = \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', y) \, dy'. \tag{5.25}$$

The mean member loss of an ensemble indicates how much the predictions of the individual ensemble members differ from the true outcome y. The expectation is taken over the set of ensemble members, according to the weight distribution $\mathbf{w}$.

This quantifies formally the notion of an ensemble whose members are accurate: The ensemble members are more (less) accurate if and only if the ensemble has a lower (higher) mean member loss. We are now ready to introduce the general definition of ensemble diversity, which is applicable to any given loss function.

Definition 5.4. *The* diversity *of ensemble* $C = \langle n, \mathbf{c}, \mathbf{w}, V \rangle$ *on input* $\mathbf{x}$ *under loss function* l *is given by*

$$\overline{D}(\mathbf{x}) := E_C \left[E_{\widehat{P}_c(Y|\mathbf{x})} \left[l(\widehat{y}_c(\mathbf{x}), \widehat{y}(\mathbf{x})) \right] \right] = \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', \widehat{y}(\mathbf{x})) \, dy'. \tag{5.26}$$

In words, the diversity is the weighted average loss incurred by the predictions of the member models $\widehat{y}_c(\mathbf{x})$ relative to the ensemble prediction $\widehat{y}(\mathbf{x})$. Note that the diversity is independent of the true outcome y.

The definitions for loss $L(\mathbf{x}, y)$, mean member loss $\overline{L}(\mathbf{x}, y)$, and diversity $\overline{D}(\mathbf{x})$ can be averaged over the instance distribution $P(\mathbf{X}, Y)$ to obtain the expected performance measures for the ensemble on the problem domain $\langle \mathbf{X}, Y, P, l \rangle$, which we will refer to as

- expected ensemble loss:

$$L := E_{P(\mathbf{X},Y)} \left[L(\mathbf{x}, y) \right] = \int_{\mathbf{X} \times Y} L(\mathbf{x}, y) p(\mathbf{x}, y) \, d\langle \mathbf{x}, y \rangle \tag{5.27}$$

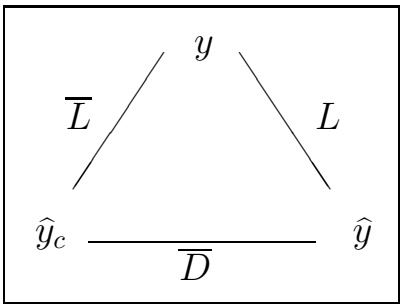

Figure 5.1: Intuitive relations between L, $\overline{L}$, $\overline{D}$ and y, $\widehat{y}$ and $\widehat{y}_c$.

- expected mean member loss:

$$\overline{L} := E_{P(\mathbf{X},Y)}\left[\overline{L}(\mathbf{x},y)\right] = \int_{\mathbf{X}\times Y} \overline{L}(\mathbf{x},y)p(\mathbf{x},y)\, d\langle \mathbf{x},y\rangle \tag{5.28}$$

- expected diversity:

$$\overline{D} := E_{P(\mathbf{X})}\left[\overline{D}(\mathbf{x})\right] = \int_{\mathbf{X}} \overline{D}(\mathbf{x})p(\mathbf{x})\, d\mathbf{x}. \tag{5.29}$$

Figure 5.1 shows the relations between the various terms just defined. It shows that the expected loss L is a function of the ensemble predictions $\widehat{y}$ and the true outcomes y, the expected mean member loss $\overline{L}$ is a function of y and the various base classifiers predictions $\widehat{y}_c$, and the expected diversity $\overline{D}$ is a function of the various base classifiers predictions $\widehat{y}_c$ and the ensemble predictions $\widehat{y}$.

We propose to write the loss $L(\mathbf{x},y)$ for a given ensemble as a function of the loss and the diversity of its members:

$$L(\mathbf{x},y) = f_l(\overline{L}(\mathbf{x},y),\overline{D}(\mathbf{x})) \tag{5.30}$$

The specific form of the functional dependency f_l will depend on the loss function l being used. We proceed by giving f_l for several commonly used loss functions. Usually one is interested in the expected performance of an ensemble over the whole domain, so we will also give the formulas for L obtained by integrating Equation 5.30 over the instance distribution.

5.3 Instantiating the Decomposition for Squared Loss

We will first show that the commonly used standard decomposition for democratic voting ensembles under squared loss (Equation 3.4) is a special case of the unified decomposition as presented in Equation 5.30. In Section 5.4 we will then instantiate the unified decomposition under 0-1 loss for two-class and multi-class problems.

The following theorem states that the commonly used decomposition for squared loss is indeed a special case of our general decomposition:

Theorem 5.1. *Let $C := \langle n, \mathbf{c}, \mathbf{w}, V \rangle$ be an ensemble of classifiers such that $Y = \widehat{Y} = \mathcal{R}$, $\widehat{Y}_c = \{P(\mathcal{R})\}$ for all $c \in \{1, \ldots, n\}$, and $V = V_{\text{dem}}$. Then, for squared loss, the ensemble loss $L(\mathbf{x}, y)$ can be written as*

$$L(\mathbf{x}, y) = f_l(\overline{L}(\mathbf{x}, y), \overline{D}(\mathbf{x})) = \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}). \tag{5.31}$$

The proof of Theorem 5.1 can be found in Appendix G on page 137.

Theorem 5.1 actually covers the more general case where the individual member classifiers are distribution classifiers that each output a probability distribution over $\mathcal{R}$. The classic regression model with value classifiers is obtained as a special case by setting $\widehat{p}_c(y'|\mathbf{x}) := I(y' = \widehat{y}_c(\mathbf{x}))$, as the following theorem shows.

Theorem 5.2. *Let $C := \langle n, \mathbf{c}, \mathbf{w}, V \rangle$ be an ensemble of classifiers such that $Y = \widehat{Y} = \mathcal{R}$, $\widehat{Y}_c = \mathcal{R}$ for all $c \in \{1, \ldots, n\}$, and $V = V_{\text{dem}}$. Then, for squared loss, the ensemble loss $L(\mathbf{x}, y)$ can be written as*

$$L(\mathbf{x}, y) = f_l(\overline{L}(\mathbf{x}, y), \overline{D}(\mathbf{x})) = \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}). \tag{5.32}$$

The proof of Theorem 5.2 can be found in Appendix G on page 139.

Under squared loss, the additivity of the loss decomposition is also preserved when integrating over the instance space:

Theorem 5.3. *For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and squared loss, the expected ensemble loss over the domain is*

$$L = \overline{L} - \overline{D}. \tag{5.33}$$

The proof of Theorem 5.3 can be found in Appendix G on page 140.

Theorem 5.1 through Theorem 5.3 are significant for two reasons. Firstly, they show that the decomposition commonly used for real-valued classes under squared loss is a special case of our general decomposition framework presented in Section 5.2. Secondly, they extend this decomposition to the case where the base classifiers, rather than producing a single real number as their prediction, may output a probability distribution over $\mathcal{R}$ instead.

5.4 Instantiating the Decomposition for 0-1 Loss

We now instantiate Equation 5.30 for 0-1 loss as the loss function, first for the case where the outcome space Y consists of two classes. In Section 5.4.2 we will generalize this to multi-class problems.

5.4.1 0-1 Loss for Two-Class Problems

The following theorem shows the relationship between ensemble loss $L(\mathbf{x}, y)$, mean member loss $\overline{L}(\mathbf{x}, y)$, and diversity $\overline{D}(\mathbf{x})$ for a single given instance $\langle \mathbf{x}, y \rangle$.

Theorem 5.4. *For ensembles* $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ *and 0-1 loss in two-class problems, the ensemble loss* $L(\mathbf{x}, y)$ *can be written as*

$$L(\mathbf{x}, y) = f_l(\overline{L}(\mathbf{x}, y), \overline{D}(\mathbf{x})) = \overline{L}(\mathbf{x}, y) + z\overline{D}(\mathbf{x}) \tag{5.34}$$

with $z = -1$ *iff* $\widehat{y}(\mathbf{x}) = y$*, and* $z = 1$ *iff* $\widehat{y}(\mathbf{x}) \neq y$*.*

The proof of Theorem 5.4 can be found in Appendix G on page 141.

To find an expression of the expected ensemble loss L over the entire domain, we need to distinguish between the expected diversity of the ensemble over T and F.

Definition 5.5. *Let* $Y := Y_1, \ldots, Y_k$ *be a discrete outcome space, and let* l *be the 0-1 loss function. Given any ensemble* C*, let* $T := \{\langle \mathbf{x}, y \rangle \in \mathbf{X} \times Y | \widehat{y}(\mathbf{x}) = y\}$ *denote the set of instances that* C *predicts correctly, and let* $F := \{\langle \mathbf{x}, y \rangle \in \mathbf{X} \times Y | \widehat{y}(\mathbf{x}) \neq y\}$ *denote the set of instances that* C *predicts incorrectly. Then, the* expected diversity over correctly predicted instances *is defined as*

$$\overline{D}_T := \int_{\langle \mathbf{x}, y \rangle \in T} \overline{D}(\mathbf{x}) p(\langle \mathbf{x}, y \rangle | \langle \mathbf{x}, y \rangle \in T)\, d\langle \mathbf{x}, y \rangle, \tag{5.35}$$

and the expected diversity over incorrectly predicted instances *is defined as*

$$\overline{D}_F := \int_{\langle \mathbf{x}, y \rangle \in F} \overline{D}(\mathbf{x}) p(\langle \mathbf{x}, y \rangle | \langle \mathbf{x}, y \rangle \in F)\, d\langle \mathbf{x}, y \rangle. \tag{5.36}$$

Naturally, the expected diversity over the entire instance space $\overline{D}$ can also be expressed in terms of the expected diversities over correctly and incorrectly predicted instances, as the following theorem shows:

Theorem 5.5. *Under 0-1 loss, the expected diversity* $\overline{D}$ *can be written as*

$$\overline{D} = (1 - L)\overline{D}_T + L\overline{D}_F. \tag{5.37}$$

The proof of Theorem 5.5 can be found in Appendix G on page 142.

For democratic ensembles, $\overline{D}_T$, $\overline{D}_F$, and $\overline{D}$ will always be between 0 and 0.5 for binary classes, and between 0 and $1 - 1/|Y|$ in general. This can be easily seen for a two class dataset. A value of $\overline{D} > 0.5$ would mean that, for some instances, more than half of the individual member classifiers votes is given to a label different from the one the ensemble is predicting, and by definition the ensemble would have to change its prediction for those instances, making $\overline{D}$ less than 0.5 again. The same argument holds for $\overline{D}_F$ and $\overline{D}_T$ and becomes even more obvious when there are more than two classes.

The following two theorems express the relationships between expected ensemble loss, expected mean member loss, and expected diversity over the entire instance space:

Theorem 5.6. *For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and 0-1 loss in two-class problems, the expected ensemble loss over the domain can be written as*

$$L = \frac{\overline{L} - \overline{D}}{1 - 2\overline{D}_F}. \tag{5.38}$$

Theorem 5.7. *For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and 0-1 loss in two-class problems, the expected ensemble loss over the domain can be written as*

$$L = \frac{\overline{L} - \overline{D}_T}{1 - \overline{D}_T - \overline{D}_F}. \tag{5.39}$$

The proofs of Theorems 5.6 and 5.7 can be found in Appendix G on pages 143 and 144, respectively. Alternatively, Theorem 5.7 can also be derived from Theorem 5.6, and vice versa, via substitution of one the variables $\overline{D}$ or $\overline{D}_T$ by the corresponding expression derived from Theorem 5.5.

Equation 5.39 appears to be rather counter-intuitive: On first sight it would seem that increasing $\overline{D}_F$ (the diversity on incorrectly predicted examples) would actually increase the overall expected ensemble loss L, rather than decrease it. However, one has to take into account that one cannot change $\overline{D}_F$ without also changing the mean member loss $\overline{L}$ at the same time. If, for example, we are to change the ensemble in some way resulting in an increase of $\overline{D}_F$ by α, this change will also result in a decrease of $\overline{L}$ by $L\alpha$.

This is markedly different from the behavior under squared loss, where it is (in principle) always possible to increase diversity while leaving the mean member loss constant, resulting in better ensemble performance. This difference in behavior, however, is a consequence only of the different properties of the loss functions. Our consistent definition of diversity and accuracy/mean member loss makes explicit this qualitative difference in ensemble behavior. In Section 5.4.3 we will show that increasing $\overline{D}_F$ will indeed decrease the expected ensemble loss.

5.4.2 0-1 Loss for Multi-Class Problems

We now instantiate the unified decomposition for 0-1 loss for multi-class problems. The proofs follow closely the two-class case; the main difference being that now, for those examples that the ensemble predicts incorrectly, increasing diversity does not automatically entail lowering mean member loss: $[\widehat{y}(\mathbf{x}) \neq y \wedge \widehat{y}_c(\mathbf{x}) \neq \widehat{y}(\mathbf{x})] \not\Rightarrow \widehat{y}_c(\mathbf{x}) = y$.

Theorem 5.8. *For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and 0-1 loss, the ensemble loss $L(\mathbf{x}, y)$ can be written as*

$$L(\mathbf{x}, y) = f_l(\overline{L}(\mathbf{x}, y), \overline{D}(\mathbf{x})) = \overline{L}(\mathbf{x}, y) + z\overline{D}(\mathbf{x}), \tag{5.40}$$

with $z = -1$ iff $\widehat{y}(\mathbf{x}) = y$, and $z = \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|x)}$ iff $\widehat{y}(\mathbf{x}) \neq y$.

The proof of this theorem can be found in Appendix G on page 145. Note that Theorem 5.4 is a special case of Theorem 5.8 with $z = \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1-\sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|x)} = 1$ for the case of $\widehat{y}(\mathbf{x}) \neq y$, since, for two-class problems, $\widehat{y}(\mathbf{x}) \neq y$ implies that $\widehat{p}_c(y|\mathbf{x}) = 1 - \widehat{p}_c(\widehat{y}(\mathbf{x})|x)$ for all ensemble members c.

As for two-class problems, an expression of the expected ensemble loss over the domain L can be obtained by integrating $L(\mathbf{x}, y)$ over the instance distribution. Let $\overline{D}_T$ and $\overline{D}_F$ denote the expected diversities on correctly and incorrectly predicted examples, respectively, as in Definition 5.5. Furthermore, let

$$\overline{D}_P := \int_{\langle \mathbf{x},y\rangle \in F} (1 - \overline{L}(\mathbf{x}, y)) p(\langle \mathbf{x}, y\rangle | \langle \mathbf{x}, y\rangle \in F)\, d\langle \mathbf{x}, y\rangle \tag{5.41}$$

denote the average weight that the individual ensemble members assign to the correct outcome, given that the ensemble as a whole predicts the same instance incorrectly. Then, the following theorems hold:

Theorem 5.9. *For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}}\rangle$ and 0-1 loss, the expected ensemble loss over the domain can be written as*

$$L = \frac{\overline{L} - \overline{D}}{1 - \overline{D}_P - \overline{D}_F}. \tag{5.42}$$

The proof of this theorem can be found in Appendix G on page 147.

The expression for the expected ensemble loss in Theorem 5.9 depends on four variables, one of which we can eliminate by substituting the expected diversity $\overline{D}$ in Equation 5.42 by its expression from Theorem 5.5:

Theorem 5.10. *For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}}\rangle$ and 0-1 loss, the expected ensemble loss over the domain can be written as*

$$L = \frac{\overline{L} - \overline{D}_T}{1 - \overline{D}_T - \overline{D}_P}. \tag{5.43}$$

The proof of this theorem can be found in Appendix G on page 148.

From Theorem 5.10 follows that for multi-class problems, on those instances that the ensemble predicts incorrectly, not all diversity contributes to reducing the ensemble loss. Out of all member classifiers that disagree with the ensemble prediction, only some will actually predict y correctly, and only those contribute to reducing the expected loss. This has an important consequence: As the number of classes increases, one can expect the optimal accuracy-diversity trade-off to shift towards ensembles with members that are more accurate but less diverse, all other things being equal.

5.4.3 Discussion

Naturally, the two-class case ($Y = \{Y_1, Y_2\}$) discussed in Section 5.4.1 is a special case of the multi-class case ($Y = \{Y_1, \ldots, Y_k\}$) discussed in Section 5.4.2. As such,

Theorem 5.4 is a special case of Theorem 5.8, Theorem 5.6 is a special case of Theorem 5.9, and Theorem 5.7 is a special case of Theorem 5.10. What makes the two-class case special is that, on those instances that the ensemble predicts erroneously, a member prediction that differs from the ensemble prediction entails the member prediction being a correct one. Consequentially, for the two-class case, it holds $\overline{D}_P = \overline{D}_F$, that is, the expected weight of member classifier votes assigned to labels different from the ensemble prediction equals the expected weight of member classifier votes assigned to the correct label, where the expectation is taken in both cases over those instances that the ensemble predicts erroneously.

Given a prediction problem, there are basically two ways to manipulate ensemble diversity:

1. by keeping T and F constant. In this case, the ensemble decisions do not change, hence the 0-1 loss does not change, but the diversities $\overline{D}_T$, $\overline{D}_F$, and $\overline{D}_P$ can be manipulated independently.

2. by changing the boundaries of T and F. In this case, the 0-1 loss is affected, and so are $\overline{D}_T$, $\overline{D}_F$, and $\overline{D}_P$.

Case 1: When keeping T and F constant and increasing $\overline{D}_T$ by α, $\overline{D}$ increases by $\alpha(1-L)$ (by Theorem 5.5) and so does $\overline{L}$ (because of Equation 5.43 and L remains constant). Theorem 5.9 confirms that L remains constant under such change. For two-class problems, if $\overline{D}_P$ is increased by α, $\overline{D}$ increases by αL (again by Theorem 5.5), and $\overline{L}$ decreases by αL (by Theorem 5.7). Theorem 5.6 confirms that L remains constant under such change. For problems with more than two classes, increasing $\overline{D}_P$ may or may not increase $\overline{D}$, depending on whether or not the member classifiers whose votes are changed agreed with the ensemble prediction before the change. However, $\overline{L}$ will decrease by αL (by Theorem 5.10), and in either case Theorem 5.9 confirms the constancy of L, since $\overline{D}_F$ decreases by exactly the same amount as $\overline{D}$.

Case 2: Now consider manipulating T and F by moving instances from F to T such that L decreases by α. Let M be the set of instances $\{\langle \mathbf{x}, y\rangle \in \mathbf{X} \times Y\}$ that moved from F to T. The contribution of those instances to $\overline{L} - \overline{D}$ is 0 by Theorem 5.8. Hence, by Theorem 5.9, we have $L - \alpha = (\overline{L} - \overline{D})/(1 - \overline{D}_P - \overline{D}_F)$, which after some manipulations gives $\overline{D}_P = (1 - \overline{D}_F) - (\overline{L} - \overline{D})/(L - \alpha)$, that is, $\overline{D}_P$ increases. This also works the other way around; increasing $\overline{D}_P$ helps decrease the expected ensemble loss L.

Let us look at a case study to make this a bit clearer. In Figure 5.2 we have three instances in the set T and two instances in the set F. There are three member classifiers for classifying each of these instances; their votes are given beside each instance. In case 1 we increase $\overline{D}_T$ while keeping T and F constant; for example changing the votes at instance 1 from TTT to TTF. This will cause an increase in $\overline{D}$ and $\overline{L}$ but L remains constant. Likewise, changing the votes for instance 4 from FFF to FTF will cause an increase in $\overline{D}_F$ and $\overline{D}$, but a decrease in $\overline{L}$, while

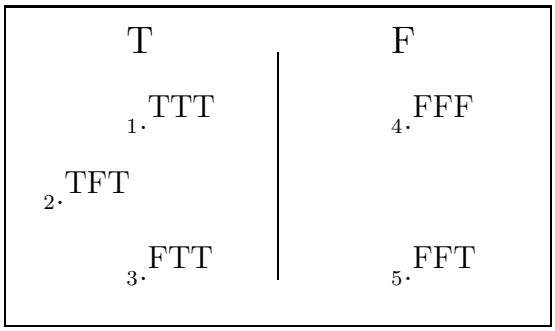

Figure 5.2: Case study of relations between L, $\overline{L}$, and $\overline{D}$.

L remains constant. This highlights that just increasing or decreasing diversity can have no effect on the benefits derived from an ensemble.

Now let us examine case 2. If we move instance 3 from the T set to the F set by changing its votes from FTT to FTF, we will cause L to increase while $\overline{D}$ hasn't changed. $\overline{L}$ will also increase, and $\overline{D}_F$ could either have increased or decreased. If we move instance 5 from the F set to the T set by changing its votes from FFT to FTT, we will cause L to decrease while $\overline{D}$ still hasn't changed. $\overline{L}$ will also decrease, and $\overline{D}_T$ could either have increased or decreased. From this we can see, that increasing the diversity over the F set is the only way to lower the ensemble loss.

The deciding influence of $\overline{D}_T$ and $\overline{D}_P$ becomes even more evident when we assume the availability of a family of base classifiers with some given, fixed mean member loss $\overline{L}$.

Consider the following example with three classes ($Y = \{a, b, c\}$), six instances ($\mathbf{X} \times Y = \{\langle \mathbf{x}_1, y_1 \rangle, \ldots, \langle \mathbf{x}_6, y_6 \rangle\}$), and a set of four classifiers $C = \{c_1, c_2, c_3, c_4\}$, each classifier being able to classify three instances correctly and three incorrectly ($\overline{L} = 0.5$). For simplicity, a democratic voting scheme is used ($V = V_{\text{maj}}$ as in Equation 2.9), and the weights are assumed to be uniform ($w_c = 1/5$ for all $c \in C$).

If all of the classifiers' predictions are exactly the same, that is, if there is no diversity, we get a situation as shown in Table 5.1. Only three instances will be classified correctly, and no gain over the individual base classifiers is made.

By increasing the diversity we can achieve a situation where all of the instances are predicted correctly (Table 5.2), although any one of the member classifiers still predicts three out of the six instances incorrectly.

However, Table 5.3 and Table 5.4 show examples where the diversity is also increased (compared to Table 5.1), but the ensemble loss stays the same (Table 5.3) or even increases (Table 5.4). Note that the ensembles shown in Table 5.3 and Table 5.4 have exactly the same mean member loss $\overline{L}$ and diversity $\overline{D}$, they only differ in the instances on which the diversity occurs ($\overline{D}_T$ and $\overline{D}_F$).

Probabilistic voting can be interpreted as majority voting where, instead of a single vote by a single classifier c_i, a set of classifiers $c_{i,1}, \ldots, c_{i,|Y|}$ vote democratically, and

$\mathbf{x}$	y	c_1	c_2	c_3	c_4	Ensemble	$L(\mathbf{x}, y)$	$\overline{L}(\mathbf{x}, y)$	$\overline{D}(\mathbf{x})$
$\mathbf{x}_1$	a	a	a	a	a	a	0	0/4	0/4
$\mathbf{x}_2$	b	b	b	b	b	b	0	0/4	0/4
$\mathbf{x}_3$	c	c	c	c	c	c	0	0/4	0/4
$\mathbf{x}_4$	a	b	b	b	b	b	1	4/4	0/4
$\mathbf{x}_5$	b	c	c	c	c	c	1	4/4	0/4
$\mathbf{x}_6$	c	a	a	a	a	a	1	4/4	0/4

$L = 3/6$
$\overline{L} = 12/24$
$\overline{D} = 0/24$
$\overline{D}_T = 0/12$
$\overline{D}_F = 0/12$
$\overline{D}_P = 0/12$

Table 5.1: Example where there is no diversity ($\overline{D} = 0$).

$\mathbf{x}$	y	c_1	c_2	c_3	c_4	Ensemble	$L(\mathbf{x}, y)$	$\overline{L}(\mathbf{x}, y)$	$\overline{D}(\mathbf{x})$
$\mathbf{x}_1$	a	b	c	a	a	a	0	2/4	2/4
$\mathbf{x}_2$	b	b	c	a	b	b	0	2/4	2/4
$\mathbf{x}_3$	c	c	a	b	c	c	0	2/4	2/4
$\mathbf{x}_4$	a	a	a	b	c	a	0	2/4	2/4
$\mathbf{x}_5$	b	c	b	b	a	c	0	2/4	2/4
$\mathbf{x}_6$	c	a	c	c	b	c	0	2/4	2/4

$L = 0/6$
$\overline{L} = 12/24$
$\overline{D} = 12/24$
$\overline{D}_T = 12/24$
$\overline{D}_F = n/d$
$\overline{D}_P = n/d$

Table 5.2: Example with high diversity and perfect classification.

$\mathbf{x}$	y	c_1	c_2	c_3	c_4	Ensemble	$L(\mathbf{x}, y)$	$\overline{L}(\mathbf{x}, y)$	$\overline{D}(\mathbf{x})$
$\mathbf{x}_1$	a	b	a	a	a	a	0	1/4	1/4
$\mathbf{x}_2$	b	a	b	b	b	b	0	1/4	1/4
$\mathbf{x}_3$	c	c	c	a	c	c	0	1/4	1/4
$\mathbf{x}_4$	a	a	b	b	b	b	1	3/4	1/4
$\mathbf{x}_5$	b	b	c	c	c	c	1	3/4	1/4
$\mathbf{x}_6$	c	a	a	c	a	a	1	3/4	1/4

$L = 3/6$
$\overline{L} = 12/24$
$\overline{D} = 6/24$
$\overline{D}_T = 3/12$
$\overline{D}_F = 3/12$
$\overline{D}_P = 3/12$

Table 5.3: Example with high diversity but no performance gain.

$\mathbf{x}$	y	c_1	c_2	c_3	c_4	Ensemble	$L(\mathbf{x}, y)$	$\overline{L}(\mathbf{x}, y)$	$\overline{D}(\mathbf{x})$
$\mathbf{x}_1$	a	a	a	a	a	a	0	0/4	0/4
$\mathbf{x}_2$	b	b	b	b	b	b	0	0/4	0/4
$\mathbf{x}_3$	c	b	b	a	c	b	1	3/4	2/4
$\mathbf{x}_4$	a	c	a	b	b	b	1	3/4	2/4
$\mathbf{x}_5$	b	b	c	c	c	c	1	3/4	1/4
$\mathbf{x}_6$	c	a	a	c	a	a	1	3/4	1/4

$L = 4/6$
$\overline{L} = 12/24$
$\overline{D} = 6/24$
$\overline{D}_T = 0/8$
$\overline{D}_F = 6/16$
$\overline{D}_P = 4/16$

Table 5.4: Example with high diversity and performance loss.

a portion $\widehat{p}_c(y|\mathbf{x}) = Y_i)$ of those votes for $y = Y_i$ while the rest votes for $y \neq Y_i$. Therefore, the same effect can be observed in probabilistically voting ensembles.

5.5 General Bounds on the Expected Ensemble Loss

Using only some weak assumptions about the characteristics of the loss function in use, we can derive upper and lower bounds on the ensemble loss in terms of mean member loss and diversity. The assumptions we have to make are the ones of *triangularity* and *symmetry*:

Definition 5.6. *A loss function $l : Y \times Y \to \mathcal{R}$ is called* triangular *iff it obeys the triangle inequality*

$$\forall a, b, c \in Y : l(a, b) + l(b, c) \geq l(a, c). \tag{5.44}$$

Definition 5.7. *A loss function $l : Y \times Y \to \mathcal{R}$ is called* symmetric *iff*

$$\forall a, b \in Y : l(a, b) = l(b, a). \tag{5.45}$$

Many commonly used loss functions are both triangular and symmetric, including e.g. 0-1 loss, squared loss, absolute loss, etc.

Theorem 5.11. *Let l be any triangular loss function. Then, for any given democratic ensemble,*

$$L(\mathbf{x}, y) \geq \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}) \tag{5.46}$$

and

$$L \geq \overline{L} - \overline{D}. \tag{5.47}$$

Theorem 5.12. *Let l be any loss function which is both triangular and symmetric. Then, for any given democratic ensemble,*

$$L(\mathbf{x}, y) \leq \overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x}) \tag{5.48}$$

and

$$L \leq \overline{L} + \overline{D}. \tag{5.49}$$

The proofs of Theorems 5.11 and 5.12 can be found in Appendix G on pages 149 and 150, respectively.

Using Theorem 5.11 and Theorem 5.12, we can thus bound the ensemble loss in terms of the mean member loss and the diversity, both from above and from below, for any triangular and symmetric loss function.

6. Applying the Loss Decomposition

While the true quantities L, $\overline{L}$, $\overline{D}$, etc. are defined in terms of expectations with respect to the (unknown) joint distribution of $\mathbf{X} \times Y$ and therefore as such are unknown, they can be easily estimated by measuring them on a given sample drawn from $\mathbf{X} \times Y$, as is standard practice for the ensemble loss L. In order to empirically verify the theorems derived in Chapter 5 as well as to illustrate their practical use, we estimated all the components of the loss decomposition for each of the 36 previously used UCI datasets using 10 runs of 10-fold cross-validation, as described in Section 4.1. For each of the 100 training/testing splits per data set, we learned an ensemble consisting of 30 unpruned decision trees on the training set. We then evaluated the ensemble on the test set, measuring expected ensemble loss L, expected mean member loss $\overline{L}$, and expected diversity $\overline{D}$, as defined in Section 5.2. We further measured expected diversity on correctly predicted examples $\overline{D}_T$, expected diversity on incorrectly predicted examples $\overline{D}_F$, and expected probability of predicting the true class given that the ensemble made a mistake $\overline{D}_P$. The averages and standard deviations of those measurements are shown in Appendix H, with one table for each ensemble method.

When plugging those averages into the formulas given in Section 5.4.1 and Section 5.4.2, the left hand side does usually not equal the right hand side. This is not a flaw of the formulas however, but due only to the averaging over the 100 training/testing splits: in general, $\text{avg}(a_1, a_2..., a_n)/\text{avg}(b_1, b_2..., b_n)$ does not equal $\text{avg}(a_1/b_1, a_2/b_2, ..., a_n/b_n)$. When plugging the estimates into the formulas for each individual training/testing split, the left hand side does indeed equal the right hand side.

In Appendix H, Column L^* shows the results of computing the expected ensemble loss that are obtained by plugging the *averaged* estimates for $\overline{L}$, $\overline{D}_T$, and $\overline{D}_P$ into Equation 5.43 (for a small number of training/testing splits, $\overline{D}_F$ and $\overline{D}_P$ were unmeasurable because the ensemble made no mistakes on the test data – for the calculation of the averages of L^*, we set $\overline{D}_P = \overline{D}_F = 0$ for those training/testing splits). When, on the other hand, the right hand side of Equation 5.43 is computed directly from the estimates for each individual training/testing split and the results averaged afterwards, the obtained values do indeed match those obtained by averaging the measured ensemble loss estimates, which are shown in column L.

Dataset	L		L^*	
	V_{prob}	V_{maj}	V_{prob}	V_{maj}
anneal	0.39±0.20	0.39±0.20	0.33±0.20	0.33±0.20
audiology	**17.56**±1.61	17.74±1.52	**17.45**±1.61	17.63±1.52
autos	**16.02**±1.13	16.07±1.12	**16.00**±1.13	16.07±1.12
balance	19.96±0.50	19.96±0.50	**19.97**±0.50	19.97±0.50
breastc	31.79±1.87	**31.58**±1.80	31.95±1.87	**31.75**±1.80
breastw	**3.98**±0.21	4.00±0.20	**3.88**±0.21	3.91±0.20
colic	14.70±0.44	**14.67**±0.56	14.72±0.44	**14.66**±0.56
credita	**14.46**±0.53	14.54±0.60	**14.46**±0.53	14.54±0.60
creditg	25.44±1.09	25.44±1.09	25.52±1.09	**25.52**±1.09
diabetes	24.09±1.20	**23.96**±1.07	24.10±1.20	**23.96**±1.07
glass	25.09±1.46	**25.05**±1.44	25.23±1.46	**25.18**±1.44
heartc	**19.40**±1.20	19.49±1.15	**19.41**±1.20	19.50±1.15
hearth	22.69±0.90	**22.49**±0.81	22.83±0.90	**22.60**±0.81
hearts	63.70±2.94	**63.68**±2.82	**63.59**±2.94	63.61±2.82
heartv	**20.04**±0.93	20.07±0.92	**19.93**±0.93	19.95±0.92
hepatitis	**18.11**±1.36	18.23±1.65	18.20±1.36	**18.06**±1.65
hypo	**0.33**±0.07	0.34±0.06	**0.31**±0.07	0.32±0.06
ionosphere	**7.44**±0.59	7.50±0.57	**7.23**±0.59	7.27±0.57
iris	**5.47**±0.88	5.67±0.79	**5.09**±0.88	5.27±0.79
krk	**16.16**±0.18	16.18±0.18	**16.16**±0.18	16.18±0.18
krkp	0.39±0.07	0.39±0.07	0.35±0.07	0.35±0.07
labor	15.27±1.73	**15.10**±1.73	13.52±1.73	**13.51**±1.73
letter	**5.74**±0.12	5.74±0.12	**5.74**±0.12	5.74±0.12
lymph	17.65±1.95	17.65±1.95	17.54±1.95	17.54±1.95
phoneme	**9.96**±0.19	10.17±0.16	**9.96**±0.19	10.17±0.16
primary	**60.68**±1.29	60.77±1.58	**60.73**±1.29	60.83±1.58
satimage	9.19±0.17	**9.17**±0.15	9.19±0.17	**9.16**±0.15
segment	**2.21**±0.18	2.26±0.18	**2.19**±0.18	2.24±0.18
shuttle	**0.02**±0.00	0.02±0.00	**0.02**±0.00	0.02±0.00
sick	**1.13**±0.11	1.15±0.13	**1.11**±0.11	1.13±0.13
sonar	**20.33**±2.57	20.90±2.52	**20.38**±2.57	20.98±2.52
soybean	8.05±0.49	**7.99**±0.46	8.04±0.49	**7.98**±0.46
splice	5.86±0.06	5.86±0.06	5.86±0.06	5.86±0.06
vehicle	**24.74**±0.57	24.85±0.71	**24.68**±0.57	24.80±0.71
voting	4.25±0.38	**4.23**±0.36	**4.05**±0.38	4.09±0.36
waveform	17.12±0.35	**17.10**±0.37	17.13±0.35	**17.11**±0.37
Mean	8.16	8.19	8.00	8.03

Table 6.1: Loss comparison of *Bagging*$(1;30)$ with Probabilistic vs. Majority Vote.

	L	$\overline{L}$	$\overline{D}$	$\overline{D}_T$	$\overline{D}_F$	$\overline{D}_P$	L^*
V_{prob}	8.16	11.68	7.74	6.17	26.20	20.41	8.00
V_{maj}	8.19	11.50	7.43	5.86	25.67	20.04	8.03

Table 6.2: Loss decomposition components for $Bagging(1;30)$ with Probabilistic vs. Majority Vote.

6.1 Probabilistic vs. Majority Vote

The original Bagging algorithm presented in [8] uses only the predicted outcomes, i.e., the member classifiers are value classifiers as in Definition 2.1, and the ensemble predicts the outcome most frequently predicted by the member classifiers, in accordance with Equation 2.9. This is also the most commonly used version of Bagging. However, many types of classifiers are able to make probabilistic predictions, i.e. to output a probability distribution over the outcome space rather than just a single predicted outcome. To make use of those probabilistic predictions in an ensemble, all that has to be done is a suitable extension to the voting function in order to enable it to combine whole probability distributions rather than single predictions. The extension shown in Equation 2.11 is straightforward: The voting function averages the member classifiers' belief distributions for each outcome and predicts the outcome with the highest average belief. As shown in Section 5.1, both voting functions Equation 2.9 and Equation 2.11 are democratic voting functions as in Definition 5.1, and Equation 2.11 is a generalization of Equation 2.9.

Furthermore, for ensembles whose members are distribution classifiers, each member classifier can be seen as a collection of many value classifiers, out of which a portion $\widehat{p}_c(y|\mathbf{x})$ predicts the corresponding outcome y. Therefore, given that the benefits of using ensembles presumably grow as the number of member classifiers grows, we would expect probabilistic voting to perform better than simple majority vote. To test this hypothesis, we ran the loss decomposition experiments for both $Bagging(1;30)$ with majority vote and $Bagging(1;30)$ with probabilistic voting. The results are shown in Table H.1 (majority vote) and Table H.2 (probabilistic voting) on pages 151–152.

Table 6.1 directly compares the ensemble loss for the two voting functions on a per dataset basis, whereas Table 6.2 shows the averages (in terms of the geometric mean) over all datasets for all variables of the loss decomposition. As expected, probabilistic voting slightly outperforms simple majority vote: it performs slightly better on 17 datasets and slightly worse on 11 datasets, although none of the differences is statistically significant. Bagging with probabilistic voting achieves an average ensemble loss of 8.16 percent, versus an average ensemble loss of 8.19 percent using majority vote.

This reduction in expected ensemble loss is achieved despite a slight increase in the expected member loss $\overline{L}$ and comes with a slight increase in expected diversity $\overline{D}$. (Although in both cases the ensembles were learned and evaluated on the same

	L	$\overline{L}$	$\overline{D}$	$\overline{D}_T$	$\overline{D}_F$	$\overline{D}_P$	L^*
Bagging$(0.5; 30)$	8.73	13.43	9.79	8.03	30.44	23.24	8.55
Bagging$(1; 30)$	8.16	11.68	7.74	6.17	26.20	20.41	8.00
Bagging$(2; 30)$	8.13	10.63	6.07	4.70	21.57	17.05	8.02
Cragging$(2; 15)$	8.37	12.53	8.77	7.14	28.44	21.91	8.21
Cragging$(3; 10)$	8.22	11.50	7.29	5.77	24.02	18.82	8.11
Cragging$(30; 1)$	9.14	10.09	3.16	2.24	11.53	9.33	9.12

Table 6.3: Comparison of sampling schemes.

100 training/testing split and consist of the same decision trees, the definition of expected mean member loss according to Equation 5.25 and Equation 5.28 nevertheless leads to slightly different values of $\overline{L}$.) The small increase in $\overline{D}_P$ is able to make up for the proportionally higher increase in $\overline{D}_T$, resulting overall in the slight reduction of expected ensemble loss.

These experiments confirm empirical observations made in [3]. Apart from the marginally better performance, the added expense of storing and handling the individual member classifiers' belief distributions may also be justified by other considerations, such as when a probability distribution over the outcome space is desired as ensemble output.

6.2 Comparison of Sampling Schemes

We have already seen in Section 4.2 that the "standard" Bagging algorithm can be improved upon frequently by varying the scheme according to which the instances passed to the base classifier inducer are sampled from the original training set. Representatively we consider here varying the resampling rate s (*Bagging*$(0.5; 30)$ with $s := 0.5$, standard *Bagging*$(1; 30)$ with $s := 1$, and *Bagging*$(2; 30)$ with $s := 2$), as well as using cross-validation sampling with r runs and f folds instead of bootstrap sampling (*Cragging*$(30; 1)$ with $r := 1$ and $f := 30$, *Cragging*$(2; 15)$ with $r := 15$ and $f := 2$, and *Cragging*$(3; 10)$ with $r := 10$ and $f := 3$). All schemes generate ensembles with 30 member classifiers and consistently use V_{prob} (Equation 2.11) as the voting function. Appendix H shows the measured values of the loss decomposition components for each tested problem domain. The same data is shown again in Appendix I, organized differently. While Appendix H has one table per sampling scheme, Appendix I contains one table for each loss decomposition component. Thus, the tables in Appendix H are suited for comparisons regarding one sampling scheme across different problem domains, while the tables in Appendix I facilitate comparisons of different sampling schemes against each other. In each row of the tables in Appendix I, the best-performing ensemble method (i.e., the combination of decomposition variables resulting in the lowest expected ensemble loss) is marked in bold.

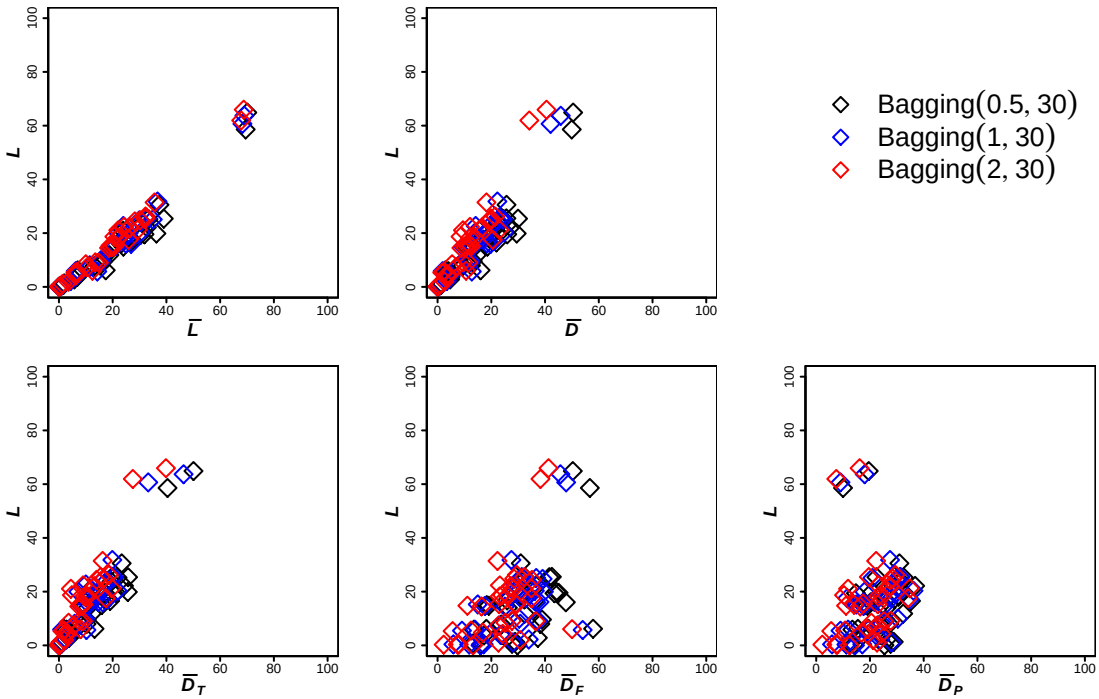

Figure 6.1: Influence of decomposition variables on ensemble loss for *Bagging*$(0.5; 30)$, *Bagging*$(1; 30)$, and *Bagging*$(2; 30)$.

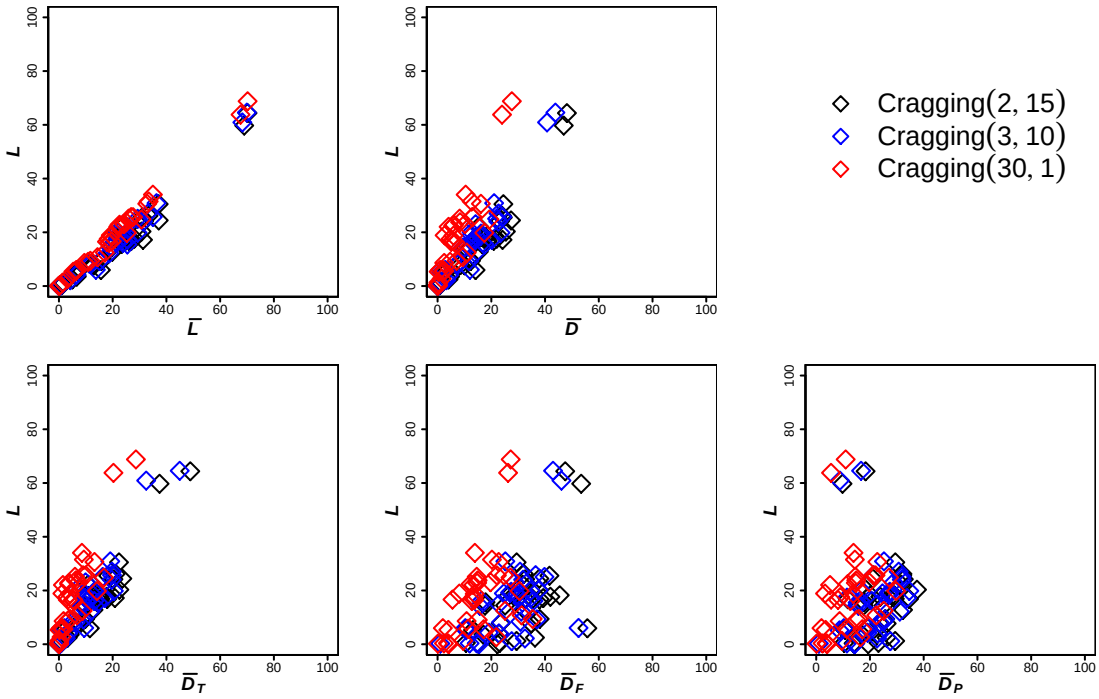

Figure 6.2: Influence of decomposition variables on ensemble loss for *Cragging*$(2; 15)$, *Cragging*$(3; 10)$, and *Cragging*$(30; 1)$.

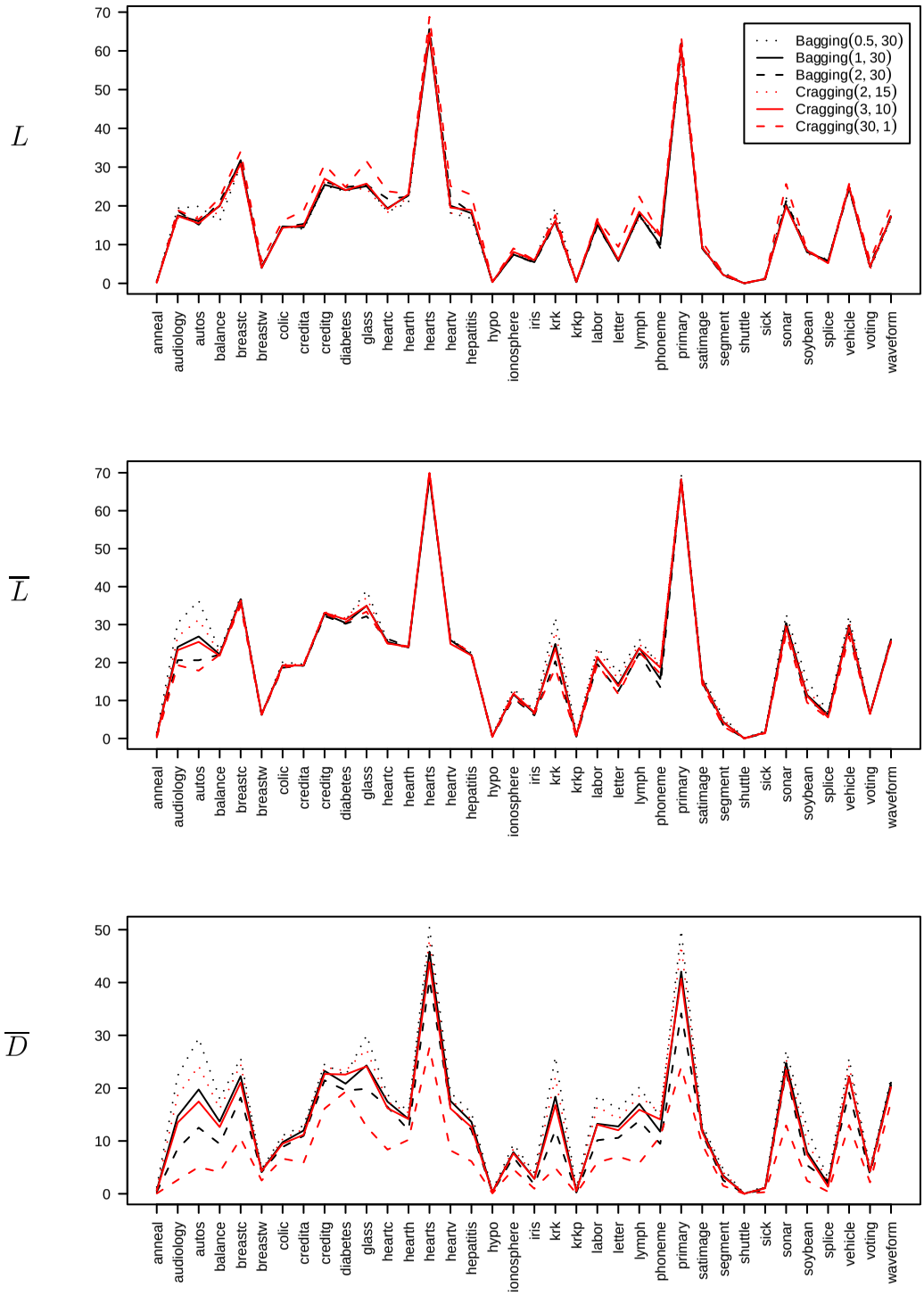

Figure 6.3: Measured values of L, $\overline{L}$, and $\overline{D}$.

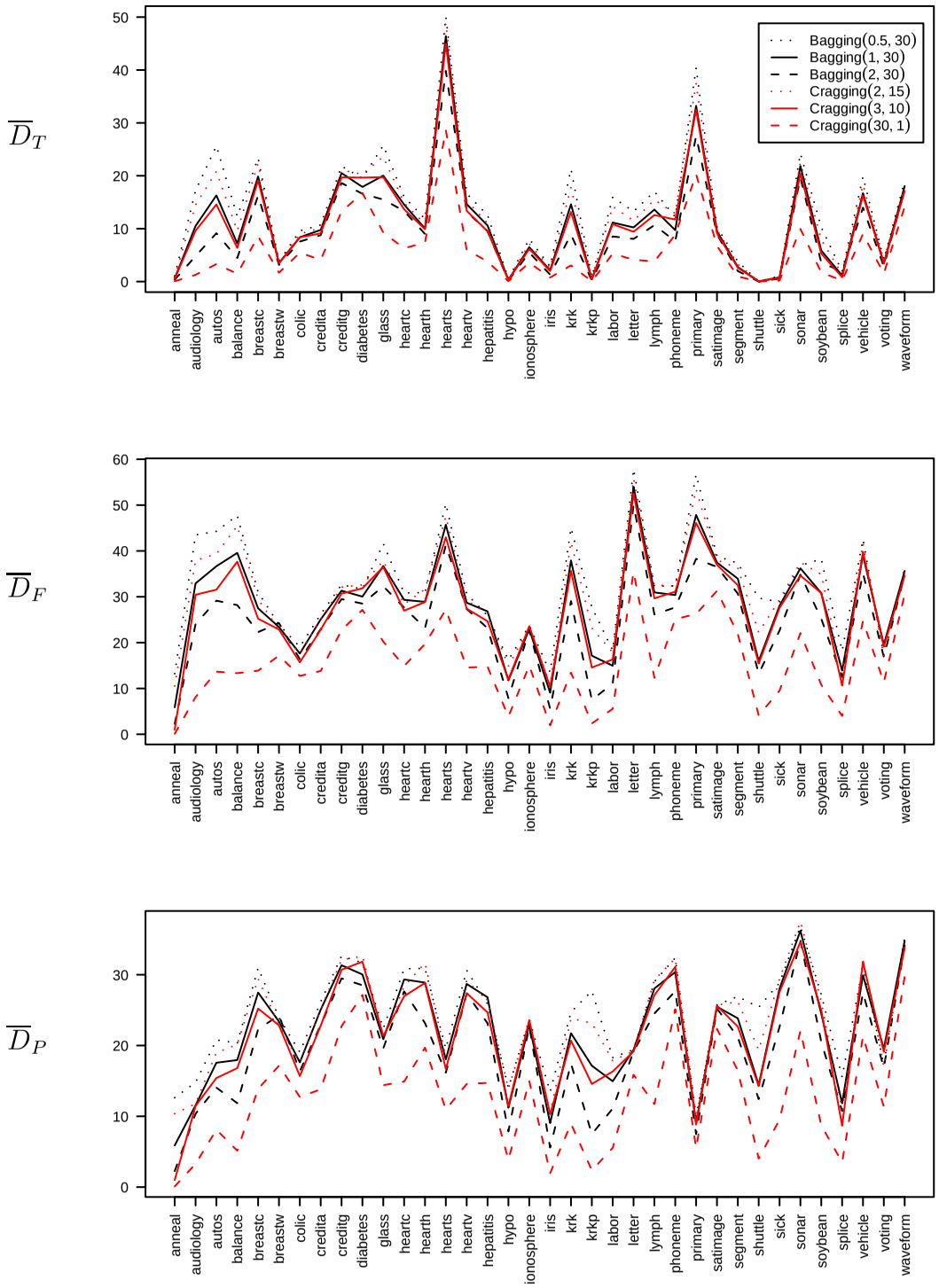

Figure 6.4: Measured values of $\overline{D}_T$, $\overline{D}_F$, and $\overline{D}_P$.

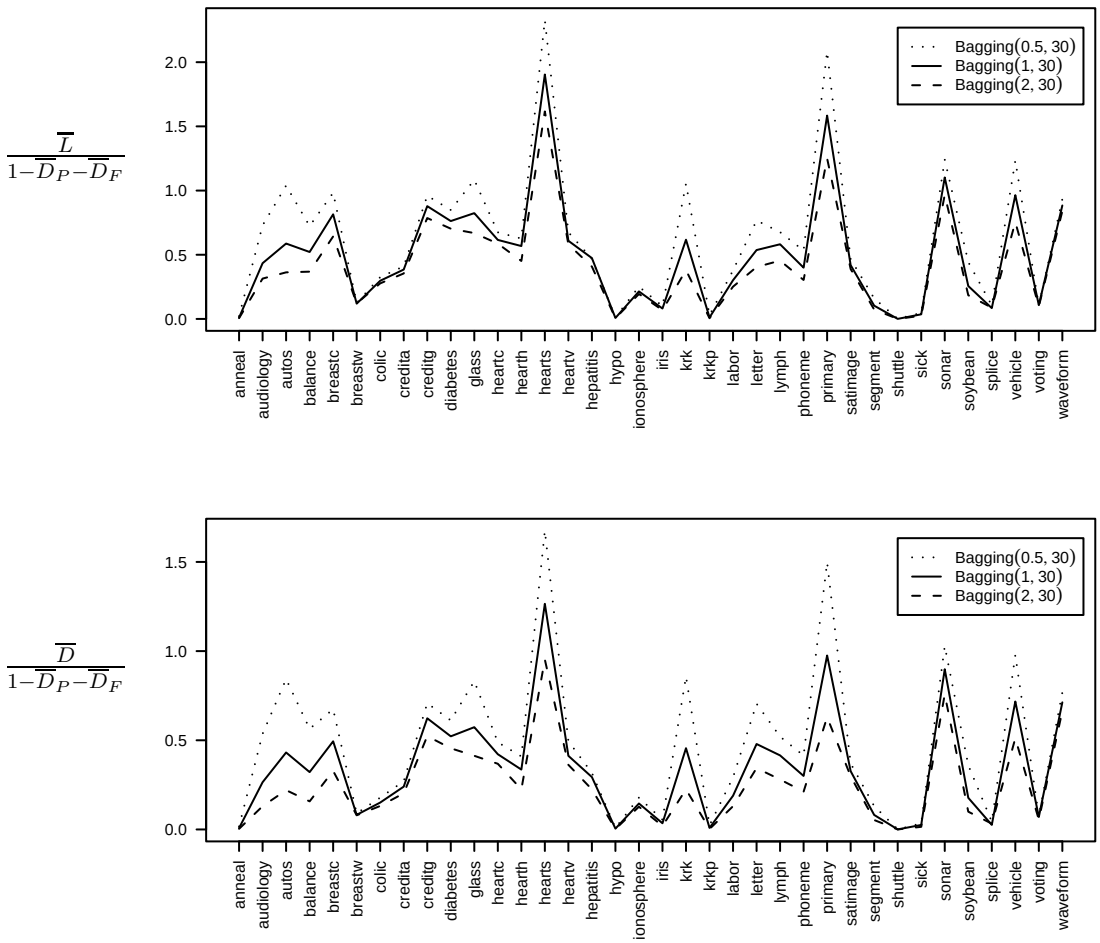

Figure 6.5: Values of $\overline{L}/(1-\overline{D}_P-\overline{D}_F)$ and $\overline{D}/(1-\overline{D}_P-\overline{D}_F)$ for $Bagging(0.5;30)$, $Bagging(1;30)$, and $Bagging(2;30)$.

Table 6.3 shows the geometric means for all components of the loss decomposition over all tested problem domains in percent. All tested sampling schemes outperform on average a single base classifier, which has an average expected loss of 9.49 percent (Table B.1 on page 85). However, when considering individual datasets (Table I.1 on page 158), the expected ensemble loss L of ensembles including those learned by standard Bagging can – contrary to popular beliefs – sometimes be greater than that of the base learner ($L_{C4.5}$), independent of whether majority vote or probabilistic voting is used as the voting function. Here, this happens for the datasets annealing, krkp, splice and heart-h (although not significantly).

This is a direct manifestation of the accuracy-diversity trade-off: In order to learn diverse classifiers, bagging generates bootstrap replicates of the original data set. Those bootstrap replicates will on average contain only about 63.2 percent of the original training examples, resulting in member classifiers which are less accurate

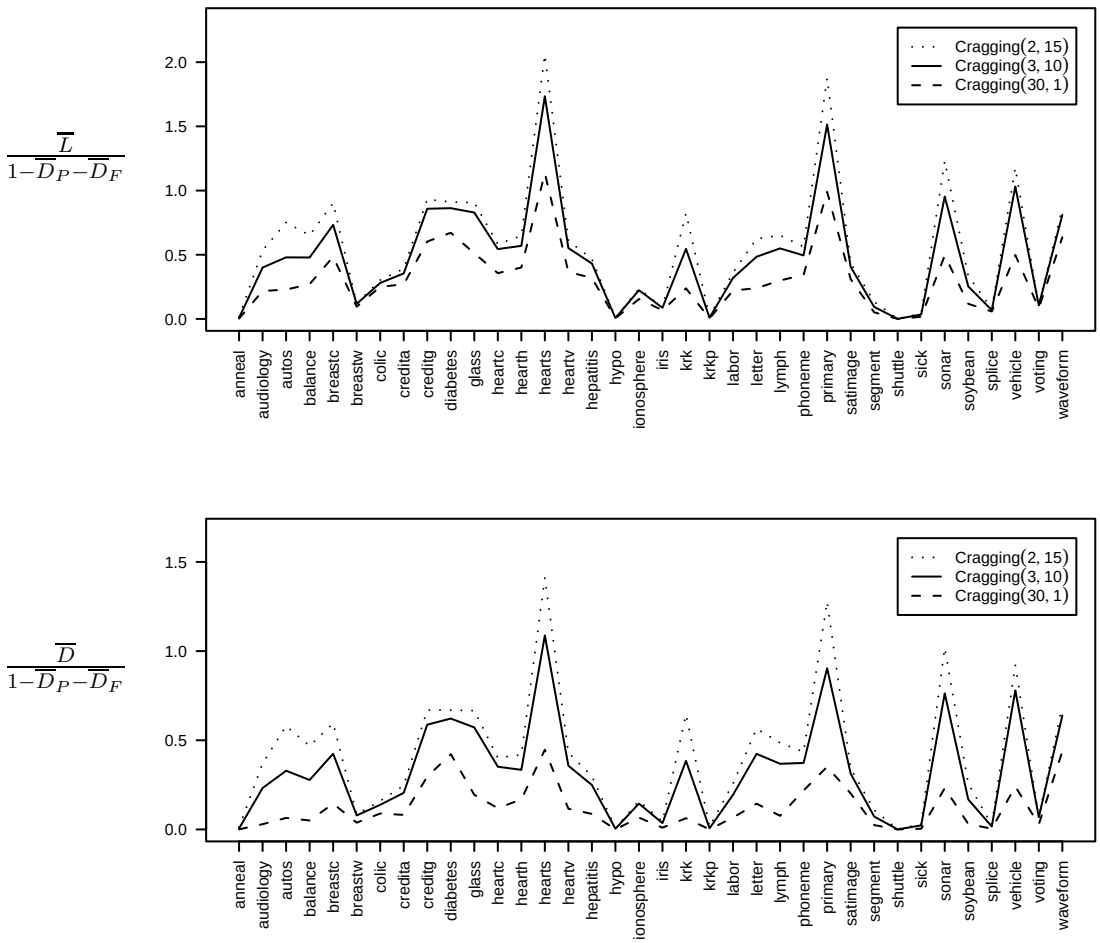

Figure 6.6: Values of $\overline{L}/(1-\overline{D}_P-\overline{D}_F)$ and $\overline{D}/(1-\overline{D}_P-\overline{D}_F)$ for *Cragging*$(2;15)$, *Cragging*$(3;10)$, and *Cragging*$(30;1)$.

than a single classifier learned from all the original data; i.e. $\overline{L} > L_{C4.5}$. Sometimes the gain in diversity will not be enough to compensate for this loss in accuracy.

What is more, the expected ensemble loss L can sometimes be greater than the expected mean member loss $\overline{L}$ – namely when $\overline{LD}_P > (1-\overline{L})\overline{D}_T$, as is the case for all tested ensemble methods on the annealing dataset. The possibility of having $L > \overline{L}$ even when $\overline{L} < 0.5$ and the ensemble members are diverse is contradictory to common expectations among researchers and markedly different from ensemble behavior under squared loss.

Table I.1 also shows that, out of the sampling methods tried, *Bagging*$(1;30)$ performs best on only 3 out of 36 datasets. On the the remaining 33 datasets it is outperformed by at least one of the other sampling methods. Thus, it is beneficial to consider alternative sampling schemes besides "normal" Bagging when building classifier ensembles. As none of the sampling schemes consistently outperforms the

others, a sensible approach would be to "tweak" the sampling scheme using a hold-out set of evaluation data. However, the enormous computational effort required to construct and evaluate multitudes of classifier ensembles may present a serious obstacle to this undertaking. Ideally, one would like to be able to determine the performance of the ensemble methods in advance, using only properties of the data set and the base learner. While the inherent complexity of the base learner C4.5 prevents a rigorous theoretical predictive analysis, we can still use the loss decomposition from Chapter 5 to perform a comparative analysis among the sampling schemes, in order to at least reduce somewhat the number of parameter settings that have to be examined.

Consider Figure 6.1 and Figure 6.2 on page 68, which show the ensemble loss (y-axis) vs. each of the loss decomposition components (x-axis). In each of the plots, one point is shown for each data set and ensemble method. The remarkable similarity between Figure 6.1 and Figure 6.2 reinforces the hypothesis from Chapter 3 that similar changes to diversity and member accuracy will lead to similar changes in ensemble loss, independent of how those changes were arrived at.

Figure 6.3 on page 69 and Figure 6.4 on page 70 show the measured values of the loss decomposition components for all tested sampling schemes. While there is no obvious pattern discernible for the expected ensemble loss L itself (Figure 6.3 top), the individual components of the ensemble loss do seem to change in a consistent manner when varying the sampling scheme. That is, we usually have:

- $\overline{L}_{\mathcal{B}(0.5;30)} > \overline{L}_{C(2;15)} > \overline{L}_{\mathcal{B}(1;30)} > \overline{L}_{C(3;10)} > \overline{L}_{\mathcal{B}(2;30)} > \overline{L}_{C(30;1)}$
- $\overline{D}_{\mathcal{B}(0.5;30)} > \overline{D}_{C(2;15)} > \overline{D}_{\mathcal{B}(1;30)} > \overline{D}_{C(3;10)} > \overline{D}_{\mathcal{B}(2;30)} > \overline{D}_{C(30;1)}$
- $\overline{D}_{T,\mathcal{B}(0.5;30)} > \overline{D}_{T,C(2;15)} > \overline{D}_{T,\mathcal{B}(1;30)} > \overline{D}_{T,C(3;10)} > \overline{D}_{T,\mathcal{B}(2;30)} > \overline{D}_{T,C(30;1)}$
- $\overline{D}_{F,\mathcal{B}(0.5;30)} > \overline{D}_{F,C(2;15)} > \overline{D}_{F,\mathcal{B}(1;30)} > \overline{D}_{F,C(3;10)} > \overline{D}_{F,\mathcal{B}(2;30)} > \overline{D}_{F,C(30;1)}$, and
- $\overline{D}_{P,\mathcal{B}(0.5;30)} > \overline{D}_{P,C(2;15)} > \overline{D}_{P,\mathcal{B}(1;30)} > \overline{D}_{P,C(3;10)} > \overline{D}_{P,\mathcal{B}(2;30)} > \overline{D}_{P,C(30;1)}$

What is more, not only the directions but also the magnitudes of the variations seem to be consistent – and therefore predictable. This behavior was to be expected for the expected mean member loss $\overline{L}$ as well as the expected diversity $\overline{D}$ (see Chapter 3) and to some extent for the expected diversities over correctly and incorrectly predicted instances, $\overline{D}_T$ and $\overline{D}_F$, due to the nature of the ensemble learning process via data resampling. It is not immediately obvious why $\overline{D}_P$ (the expected probability of an ensemble member predicting correctly, given that the ensemble a whole predicts incorrectly) behaves in such a consistent manner; e.g. why $\overline{D}_P$ for example consistently decreases when changing the sampling scheme from $Bagging(0.5; 30)$ to $Bagging(2; 30)$. We believe that this happens because, when the base classifier has more training instances available, instances will tend to be predicted correctly more often by the ensemble as a whole (i.e, instances move from F to T), and the only instances remaining in F are those which are inherently difficult to predict.

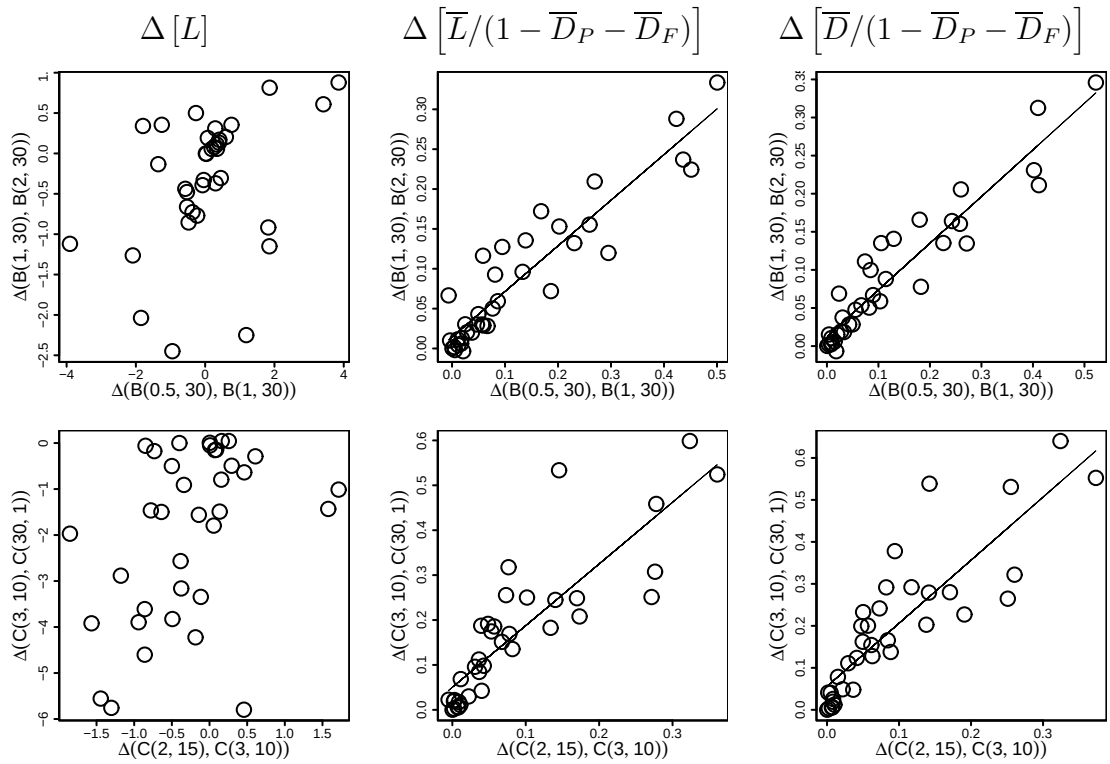

Figure 6.7: Consistency of changes in L, $\overline{L}/(1-\overline{D}_P-\overline{D}_F)$, and $\overline{D}/(1-\overline{D}_P-\overline{D}_F)$ when switching sampling schemes.

As a result of this consistency, we can infer expected values for the ensemble loss components (and hence for the ensemble loss itself) for any one sampling scheme from the measured values of the ensemble loss components of other sampling schemes. Using Theorem 5.9 from page 59, we can write the expected ensemble loss L as

$$L = \frac{\overline{L}-\overline{D}}{1-\overline{D}_P-\overline{D}_F} = \frac{\overline{L}}{1-\overline{D}_P-\overline{D}_F} - \frac{\overline{D}}{1-\overline{D}_P-\overline{D}_F} \tag{6.1}$$

The values of the two terms $\overline{L}/(1-\overline{D}_P-\overline{D}_F)$ and $\overline{D}/(1-\overline{D}_P-\overline{D}_F)$ for the surveyed problem domains are shown in Figure 6.5 and Figure 6.6, for the sampling schemes with and without replacement, respectively. It is evident that the values of $\overline{L}/(1-\overline{D}_P-\overline{D}_F)$ and $\overline{D}/(1-\overline{D}_P-\overline{D}_F)$ behave in a coherent manner across all problem domains. When changing the sampling parameters, not only the directions of the changes of the two terms are consistent across the problem domains but also the magnitudes of the changes.

Consider also Figure 6.7, which shows the differences in L (left), $\overline{L}/(1-\overline{D}_P-\overline{D}_F)$ (middle), and $\overline{D}/(1-\overline{D}_P-\overline{D}_F)$ (right) between the sampling schemes for all datasets. The top row shows the differences when switching from $Bagging(0.5;30)$ to $Bagging(1;30)$, versus the differences when switching from $Bagging(1;30)$ to $Bagging(2;30)$. The bottom row shows the differences when switching from

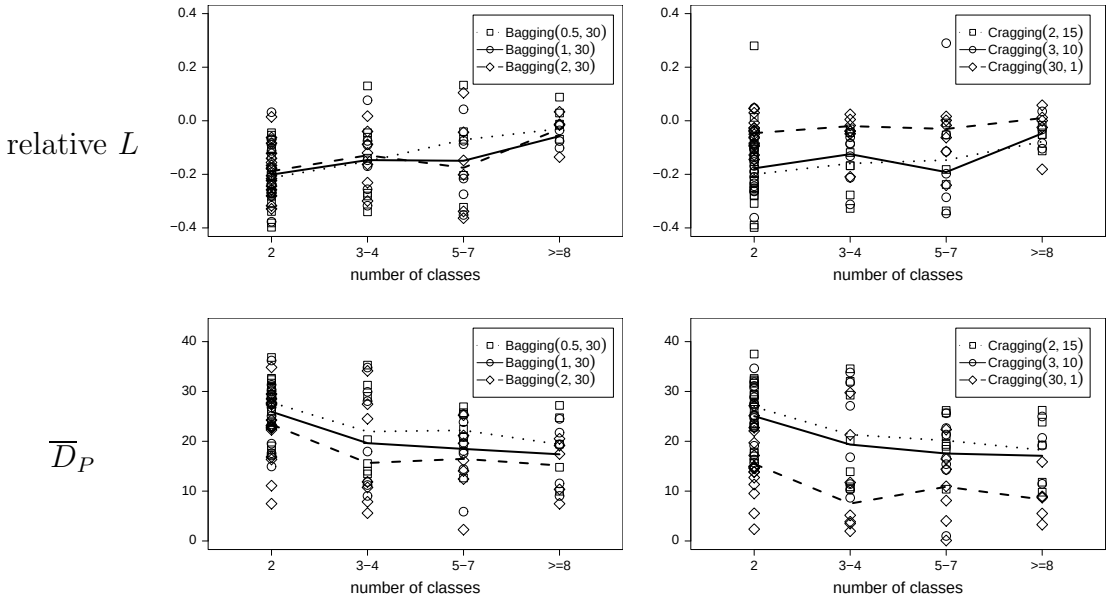

Figure 6.8: Relative ensemble losses and $\overline{D}_P$ vs. number of classes.

Cragging$(2;15)$ to *Cragging*$(3;10)$, versus the differences when switching from *Cragging*$(3;10)$ to *Cragging*$(30;1)$. There are no direct apparent relations between the differences in ensemble losses (left). However, this is because of the loss of information due to the "hidden" variables. The differences of $\overline{L}/(1-\overline{D}_P-\overline{D}_F)$ (middle), and $\overline{D}/(1-\overline{D}_P-\overline{D}_F)$ (right) do exhibit clear dependencies, and the $\langle \mathbf{x}, y\rangle$-coordinates of the $\Delta\,[L]$ plot on the left can be obtained by taking the difference of the $\langle \mathbf{x}, y\rangle$-coordinates of the other two plots (by Equation 6.1).

Thus, while we are still not able to directly predict ensemble performance without actually going through the ensemble learning process, the ensemble loss decomposition can provide some guidance as to sensible choices of ensemble learning algorithms. It can also be used nicely to predict the performance of some ensemble algorithms indirectly from the performance of other ensemble algorithms, thus reducing the number of parameter settings that have to be evaluated.

6.3 Number of Classes

In Section 5.4.2 we showed that, under 0-1 loss and on problems with more than two classes, not all diversity contributes to reducing the ensemble loss on incorrectly predicted instances, but only those member classifiers which actually predict the correct outcome.

As a measure of the amount of ensemble diversity that really entails a reduction in ensemble loss, we defined $\overline{D}_P$ as the probability of a member classifier predicting the correct outcome, given that the ensemble as a whole predicts the outcome incorrectly.

Consequently we stipulated that, under 0-1 loss and all other things being equal, ensemble algorithms like Bagging and Cragging should in general work better on problems with fewer classes.

Obviously, "all other things" are *not* equal when considering real-world datasets. Nevertheless, we believe that the data in Figure 6.8 exhibits general trends that support this hypothesis.

The top row of graphs in Figure 6.8 shows the ensemble losses relative to those of a single base classifier versus the number of classes $|Y|$. For each ensemble method M, the relative ensemble loss is $(L_M - L_{C4.5})/L_{C4.5}$ and measures how bad an ensemble performs compared to a single base classifier. A point is shown for each dataset and ensemble method, with the lines drawn through the averages of relative ensemble losses taken over the datasets.

There does seem to be a general tendency for the relative ensemble loss to increase as $|Y|$ increases. This effect is more pronounced for ensembles with lower mean member accuracy and higher diversity ($Bagging(0.5; 30)$, $Cragging(2; 15)$) than for ensembles with the trade-off more placed in direction of higher mean member accuracy and lower diversity ($Bagging(2; 30)$, $Cragging(30; 1)$). The bottom row of graphs in Figure 6.8 shows that the general increase in relative ensemble loss is accompanied by a general decrease in $\overline{D}_P$ as the number of classes $|Y|$ increases.

Thus, it may be advisable to decompose multiple-class problems into a series of two-class problems prior to applying ensemble methods. Error-Correcting Output Coding ([19]) is an ensemble technique designed to do just this. It can be applied independently of or conjointly with other ensemble methods such as Bagging or Cragging.

7. Conclusions and Further Research

7.1 Summary

In recent years, one of the most active areas of research in supervised learning has been to study methods for constructing ensembles of classifiers ([17]). Bagging is one of the most popular of those methods. It has been observed that ensembles in general and those constructed via Bagging in particular often significantly outperform the individual classifiers they consist of.

However, a thorough theoretical understanding of ensembles has been lacking so far, and with that precise answers to crucial questions such as exactly why, how, and under which conditions they perform well. Rather, analysis of ensembles has been rather ad-hoc in nature, and so have been the resulting explanations of ensemble performance. In Chapter 4 we surveyed popular methods for ensemble analysis, and pointed out some of their shortcomings.

Although it is generally agreed upon that diversity among the component classifiers is the principal source of the performance gains, a rigorous theoretical analysis has been missing – perhaps due to the fact there was no coherent definition of what exactly is diversity and how it is to be measured. Also often overlooked is the fact that member accuracy and diversity are two goals that directly contradict each other, and therefore constructing a good ensemble involves finding a good compromise between those two.

In Chapter 3 we showed that the optimal trade-off between member accuracy and diversity depends on properties of both the base classifier inducer and the problem domain. Finding the optimal trade-off necessitates an analysis of the exact quantitative relationships between ensemble performance, ensemble member accuracy, and diversity.

In Section 5.2 we proposed general definitions of accuracy and diversity among a set of ensemble members. Those definitions are applicable to all single-stage voting ensembles, under a wide range of voting functions and under any given loss function. We showed that the classic definitions under squared loss are but a special case of our general definitions, and extended the classic loss decomposition under squared

loss to the case where the member classifiers are distribution classifiers.

We then instantiated the loss decomposition for 0-1 loss and derived the exact quantitative relationships between the ensemble loss components, thus revealing the exact numerical form of the accuracy-diversity trade-off under 0-1 loss. This held valuable insights into ensemble behavior and helped explain some unexpected experimental results regarding the performance of ensemble methods.

Under squared loss, member accuracy and diversity are necessary and sufficient conditions for an ensemble to outperform its members. Using consistent definitions for accuracy and diversity, we found that under 0-1 loss accuracy and diversity are still necessary conditions for an ensemble to outperform its members but no longer sufficient ones. Thus, this difference in ensemble behavior appears to arise directly from the choice of 0-1 loss as the performance measure.

In Section 5.5 we proved general upper and lower bounds for the expected ensemble loss L in terms of the expected mean member loss $\overline{L}$ and the expected diversity $\overline{D}$. The lower bound holds for any transitive loss function. The upper bound holds for any loss function which is both transitive and symmetric.

Given the nature of the accuracy-diversity trade-off, the question arises whether Bagging can be improved upon by ensemble methods which place the emphasis either on more accurate or on more diverse ensembles. Experimental results show indeed that simple variations of the sampling scheme can frequently lower (or increase) the expected ensemble loss drastically. The optimal trade-off is dependent on both properties of the base learner and properties of the problem domain.

Faced with a particular problem domain, a practitioner has to choose a good ensemble learning method appropriate for this domain. One approach to this is to test several methods with different parameter settings and choose the one that performs best. This approach is not always feasible, however, for reasons of both computational complexity and statistical validity. In Chapter 6 we showed how the ensemble loss decomposition can de used to reduce the number of ensemble methods and parameter settings that must be experimentally tried.

7.2 Further Research

The following problems remain open and subject to further research:

7.2.1 Weighted Ensembles

An immediate practical application of the ensemble loss decomposition from Chapter 5 would arise from solving the following optimization problem for certain loss functions: Given a set of member classifiers with given member losses and diversities, find a set of weights that minimizes the expected ensemble loss.

7.2.2 Other Ensemble Learning Methods

Besides Bagging and Cragging, there exist many more single-stage ensemble methods which work by iterative resampling. These can all be analyzed under the loss decomposition framework proposed here. In each case we would seek an answer to questions such as *How does the performance of the ensemble method relate to accuracy and diversity of the ensemble members?* and *How do domain properties and method parameters affect accuracy and diversity of the ensemble members?*

In doing so, we would seek confirmation (or disproof) of the following hypotheses:

- that member accuracy and diversity are essentially what makes single-stage voting ensembles 'tick';
- that those two properties can be quantified and put into a mathematical relation with each other; as well as with ensemble performance;
- that those two are contradictory goals and therefore a trade-off has to be made;
- that "similar" parameter settings for the ensemble learning methods will lead to similar accuracy-diversity trade-offs
- that the best trade-off point depends very much of properties of the dataset at hand; and
- that equivalence relations can be found between different ensemble learning methods in the sense that, given a dataset, method A with parameters a and method B with parameters b will induce ensembles with the same accuracy and diversity, and hence the same ensemble performance.

Specifically, we would like to investigate theoretically and experimentally the following methods:

- Recent work in the statistical community ([11, 12, 13, 30]) has come up with subagging (subsample aggregating), another iterative sampling scheme similar to Bagging and Cragging. In each iteration, subagging draws a bootstrap sample without replacement, with a subsample size $|S_i| := \alpha |S|$, where $|S|$ is the size of the original training sample, and α is a parameter to the learning method. Theoretical results in [11] suggest that, under squared loss, subagging may improve on both Bagging and Cragging.
- Instead of re-sampling over the instances, one can also re-sample over the set of features. Or one can re-sample simultaneously over both features and instances ([54, 81, 42]). Experiments showed promising performance improvements over "standard" Bagging.
- Two alternatives to Bagging are Boosting ([71] and Randomization ([16]). It seems possible to construct a unifying theoretical framework by viewing them all as iterative sampling from some space of candidate classifiers.

- Another approach to iteratively generating accurate yet diverse classifiers, instead of re-sampling, is to add some specified amount of random noise to the training data itself. This approach has the advantage that it is very general, i.e. it is applicable to a wide range of problem domains, loss functions, and base learners. It is also simple enough to be theoretically analyzable, and the amounts of member accuracy lost and diversity gained are controllable to a very fine degree.

7.2.3 Other Loss Functions

We would also like to see the unified decomposition instantiated for loss functions other than 0-1 loss and squared loss. In particular, there are many classification and regression problems where some types of mistakes are more costly than others. For those problems, asymmetric performance measures are more appropriate than symmetric ones. This makes asymmetric performance measures a difficult but attractive research target.

An alternative promising route to asymmetric performance measures could be to focus on ensembles which are distribution classifiers, i.e., ensembles which return as output a belief distribution over the outcome space, together with an appropriate performance measure. Theoretical results from [38] suggest that using log loss or Kullback-Leibler Divergence as performance measures might lead to simpler (and therefore more intuitive and easier to analyze) ensemble loss decompositions.

The first step on this route could be to investigate how ensemble performances under those two different paradigms relate to each other.

7.2.4 Other Voting Functions

There are voting functions which are not covered by the definition of democratic voting as in Definition 5.1, such as aristocratic or progressive voting. Those voting schemes are not covered by our particular instantiation of the decomposition for democratic voting schemes under 0-1 loss (Equation 5.30).

The decomposition could be instantiated for those voting schemes as well, by substituting the voting function $V_{\text{dem}}(\widehat{\mathbf{y}}(\mathbf{x}), \mathbf{w})$ accordingly. However, results in [6] suggest that this may not be a fruitful route to pursue.

7.2.5 Other Base Learners

The goal of the experiments conducted for this thesis was to compare different ensemble methods with each other, therefore all experiments where conducted using the same base learning algorithm, J4.8, for consistency. The choice of J4.8 as the base learner was motivated by the fact that it is one of the most popular learning algorithms readily available. Furthermore, it has been frequently observed that J4.8

is a good choice as a base learner for ensemble methods (e.g. [16, 59, 66]), as its high instability ("variance") will lead to member classifiers which are both accurate (at least on the training data) and highly diverse.

The theoretical results in Chapter 5 are completely independent of the choice of the base learner. All our experimental results are consistent with those theoretical results. We would therefore expect similar experimental results if the experiments were repeated using other base learners, such as e.g. Naive Bayes or simple Decision Stumps. However, those learning algorithms choose their model from an effectively smaller model search space, resulting in lower variance (less instability) than the unpruned decision trees used in this thesis. Therefore, while we still would expect the same effects as those observed here, we would also expect these effects to become less pronounced.

Ideally, we would like a theory that tells us in advance which ensemble learning method with which parameters will perform best for a given problem domain. This is a challenging research question which remains open, even for the case of comparing some simple Bagging variants.

In this thesis, in-depth theoretical analysis of Bagging and Cragging was hindered in part by the fact that the base classifier inducer itself (C4.5) is inherently complex and enigmatic. The use of simpler base classifier inducers will probably not give immediate practical performance results, but could lead to further useful theoretic insights.

A good candidate for investigation is an random brute-force approach, that is, the base classifier inducer returns a randomly selected classifier out of some predetermined set of candidate classifiers, and the set of candidate classifiers is determined before any training instances have been seen.

7.2.6 Multi-Stage Ensembles

Multi stage-ensembles (Stacking, Cascading, Serial Combination etc.) are not covered in this thesis. They are more difficult to analyze than single-stage ensembles, as the outputs of some ensemble members are inherently dependent on the outputs of other ensemble members.

Some multi-stage ensembles can be transformed into equivalent single-stage voting ensembles. The extension of the framework of Chapter 5 to multi-stage ensembles requires the introduction of a variable $\widehat{X}$ for the member input space and an input transformation function $T : X -> \widehat{X}^{|C|}$.

However, the promise of this venture is questionable, as member accuracy and diversity are probably not the only factors influencing the performance of multi-stage ensembles. Other analysis frameworks look more promising. For example, just as a decision tree is a multi-stage ensemble of decision stumps, Cascading, Stacking, etc. could be analyzed as a decision tree with complicated node functions.

7.3 Conclusions

We can imagine a powerful, general theoretical framework that encompasses many of the ensemble learning methods currently in use, either as special cases or via proofs of equivalence. We would like to see a theoretical framework that can explain ensemble behavior qualitatively and quantitatively. Ideally, such a theoretical model would provide information about expected ensemble performance in advance for a given data sample, without actually going through the process of training and testing an ensemble. The form of the information provided could be absolute ("On this problem, Method A with parameters a will achieve performance p") or relative ("On this problem, Method A will perform better than Method B").

Having such a framework would not only be of immediate practical relevance, it very likely would also lead to improved ensemble learning methods and to a better understanding of machine learning in general. We can only hope that our work may have provided another tiny step towards this worthy goal.

A. Dataset Providers

We are grateful to the following individuals and organizations for providing the datasets used in the experiments.

Dataset:	*Creator / Donor Acknowledgments:*
anneal	David Sterling and Wray Buntine
audiology	Prof. Jergen at Baylor College of Medicine / Bruce Porter (University of Texas)
autos	Jeffrey C. Schlimmer (Jeffrey.Schlimmer@a.gp.cs.cmu.edu)
balance	Tim Hume (hume@ics.uci.edu)
breastc	M. Zwitter and M. Soklic (University Medical Centre, Institute of Oncology, Ljubljana)
breastw	William H. Wolberg (Wisconsin Hospitals, Madison, Wisconsin, USA) / Olvi Mangasarian (mangasarian@cs.wisc.edu)
colic	Mary McLeish and Matt Cecile (Department of Computer Science, University of Guelph, Canada) / Will Taylor (taylor@pluto.arc.nasa.gov)
credita	Ross Quinlan (quinlan@cs.su.oz.au)
creditg	Hans Hofmann (Institut für Statistik und Ökonometrie, Universität Hamburg)
diabetes	National Institute of Diabetes and Digestive and Kidney Diseases / Vincent Sigillito (vgs@aplcen.apl.jhu.edu)
glass	B. German (Home Office Forensic Science Service) / Vina Spiehler (Diagnostic Products Corporation)
heartc	Robert Detranoi (Long Beach and Cleveland Clinic Foundation)
hearth	Andras Janosi (Hungarian Institute of Cardiology, Budapest)
hearts	William Steinbrunn (University Hospital, Zurich)
heartv	Robert Detrano and V.A. Medical Center
hepatitis	G. Gong (Carnegie-Mellon University) and Bojan Cestnik (Jozef Stefan Institute, Ljubljana)
hypo	Garavan Institute and Ross Quinlan (New South Wales Institute, Sydney)
ionosphere	Vince Sigillito (Space Physics Group, Applied Physics Laboratory, Johns Hopkins University)

Dataset:	*Creator / Donor Acknowledgments:*
iris	R.A. Fisher / Michael Marshall (marshall@pluto.arc.nasa.gov)
krk	Michael Bain and Arthur van Hoff (Turing Institute, Glasgow, UK)
krkp	Alen Shapiro / Rob Holte and Peter Clark (Turing Institute, Glasgow)
labor	Industrial Relations Information Service, Ottawa, Ontario, Canada / Stan Matwin (Computer Science Dept, University of Ottawa)
letter	David J. Slate (Odesta Corporation, Evanston, IL)
lymph	M. Zwitter and M. Soklic (University Medical Centre, Institute of Oncology, Ljubljana) / Igor Kononenko (University E.Kardelj) and Bojan Cestnik (Jozef Stefan Institute, Ljubljana)
phoneme	Dominique Van Cappel and Thomson-Sintra (Sophia Antipolis Cedex, France)
primary	M. Zwitter and M. Soklic (University Medical Centre, Institute of Oncology, Ljubljana) / Igor Kononenko (University E.Kardelj) and Bojan Cestnik (Jozef Stefan Institute, Ljubljana)
satimage	Karen Hall (Centre for Remote Sensing, University of New South Wales) and Alistair Sutherland (Statistics Dept., Strathclyde University, Glasgow) / Ashwin Srinivasan (Department of Statistics and Modeling Science, University of Strathclyde, Glasgow)
segment	Carla Brodley (Vision Group, University of Massachusetts)
shuttle	NASA / Jason Catlett (University of Sydney)
sick	Garavan Institute and Ross Quinlan (New South Wales Institute, Sydney)
sonar	Terry Sejnowski (Salk Institute and the University of California at San Diego) and R. Paul Gorman (Allied-Signal Aerospace Technology Center) / Scott E. Fahlman
splice	Genbank (genbank.bio.net) / G. Towell, M. Noordewier, and J. Shavlik
vehicle	Pete Mowforth and Barry Shepherd (Turing Institute, Glasgow) / Alistair Sutherland (Statistics Dept., Strathclyde University, Glasgow)
voting	Congressional Quarterly Almanac / Jeff Schlimmer
waveform	Wadsworth International Group, Belmont, California / David Aha

B. Performance of Base Classifier

Dataset	L
anneal	0.14±0.07
audiology	18.88±0.97
autos	17.53±1.30
balance	21.95±0.69
breastc	34.35±1.99
breastw	6.22±0.35
colic	16.58±0.75
credita	19.48±0.77
creditg	31.98±0.81
diabetes	26.47±0.91
glass	32.17±1.68
heartc	25.42±1.97
hearth	22.12±0.88
hearts	68.73±2.87
heartv	26.11±1.83
hepatitis	22.00±2.24
hypo	0.37±0.06
ionosphere	9.92±0.46
iris	6.13±0.69
krk	17.99±0.14
krkp	0.38±0.11
labor	18.50±2.46
letter	11.53±0.16
lymph	23.57±2.29
phoneme	13.63±0.24
primary	62.99±1.38
satimage	14.17±0.31
segment	3.02±0.27
shuttle	0.02±0.00
sick	1.38±0.09
sonar	26.71±2.20
soybean	8.18±0.67
splice	5.45±0.09
vehicle	26.69±1.40
voting	5.52±0.43
waveform	25.08±0.44
G. Mean	9.49

Table B.1: Performance of base classifier.

C. Error curves for *Cragging*$(n; 1)$

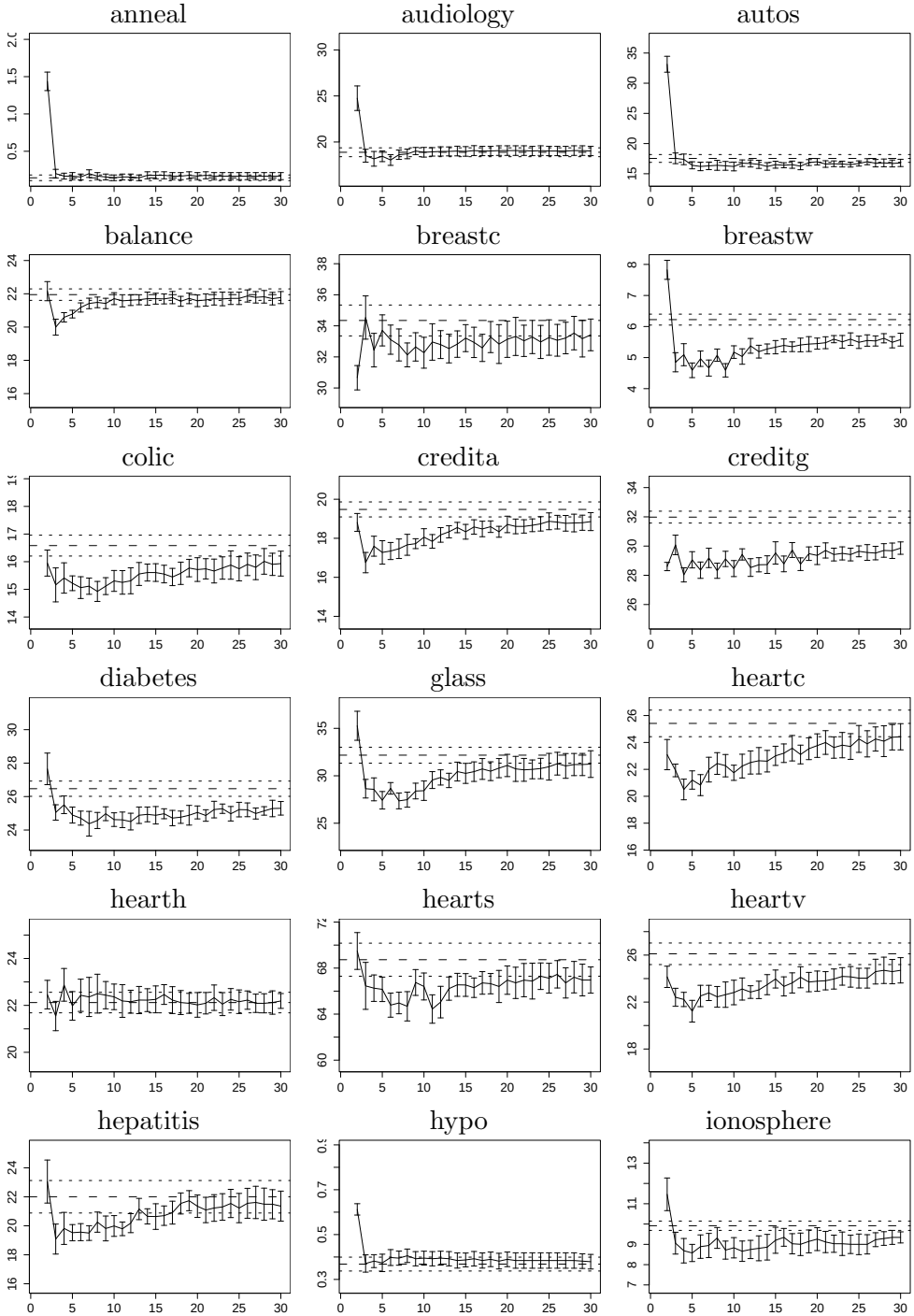

Figure C.1: Error curves for *Cragging*$(n; 1)$. *(continued on next page)*

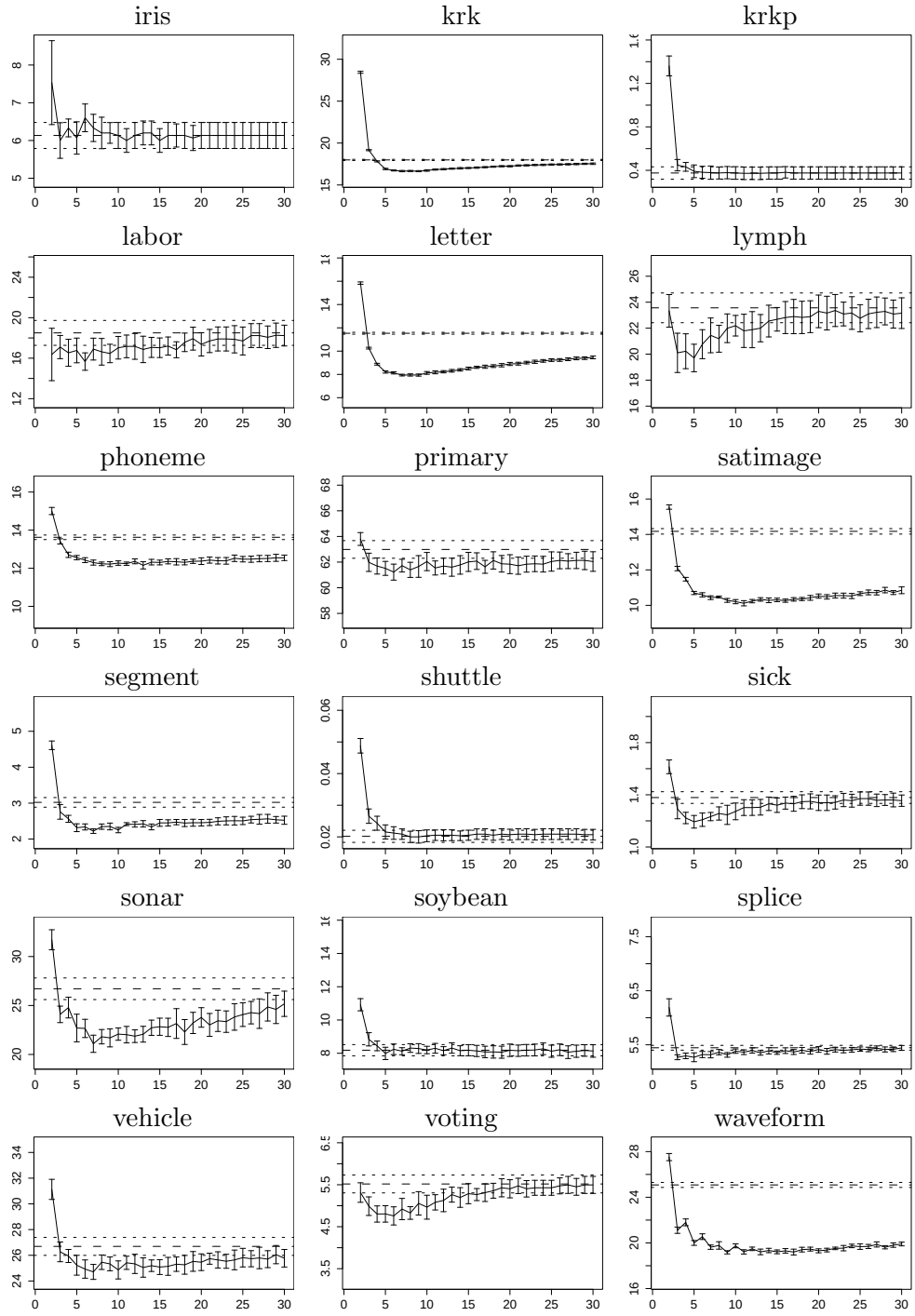

Figure C.1: Error curves for *Cragging*$(n; 1)$.

D. κ-Error Diagrams

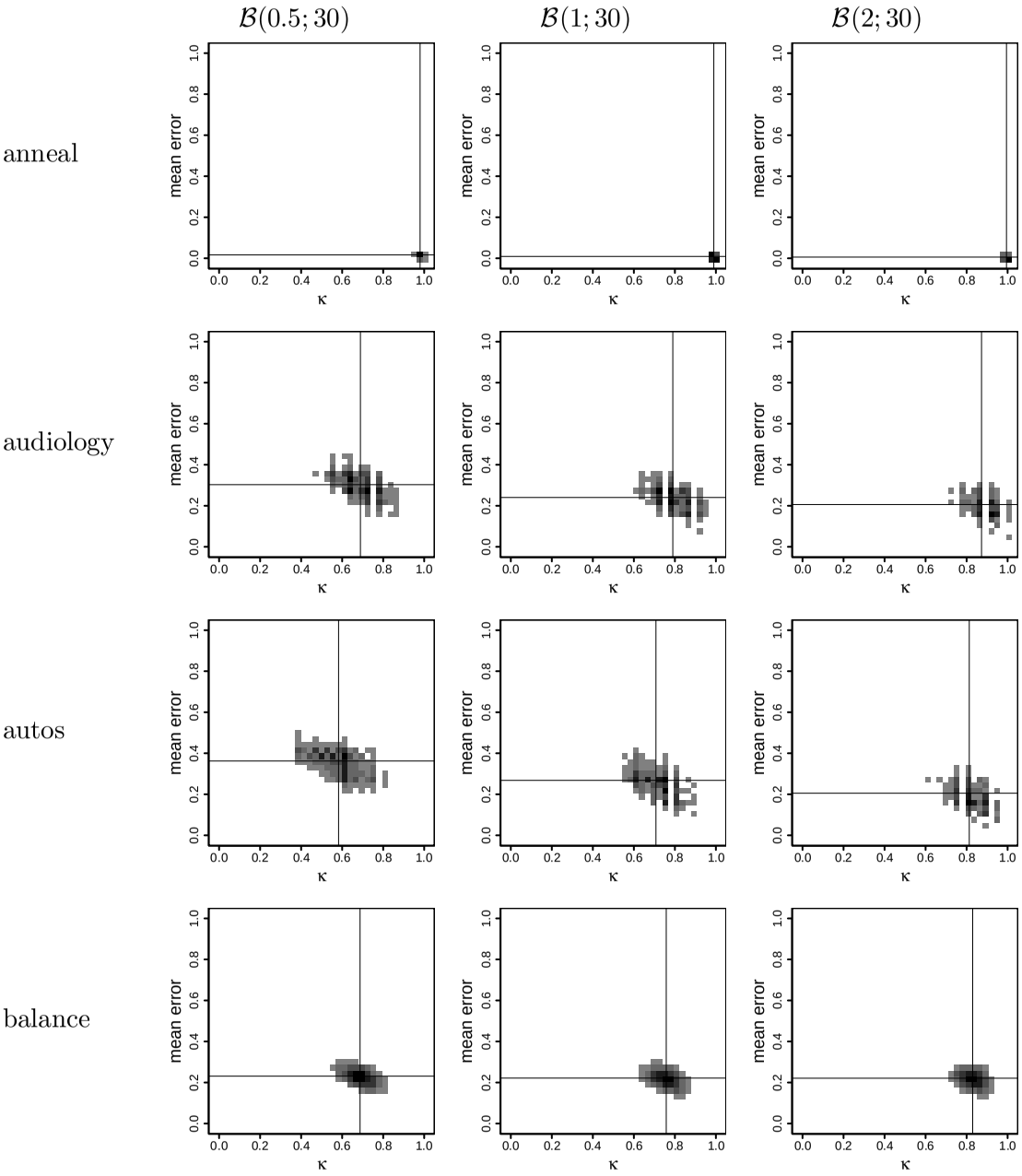

Figure D.1: κ-Error Diagrams for $\mathcal{B}(0.5; 30)$, $\mathcal{B}(1; 30)$, and $\mathcal{B}(2; 30)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

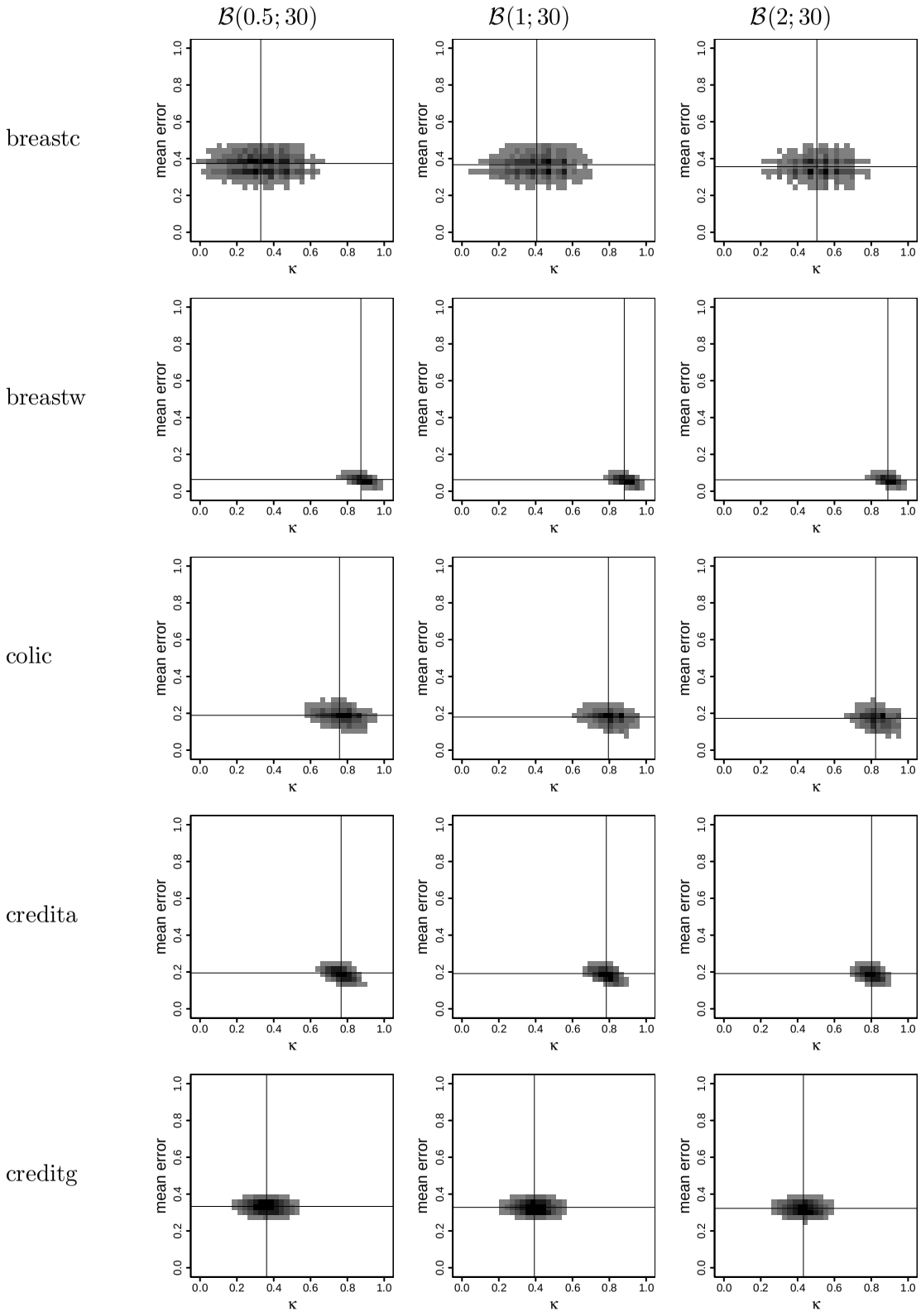

Figure D.1: κ-Error Diagrams for $\mathcal{B}(0.5; 30)$, $\mathcal{B}(1; 30)$, and $\mathcal{B}(2; 30)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

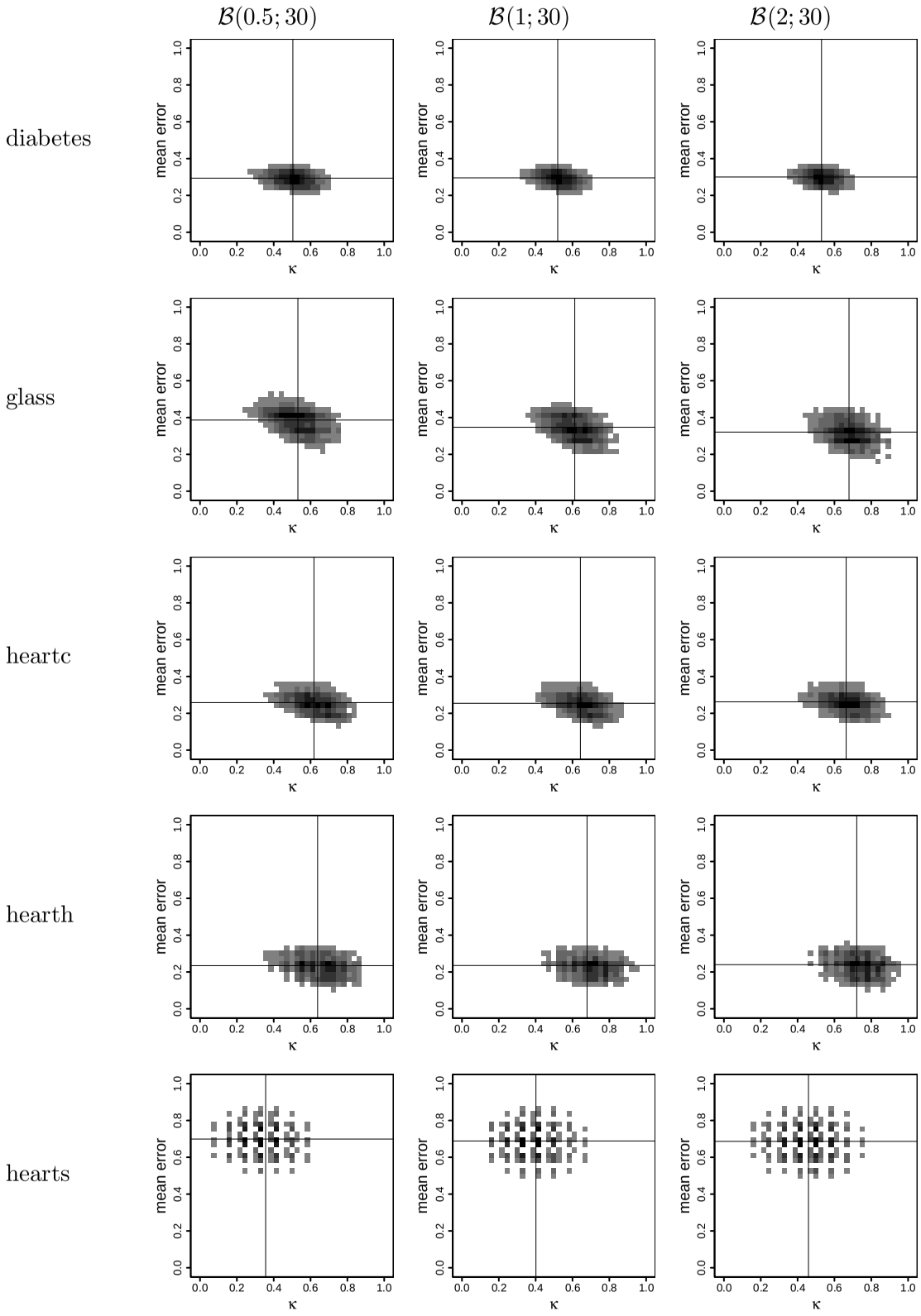

Figure D.1: κ-Error Diagrams for $\mathcal{B}(0.5; 30)$, $\mathcal{B}(1; 30)$, and $\mathcal{B}(2; 30)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

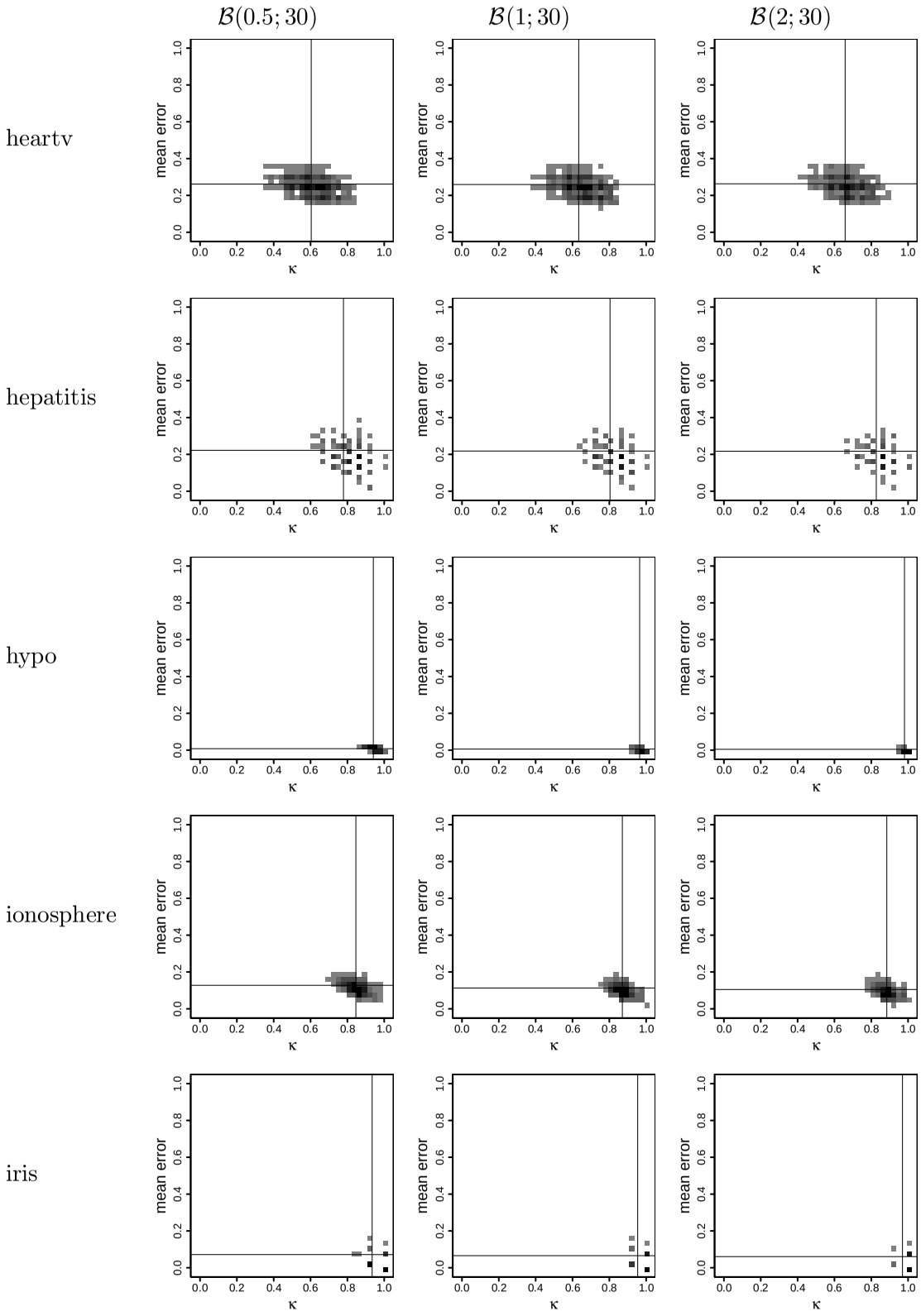

Figure D.1: κ-Error Diagrams for $\mathcal{B}(0.5; 30)$, $\mathcal{B}(1; 30)$, and $\mathcal{B}(2; 30)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

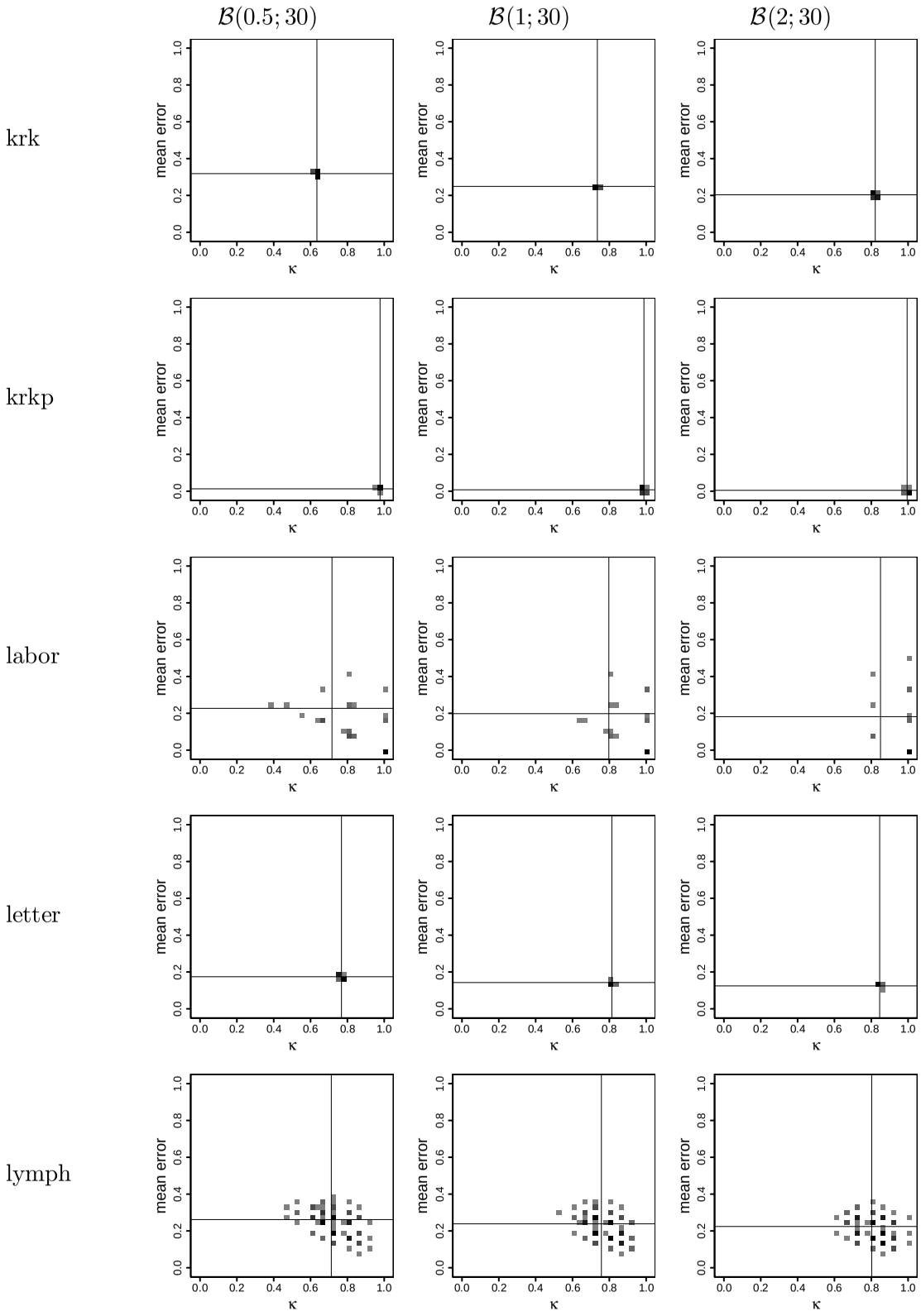

Figure D.1: κ-Error Diagrams for $\mathcal{B}(0.5; 30)$, $\mathcal{B}(1; 30)$, and $\mathcal{B}(2; 30)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

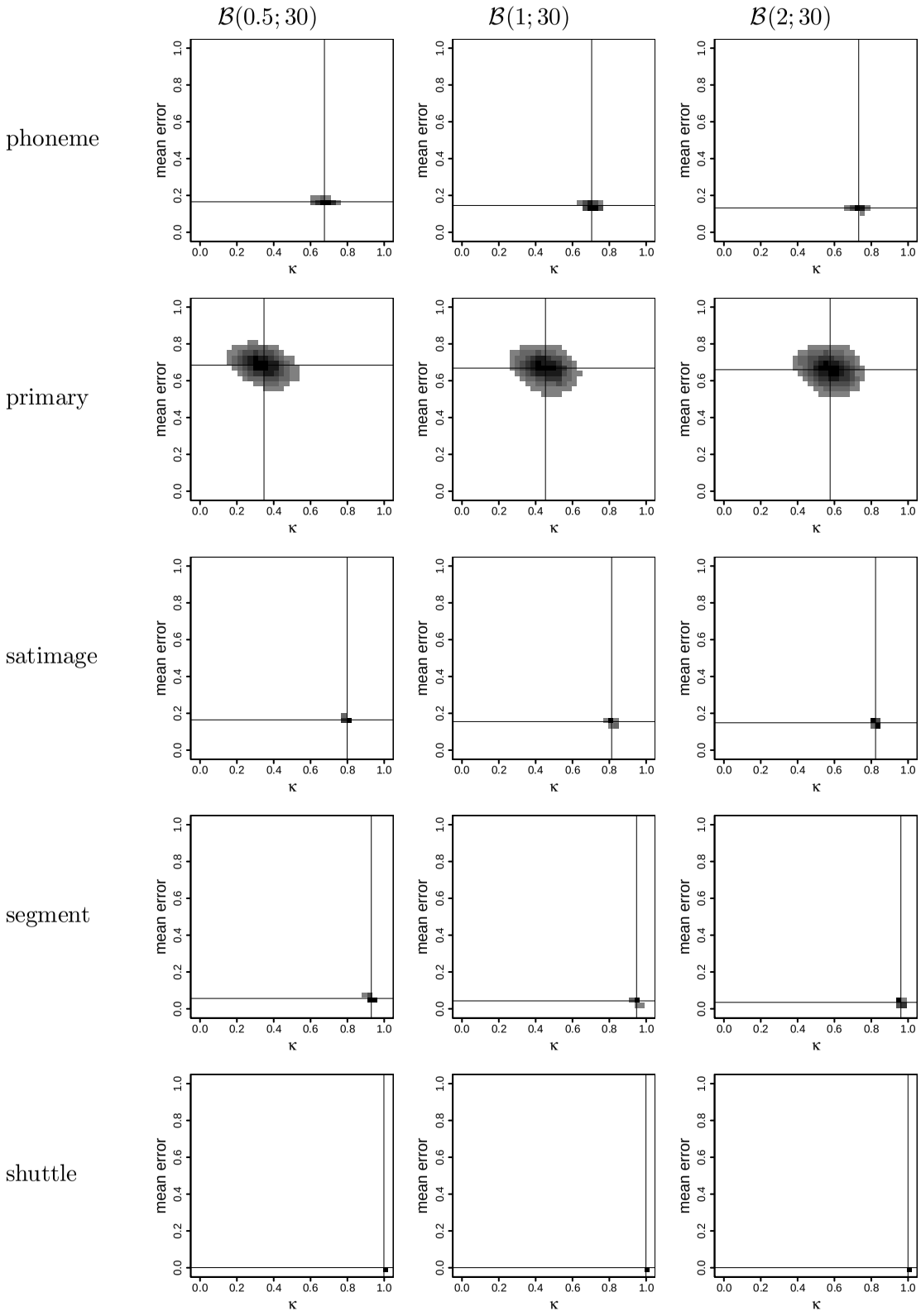

Figure D.1: κ-Error Diagrams for $\mathcal{B}(0.5; 30)$, $\mathcal{B}(1; 30)$, and $\mathcal{B}(2; 30)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

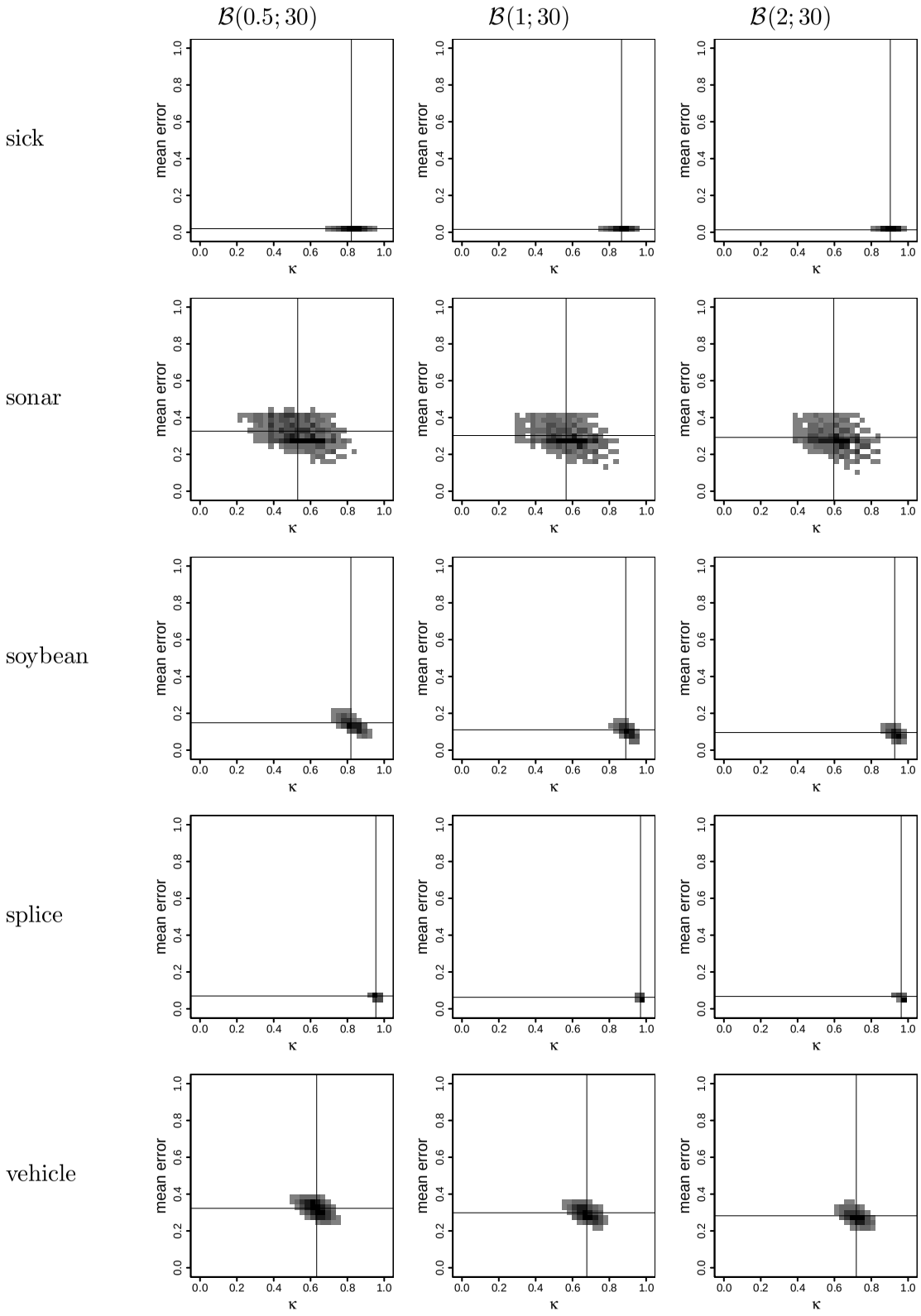

Figure D.1: κ-Error Diagrams for $\mathcal{B}(0.5; 30)$, $\mathcal{B}(1; 30)$, and $\mathcal{B}(2; 30)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

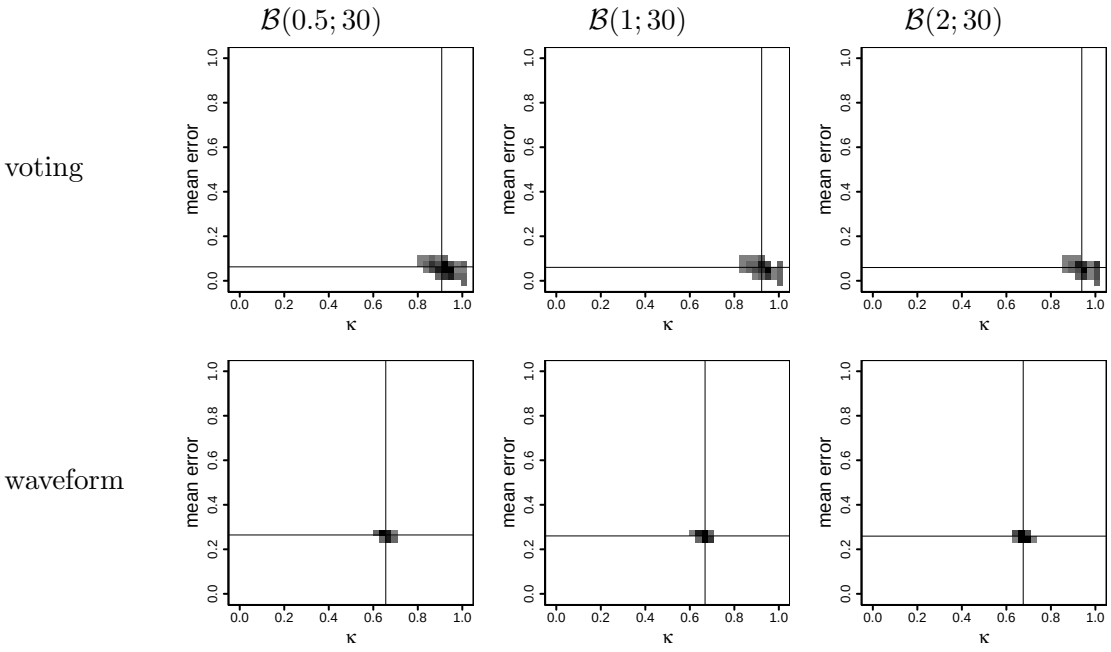

Figure D.1: κ-Error Diagrams for $\mathcal{B}(0.5;30)$, $\mathcal{B}(1;30)$, and $\mathcal{B}(2;30)$. Pairs of classifiers in the lower left corner are more accurate and more diverse.

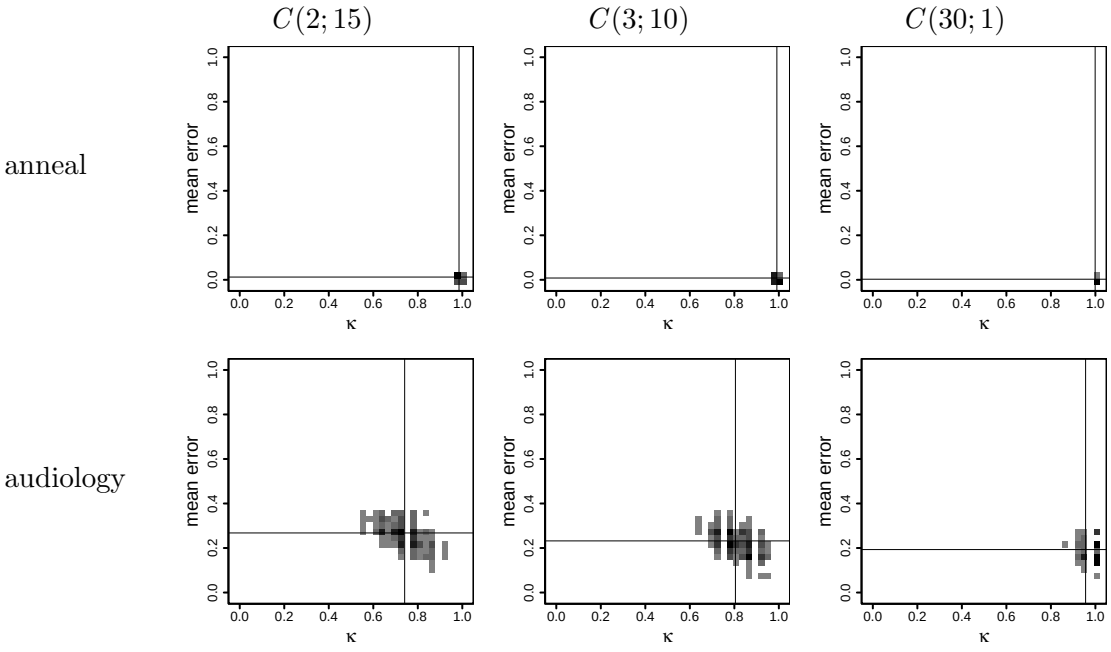

Figure D.2: κ-Error Diagrams for $C(2;15)$, $C(3;10)$, and $C(30;1)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

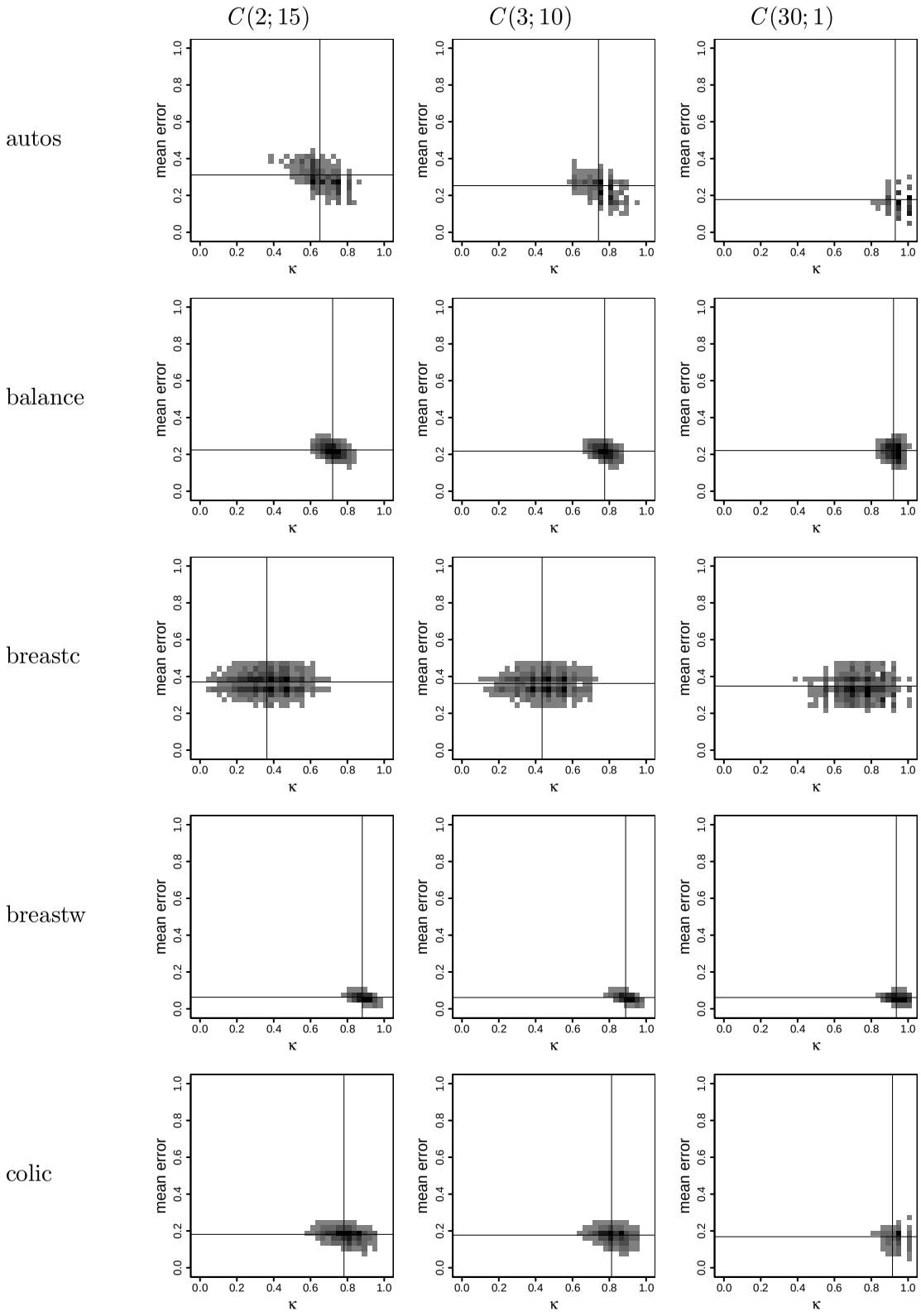

Figure D.2: κ-Error Diagrams for $C(2;15)$, $C(3;10)$, and $C(30;1)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

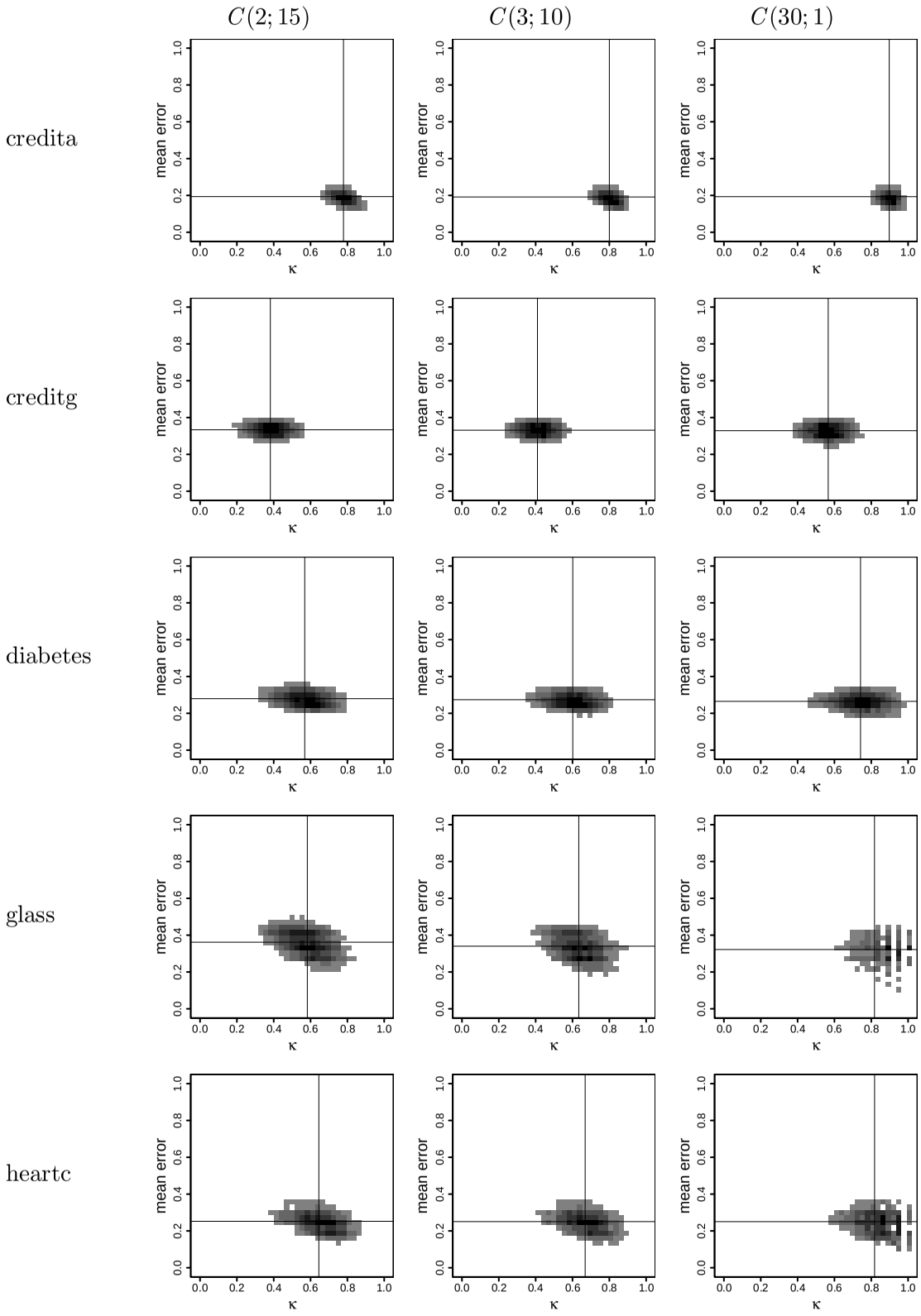

Figure D.2: κ-Error Diagrams for $C(2;15)$, $C(3;10)$, and $C(30;1)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

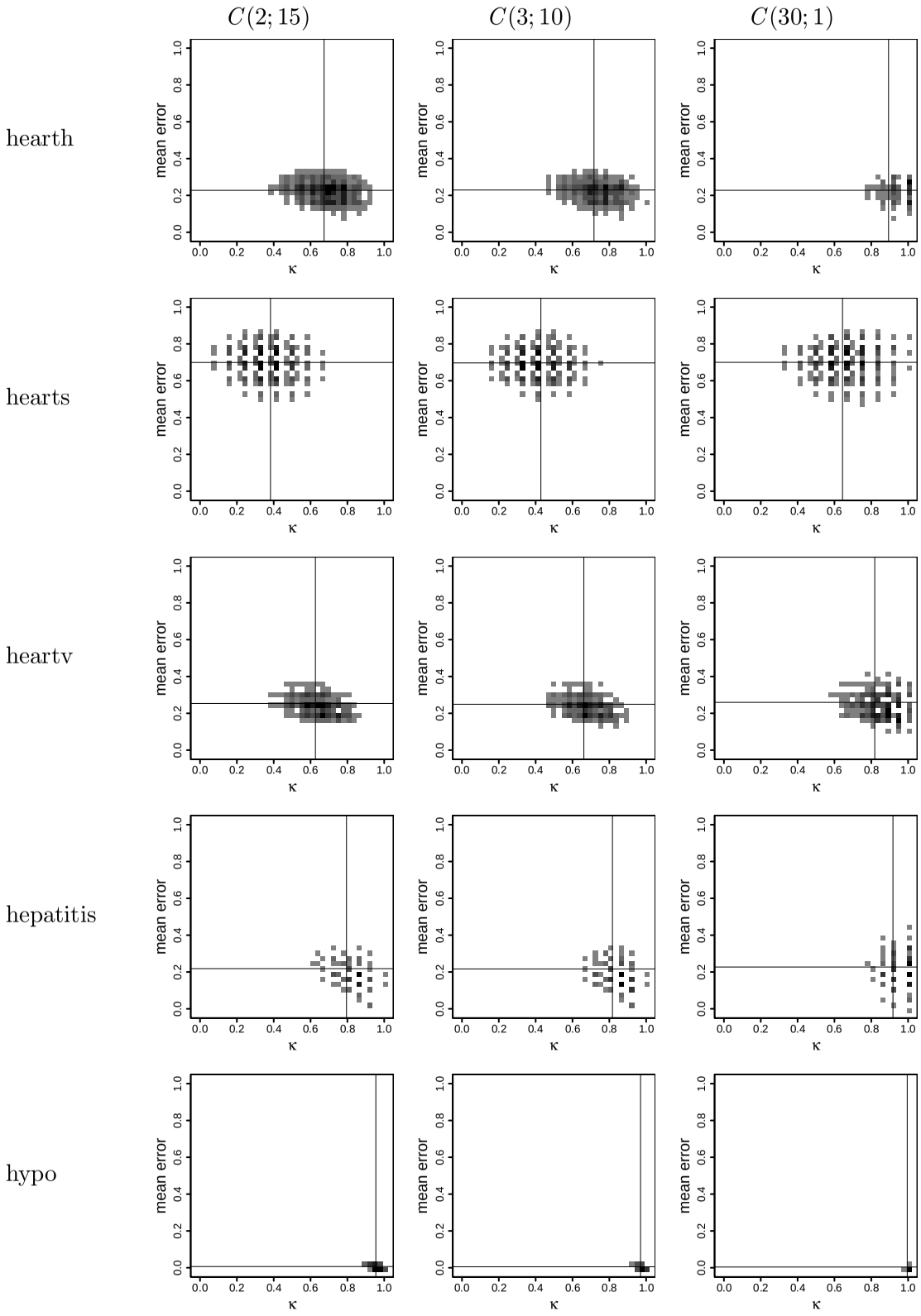

Figure D.2: κ-Error Diagrams for $C(2;15)$, $C(3;10)$, and $C(30;1)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

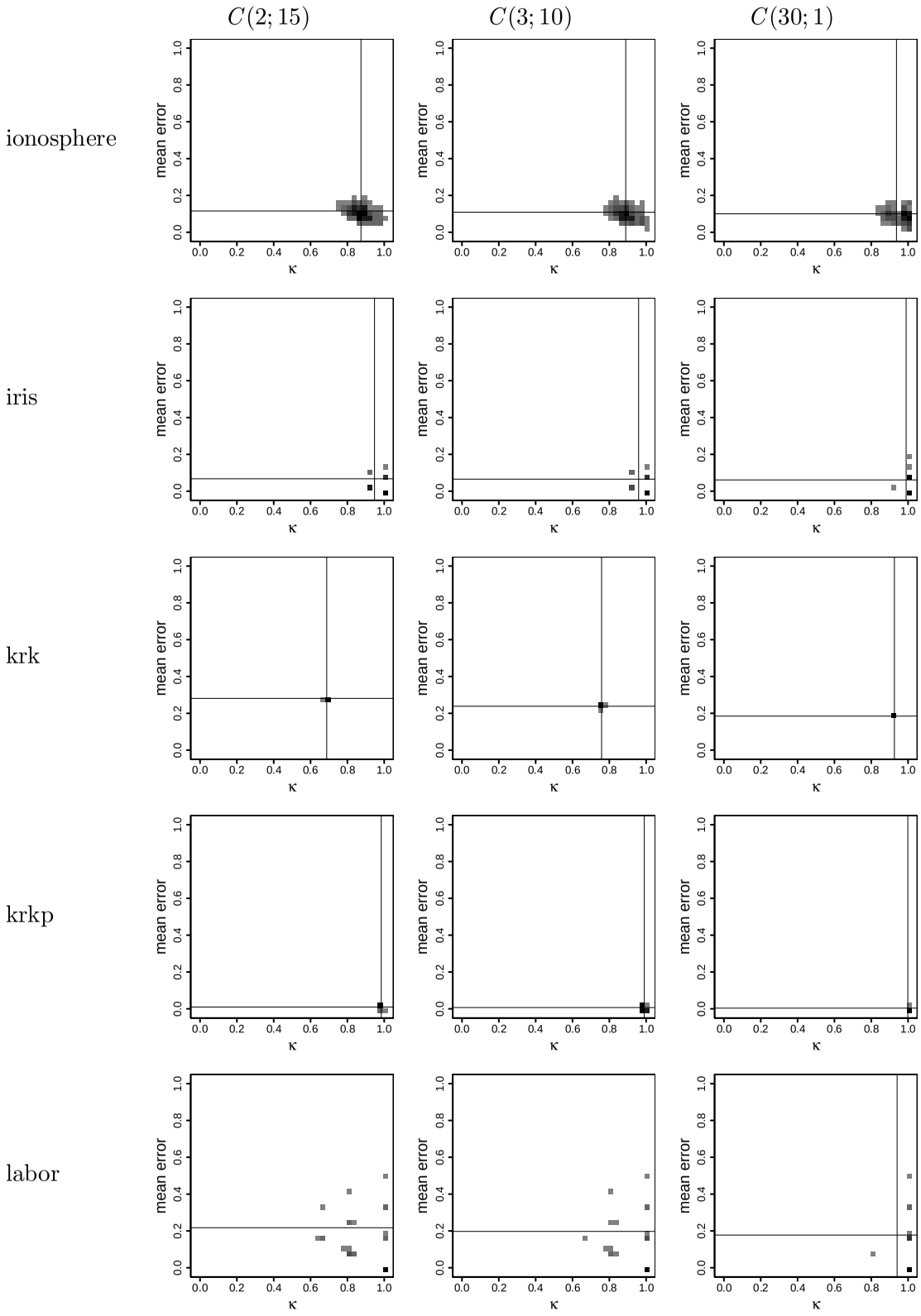

Figure D.2: κ-Error Diagrams for $C(2;15)$, $C(3;10)$, and $C(30;1)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

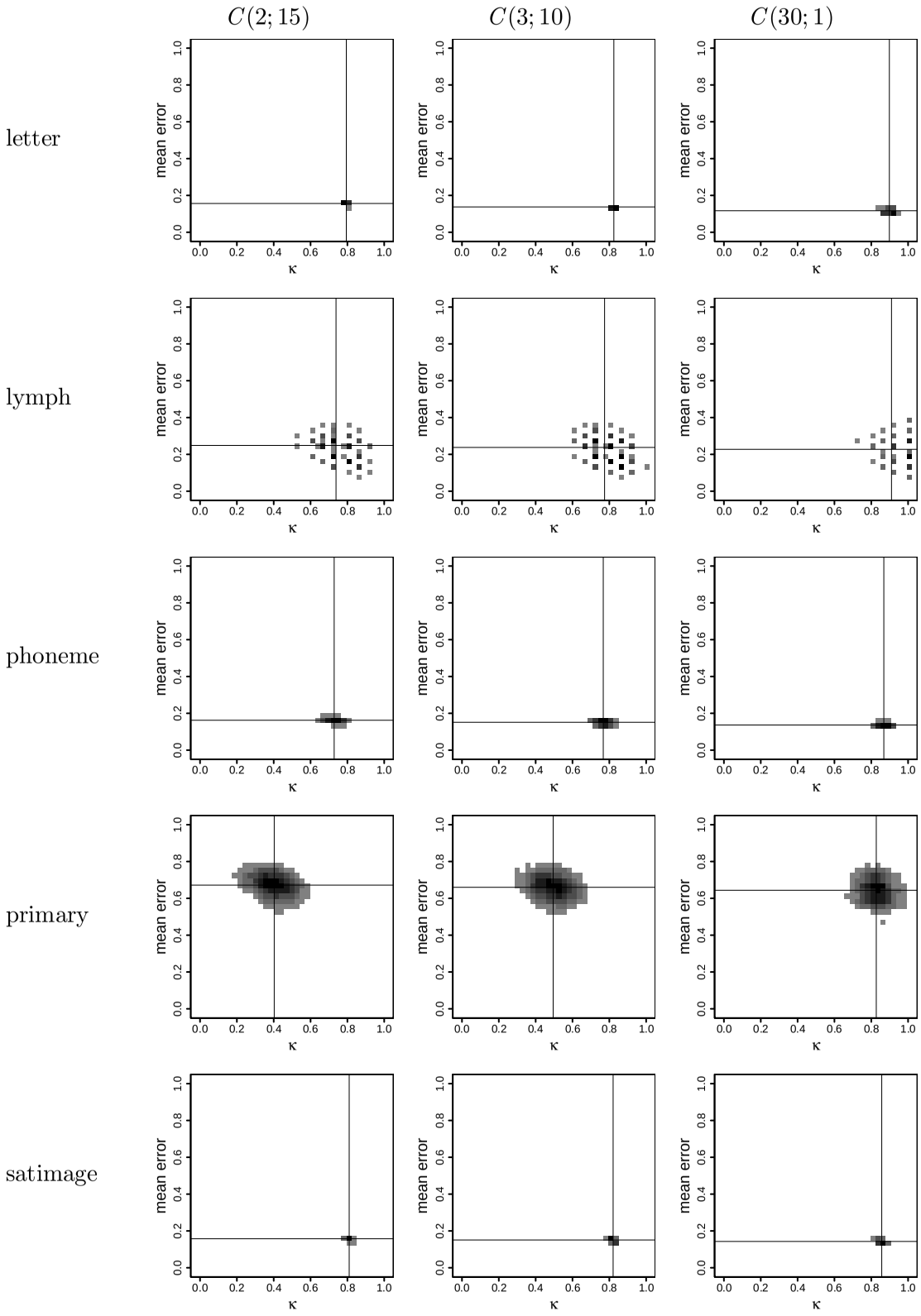

Figure D.2: κ-Error Diagrams for $C(2;15)$, $C(3;10)$, and $C(30;1)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

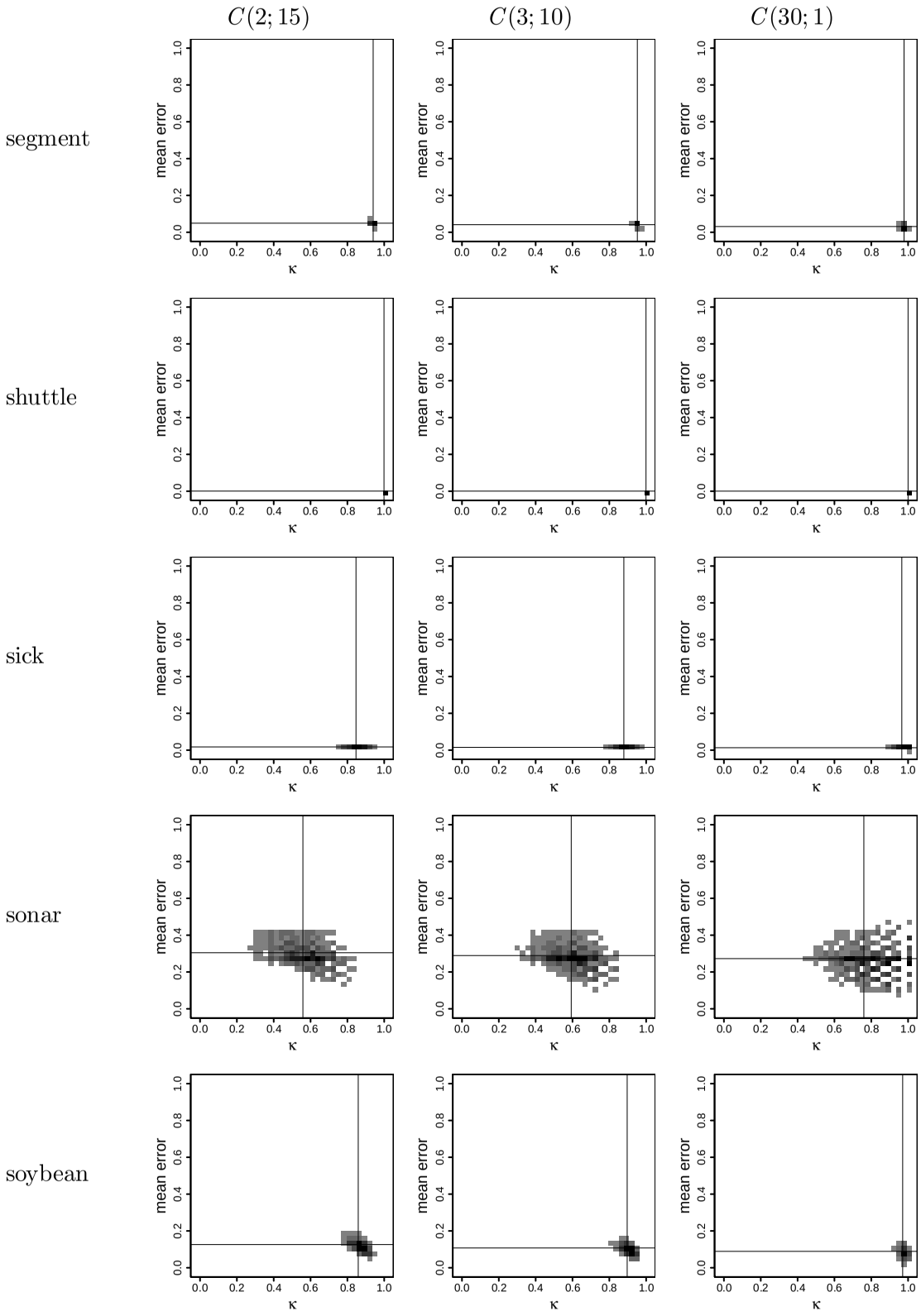

Figure D.2: κ-Error Diagrams for $C(2;15)$, $C(3;10)$, and $C(30;1)$. Pairs of classifiers in the lower left corner are more accurate and more diverse. *(continued on next page)*

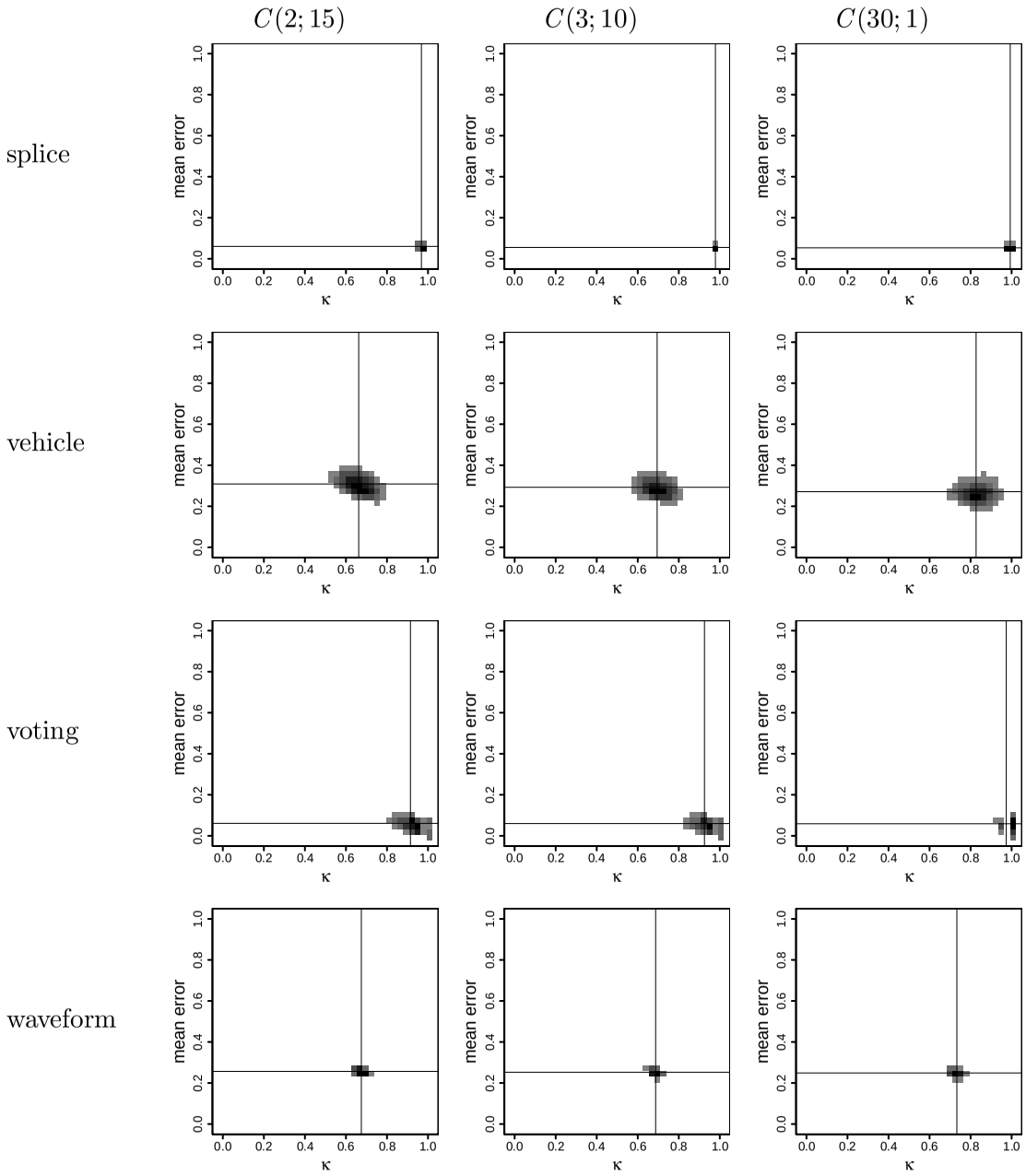

Figure D.2: κ-Error Diagrams for $C(2;15)$, $C(3;10)$, and $C(30;1)$. Pairs of classifiers in the lower left corner are more accurate and more diverse.

E. Cumulative Margin Distributions

Cumulative Margin Distributions for $Bagging(0.5; 30)$

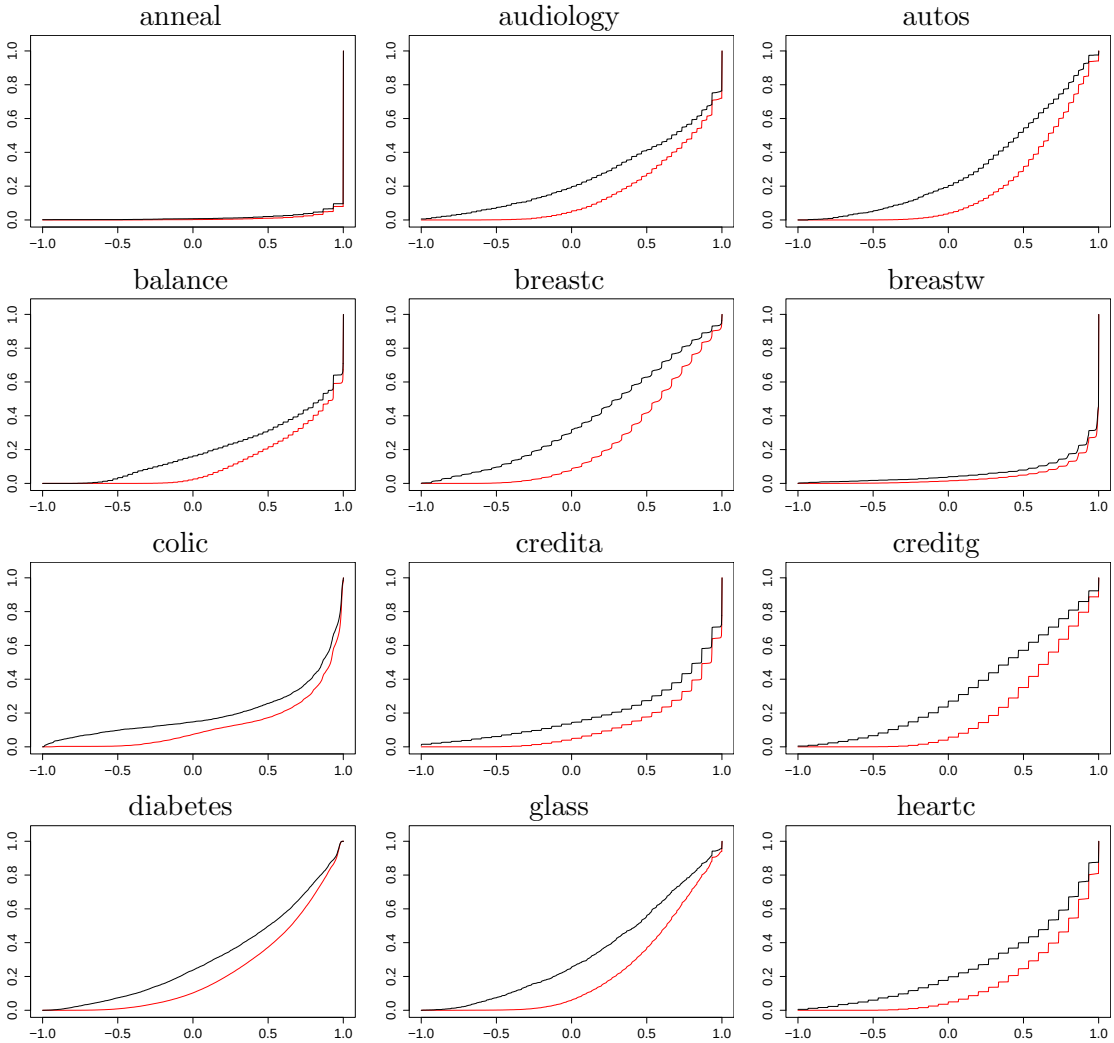

Figure E.1: Cumulative margin distributions on training (—) and test (—) data for $\mathcal{B}(0.5; 30)$ *(continued on next page)*

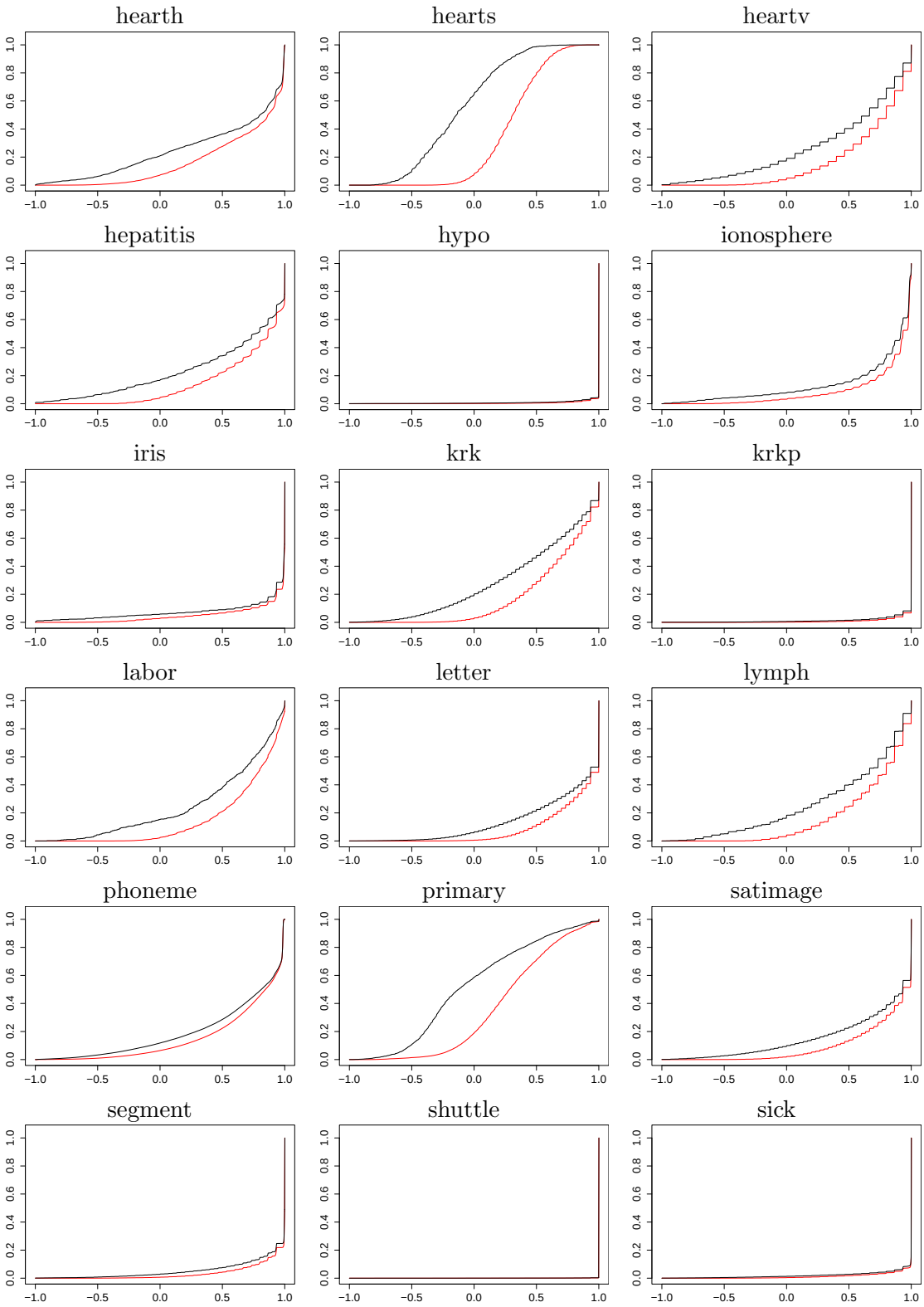

Figure E.1: Cumulative margin distributions on training (—) and test (—) data for $\mathcal{B}(0.5; 30)$ *(continued on next page)*

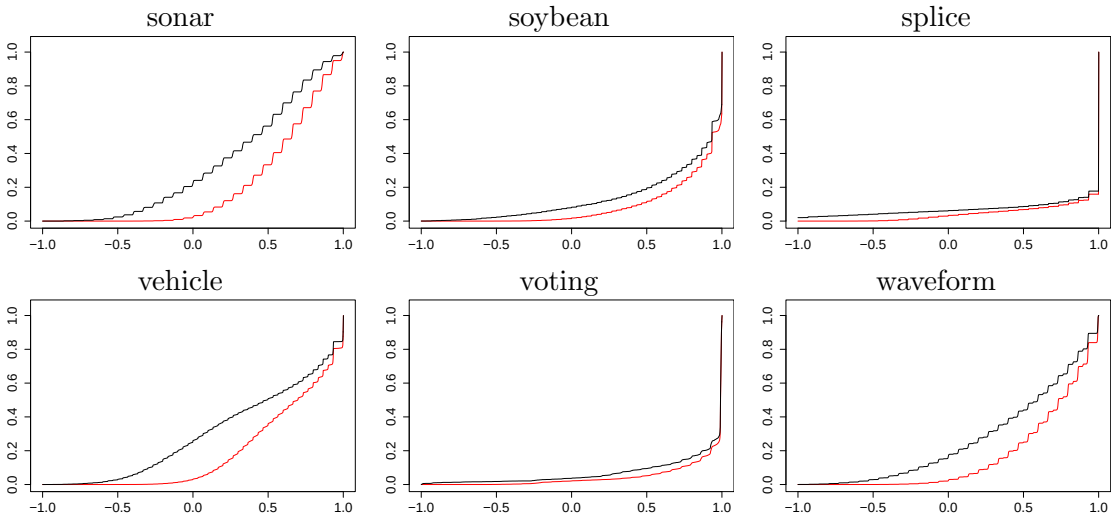

Figure E.1: Cumulative margin distributions on training (—) and test (—) data for $\mathcal{B}(0.5; 30)$

Cumulative Margin Distributions for *Bagging*$(1;30)$

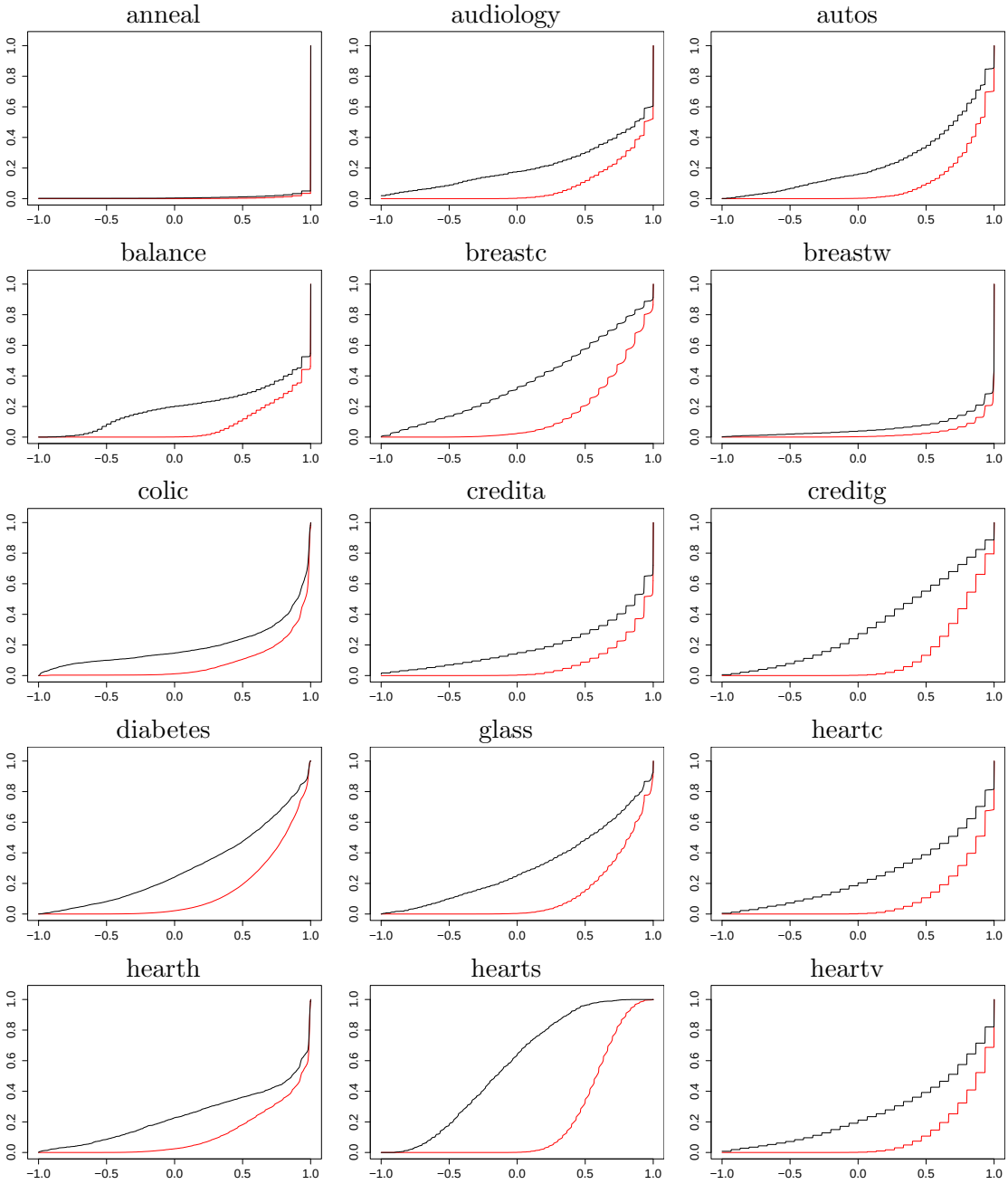

Figure E.2: Cumulative margin distributions on training (—) and test (—) data for $\mathcal{B}(1;30)$ *(continued on next page)*

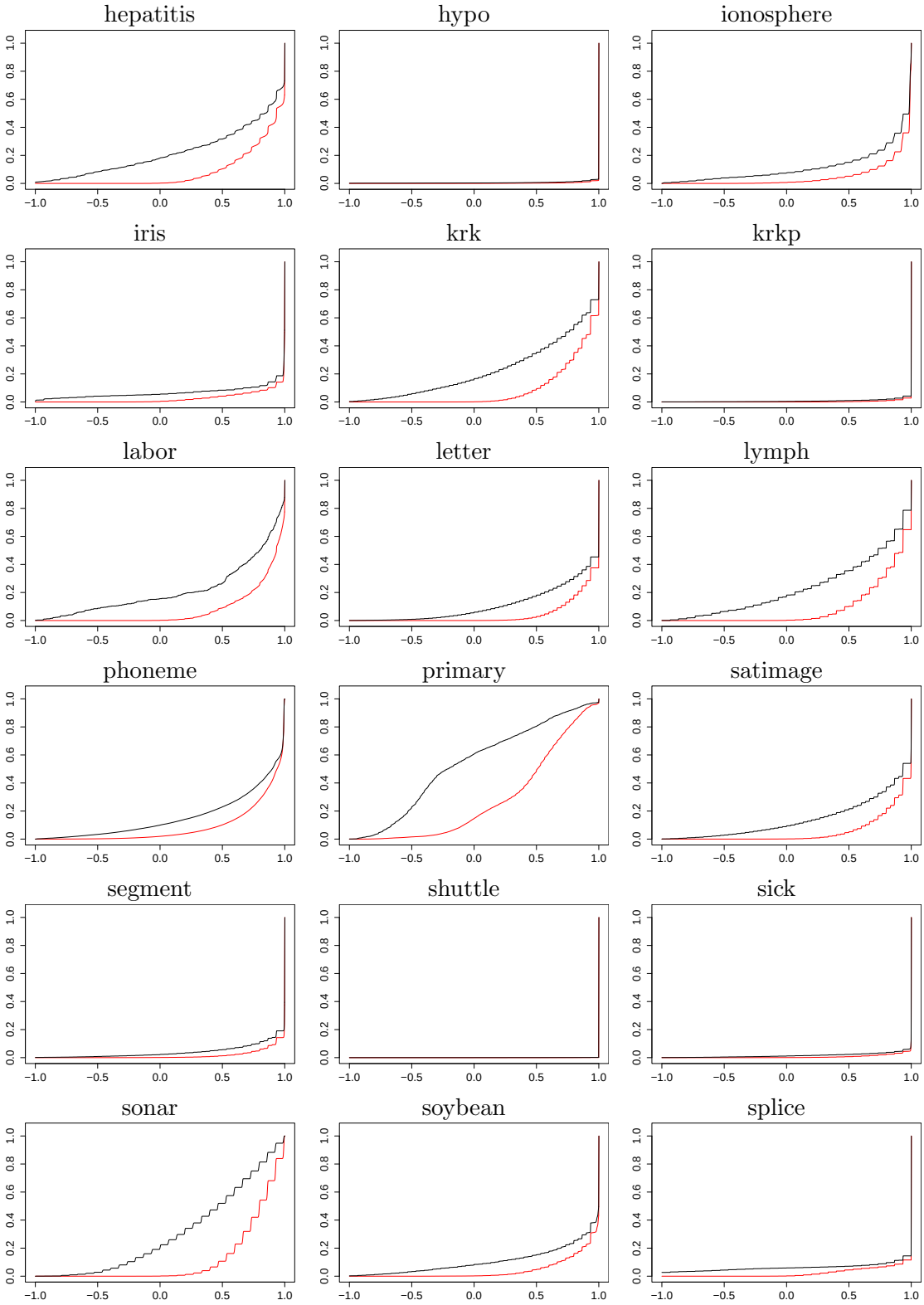

Figure E.2: Cumulative margin distributions on training (—) and test (—) data for $\mathcal{B}(1;30)$ *(continued on next page)*

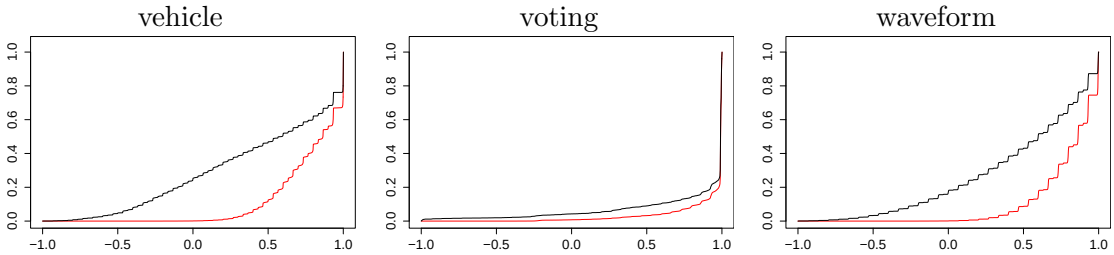

Figure E.2: Cumulative margin distributions on training (—) and test (—) data for $\mathcal{B}(1; 30)$

Cumulative Margin Distributions for $Bagging(2;30)$

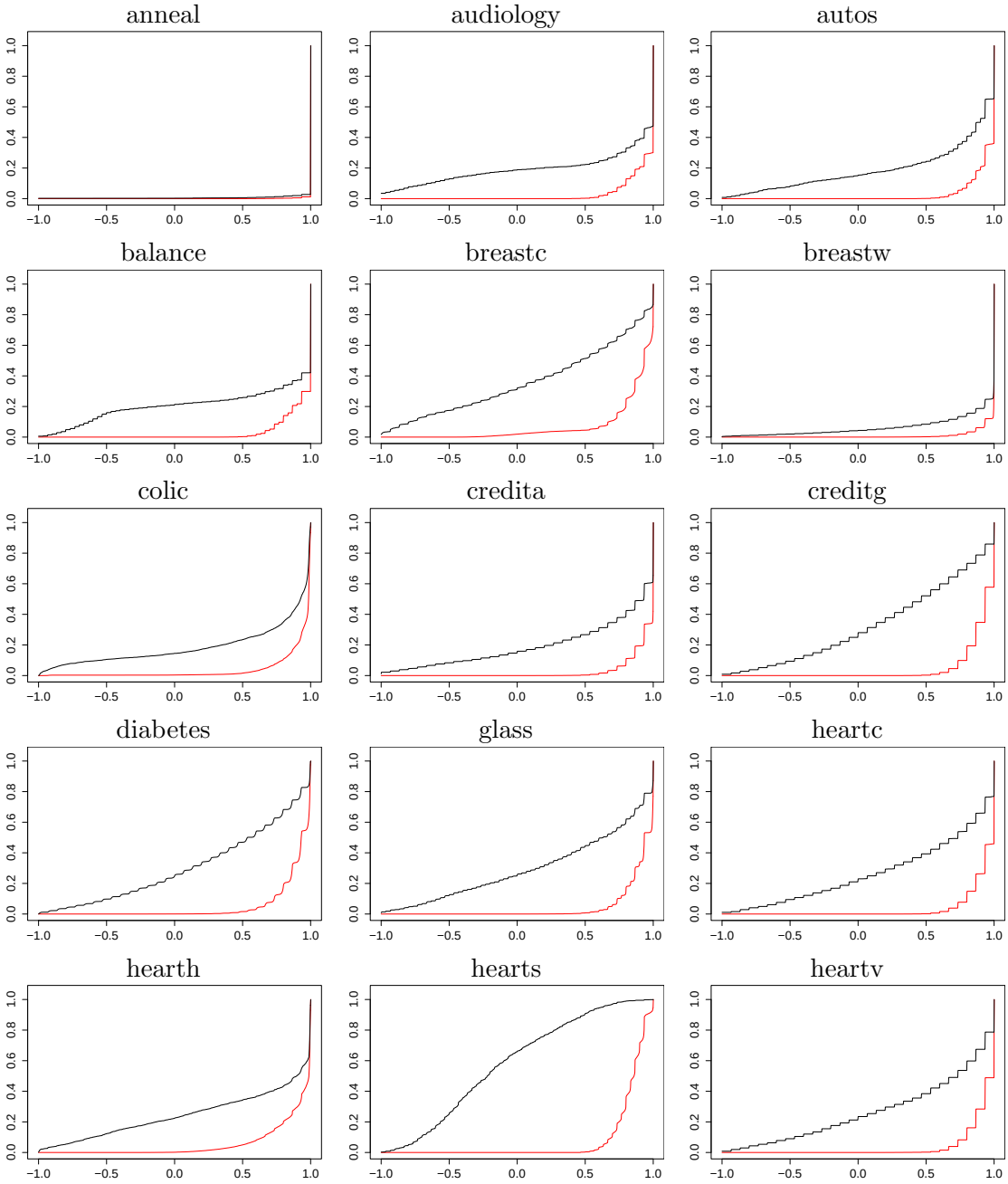

Figure E.3: Cumulative margin distributions on training (—) and test (—) data for $\mathcal{B}(2;30)$ *(continued on next page)*

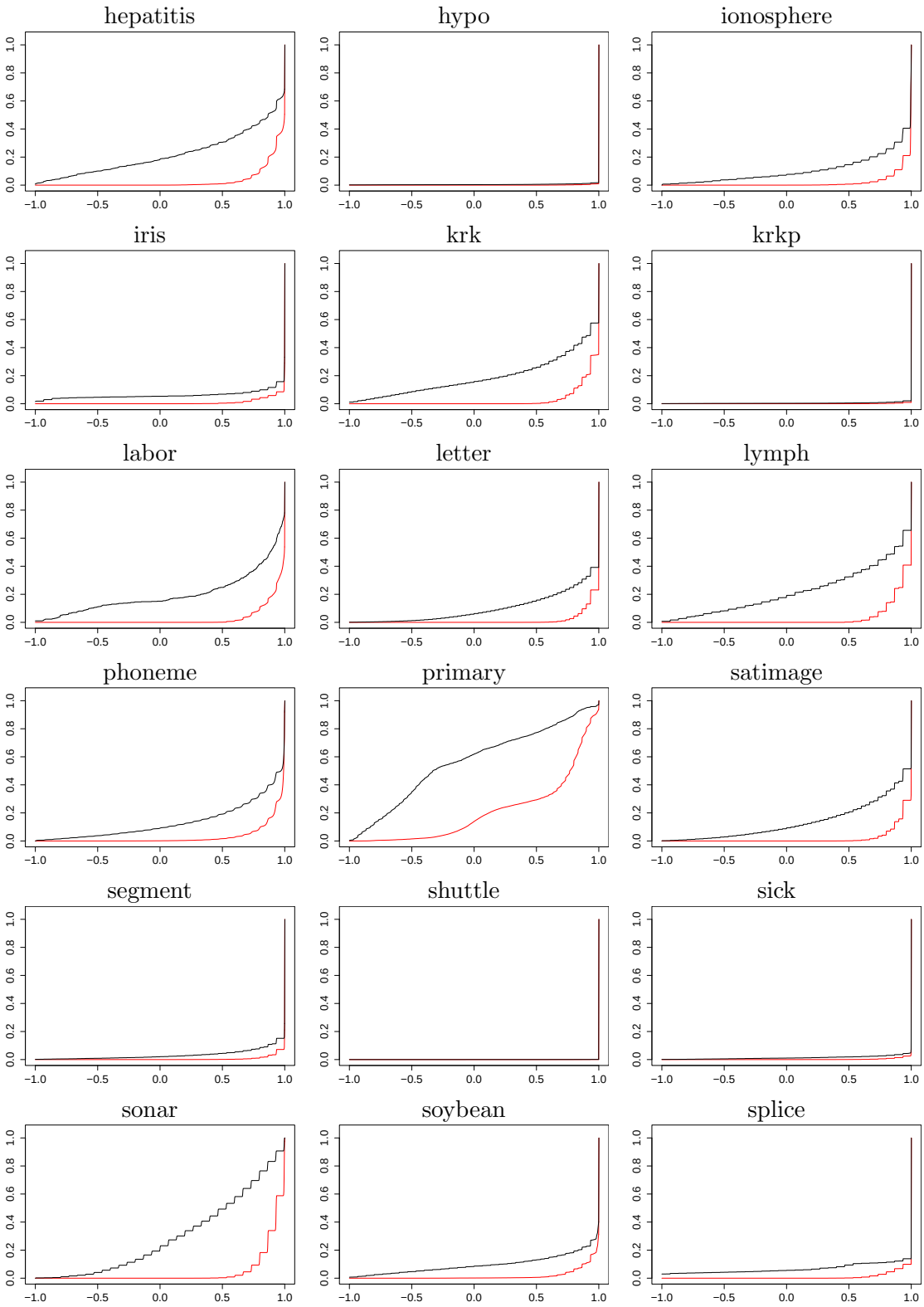

Figure E.3: Cumulative margin distributions on training (—) and test (—) data for $\mathcal{B}(2; 30)$ *(continued on next page)*

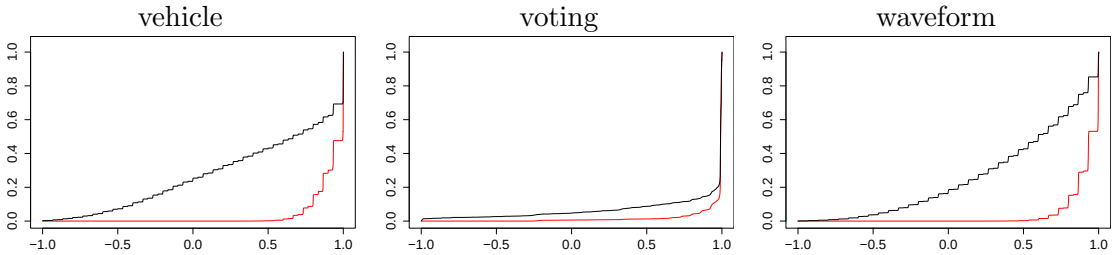

Figure E.3: Cumulative margin distributions on training (—) and test (—) data for $\mathcal{B}(2; 30)$

Cumulative Margin Distributions for *Cragging*$(2; 15)$

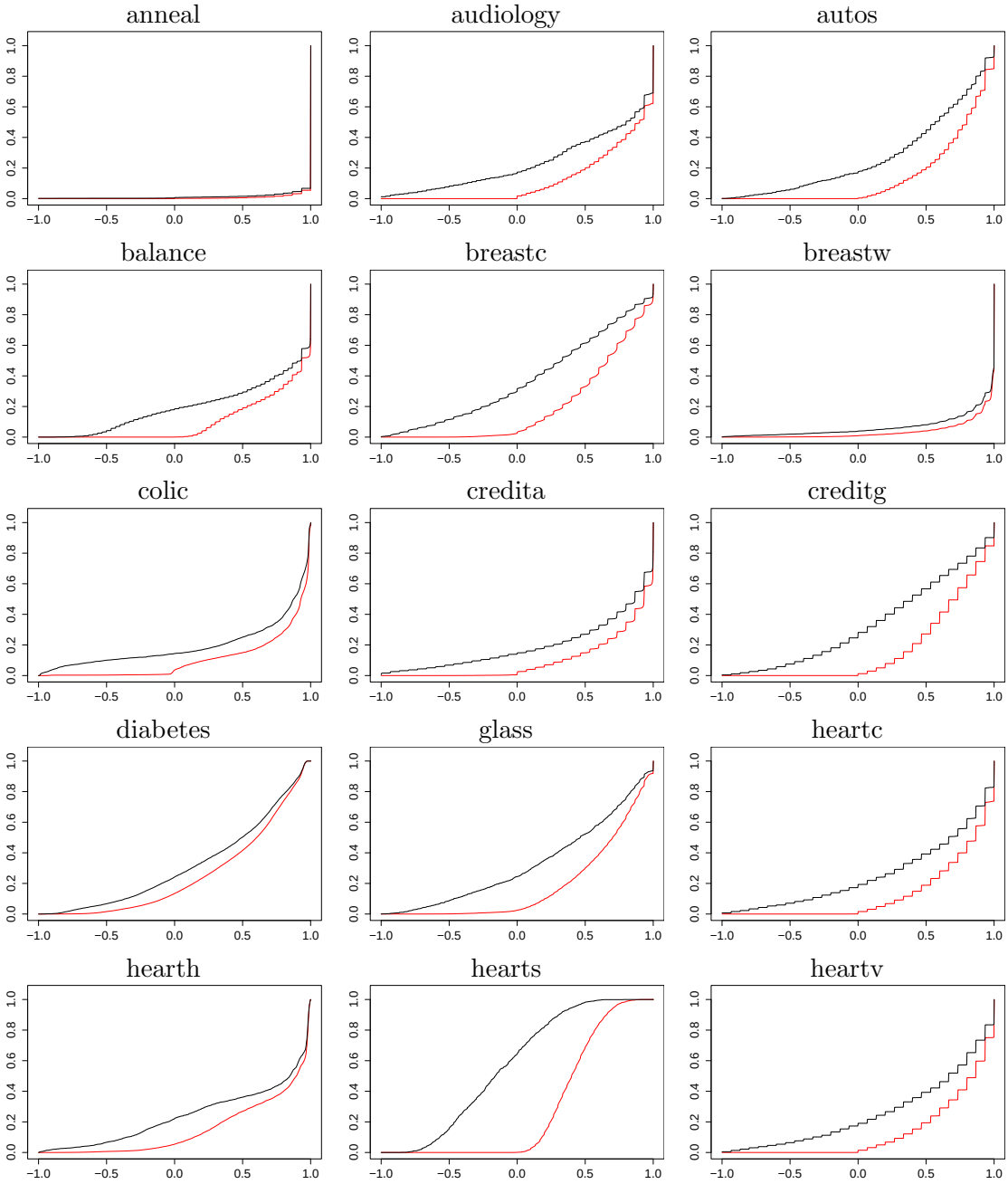

Figure E.4: Cumulative margin distributions on training (—) and test (—) data for $C(2; 15)$ *(continued on next page)*

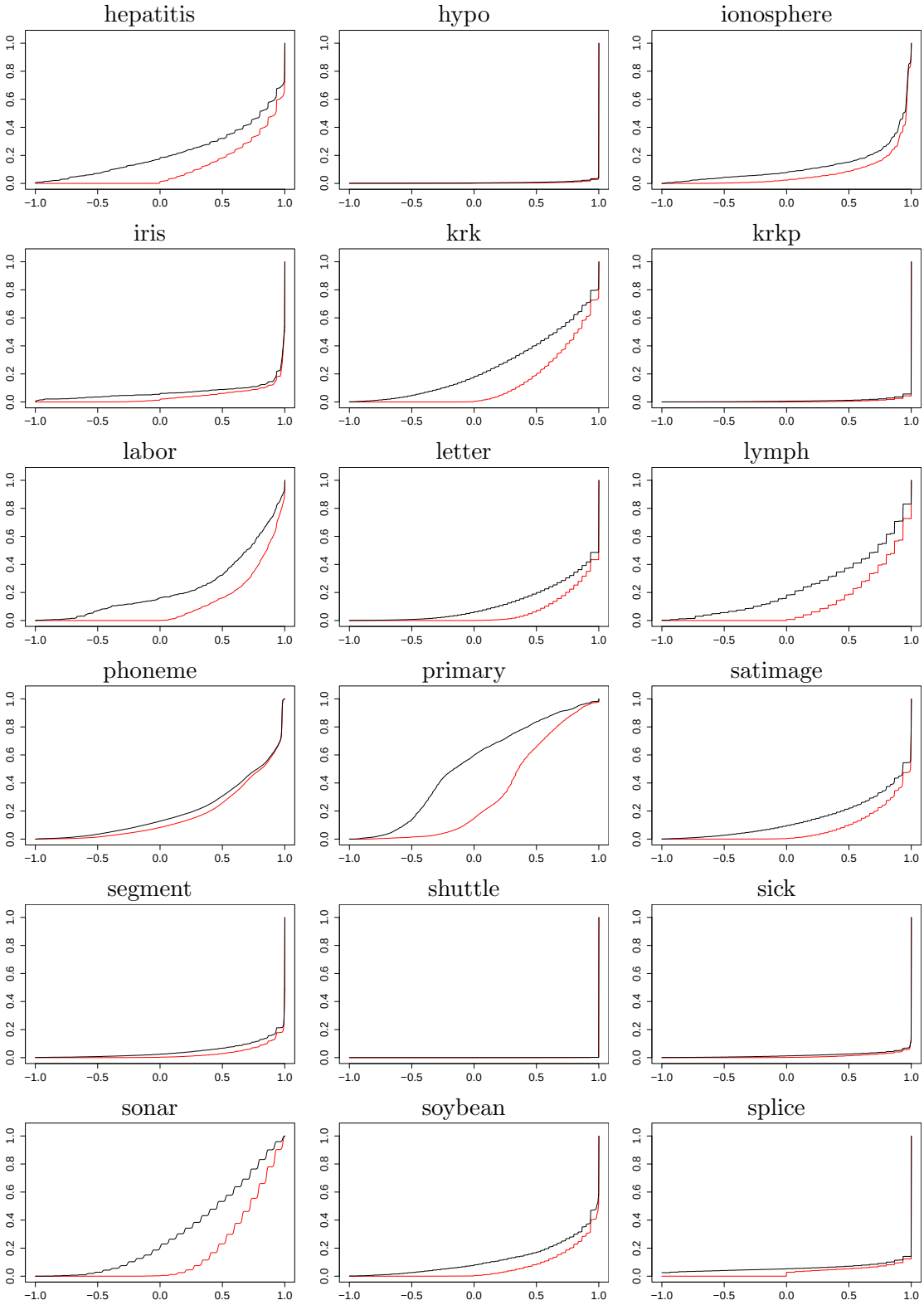

Figure E.4: Cumulative margin distributions on training (—) and test (—) data for $C(2;15)$ *(continued on next page)*

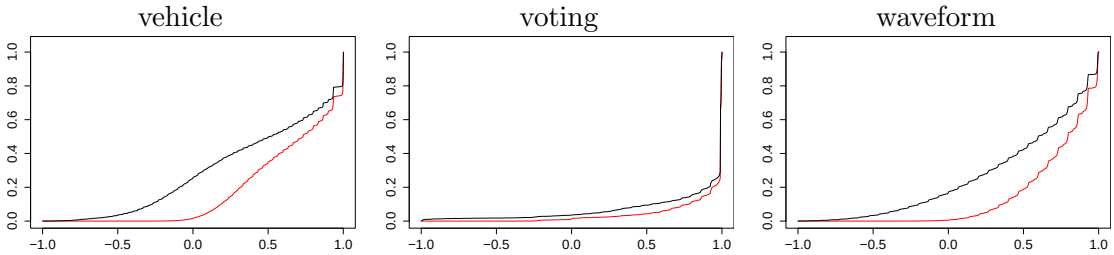

Figure E.4: Cumulative margin distributions on training (—) and test (—) data for $C(2; 15)$

Cumulative Margin Distributions for *Cragging*$(3; 10)$

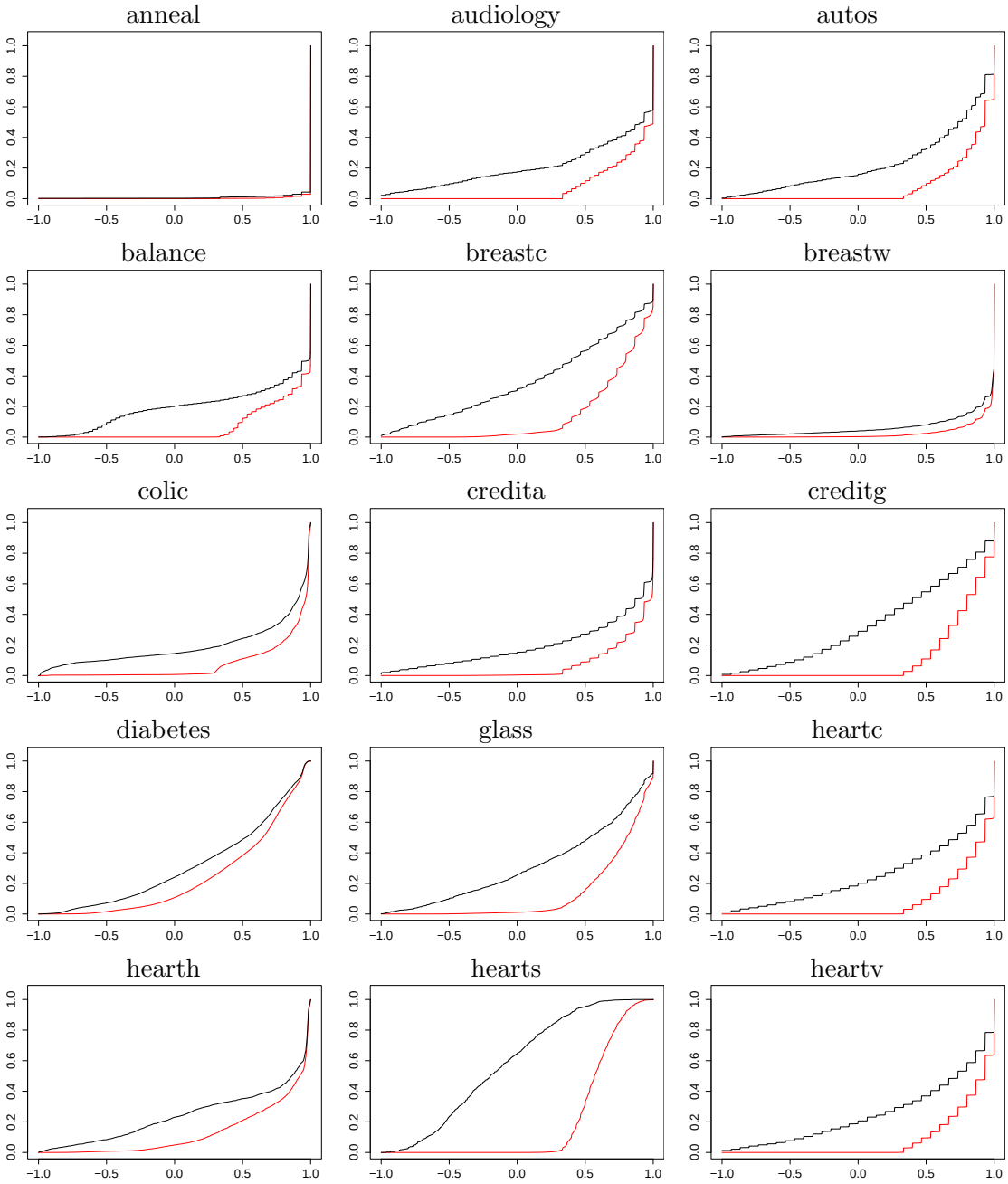

Figure E.5: Cumulative margin distributions on training (—) and test (—) data for $C(3; 10)$ *(continued on next page)*

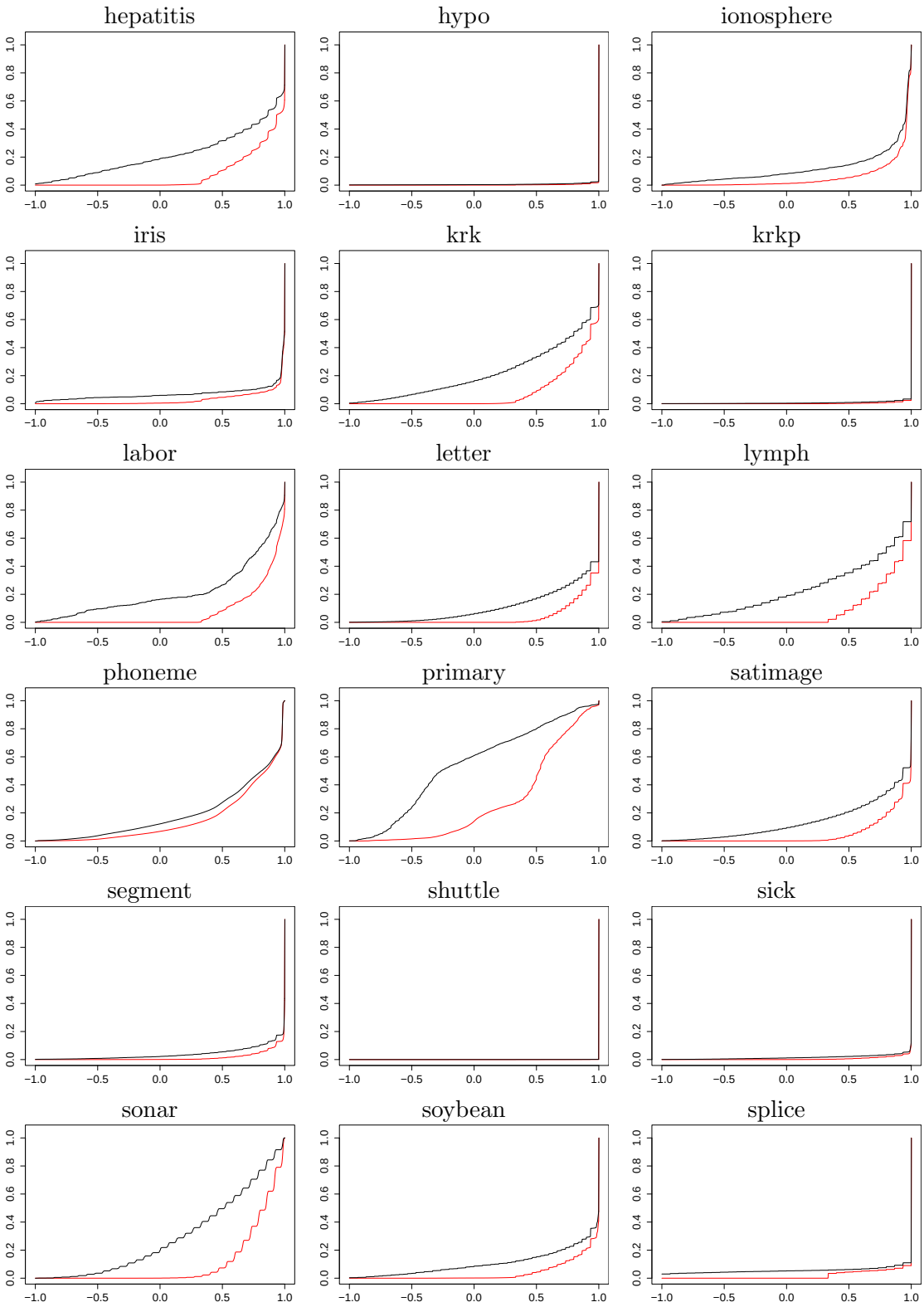

Figure E.5: Cumulative margin distributions on training (—) and test (—) data for $C(3;10)$ *(continued on next page)*

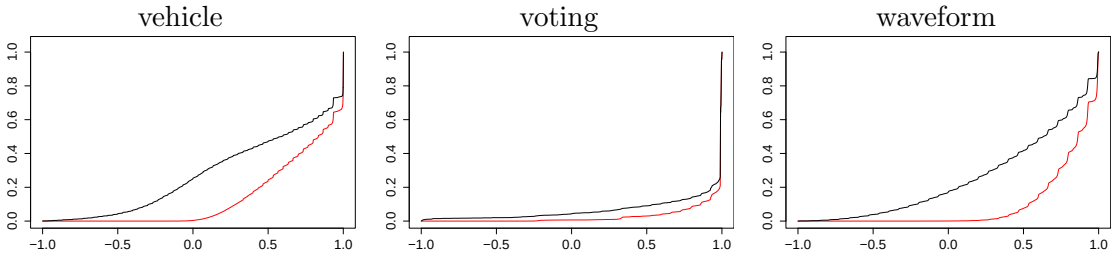

Figure E.5: Cumulative margin distributions on training (—) and test (—) data for $C(3; 10)$

Cumulative Margin Distributions for *Cragging*$(30; 1)$

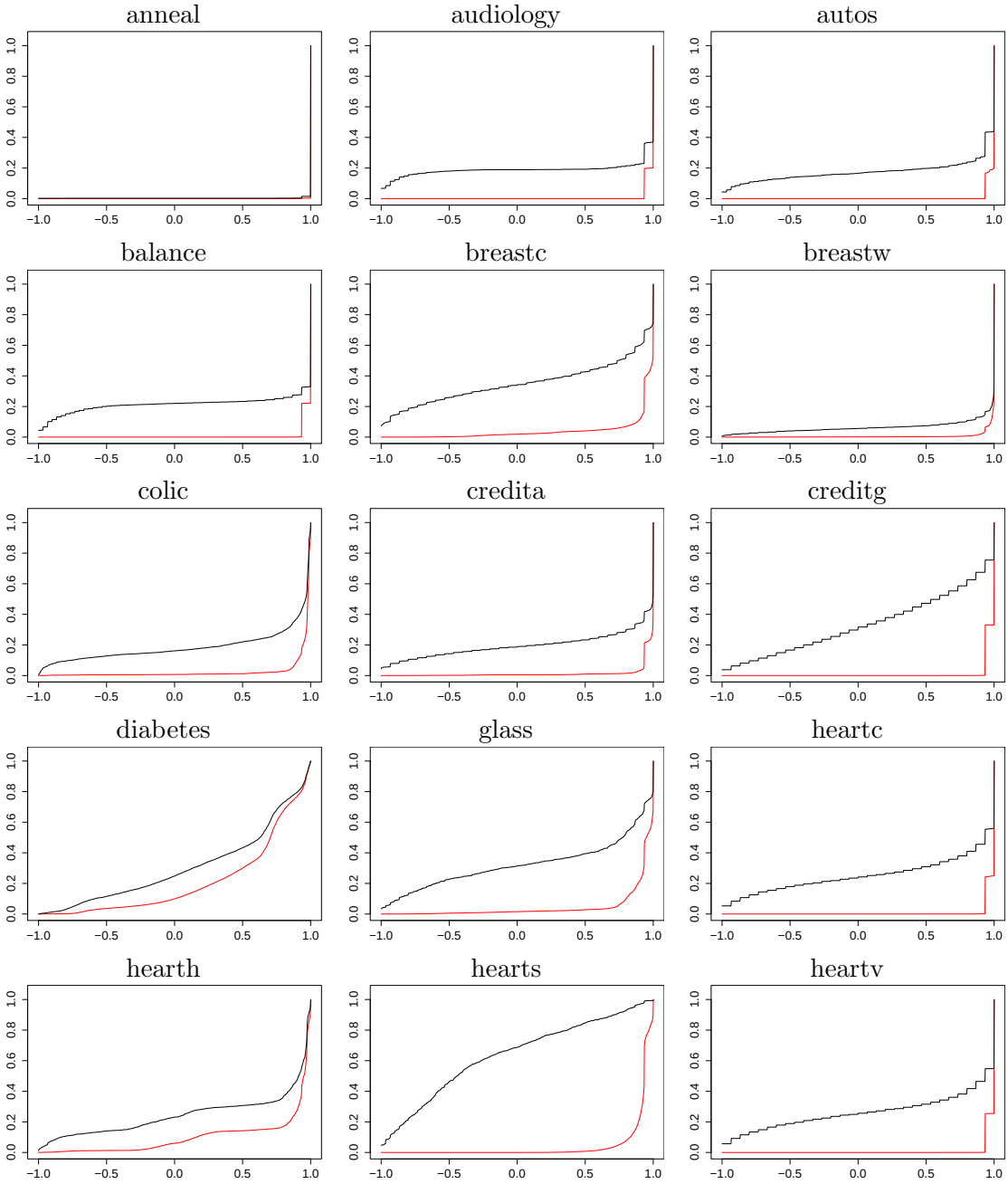

Figure E.6: Cumulative margin distributions on training (—) and test (—) data for $C(30; 1)$ *(continued on next page)*

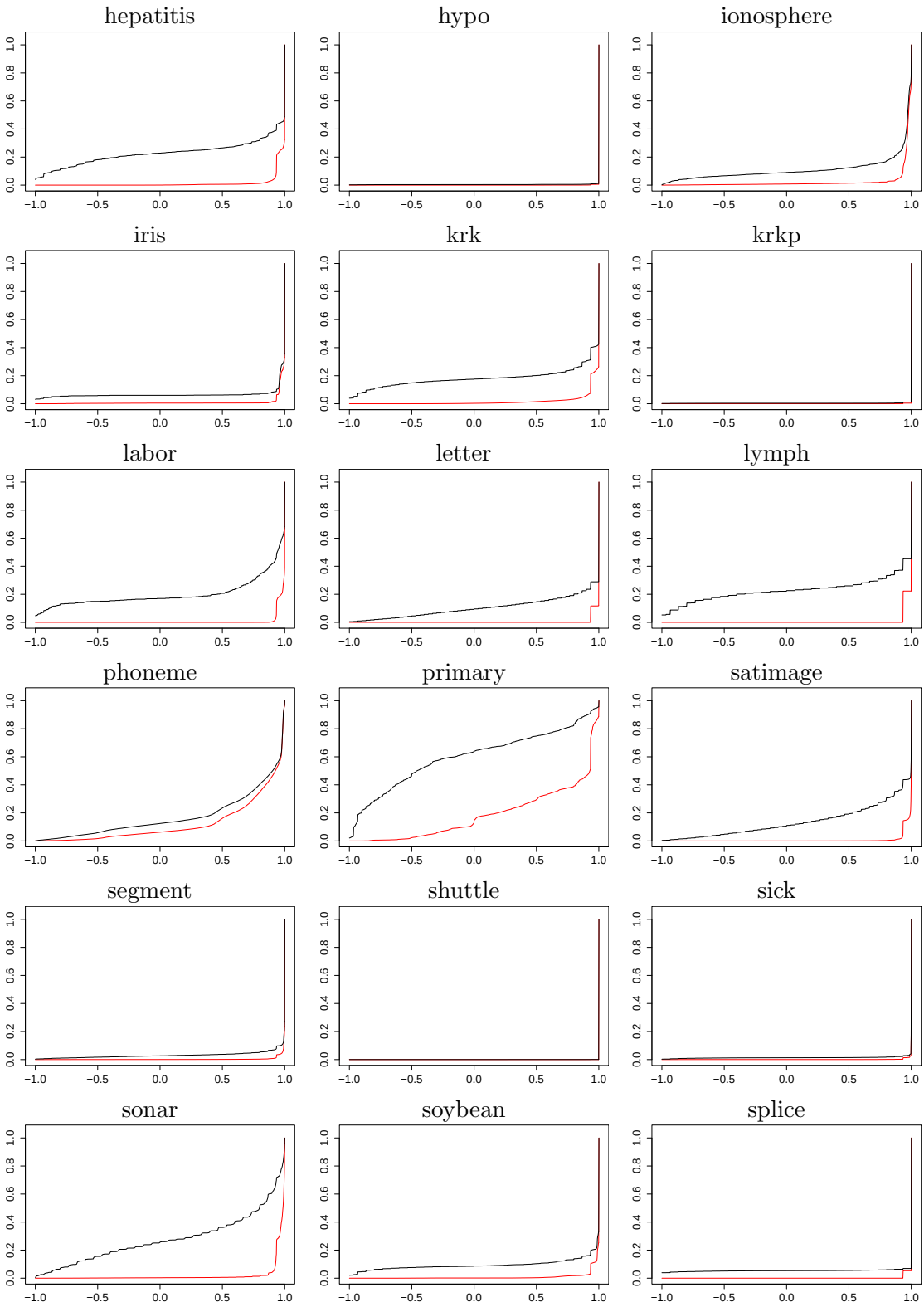

Figure E.6: Cumulative margin distributions on training (—) and test (—) data for $C(30; 1)$ *(continued on next page)*

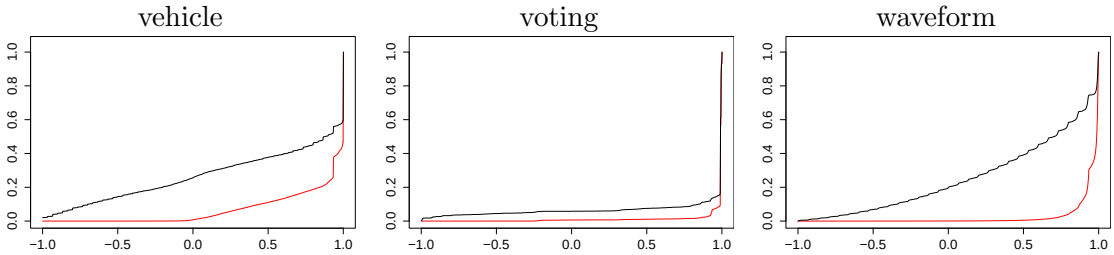

Figure E.6: Cumulative margin distributions on training (—) and test (—) data for $C(30; 1)$

Cumulative Margin Distributions for the Base Classifier

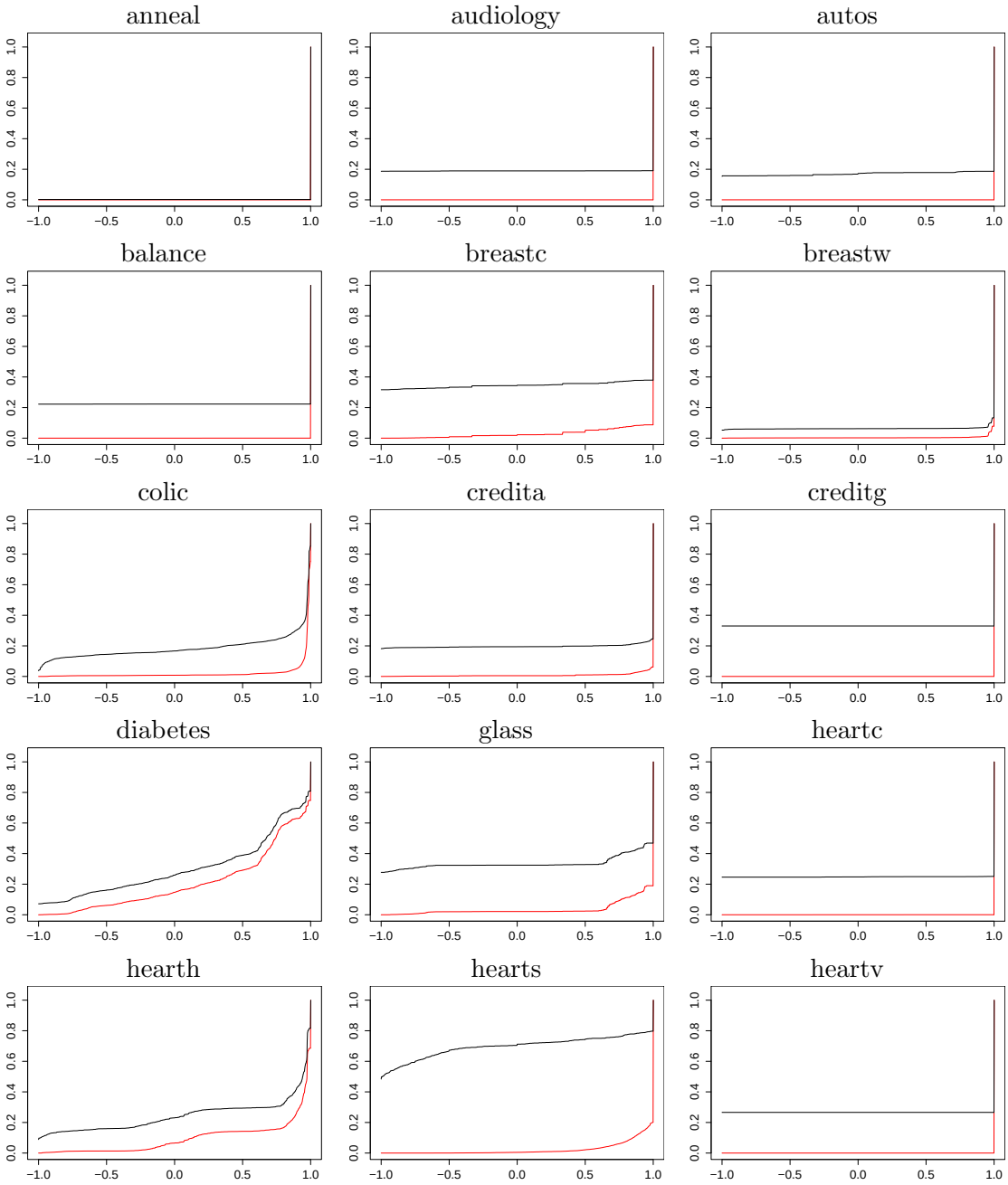

Figure E.7: Cumulative margin distributions on training (—) and test (—) data for the base classifier *(continued on next page)*

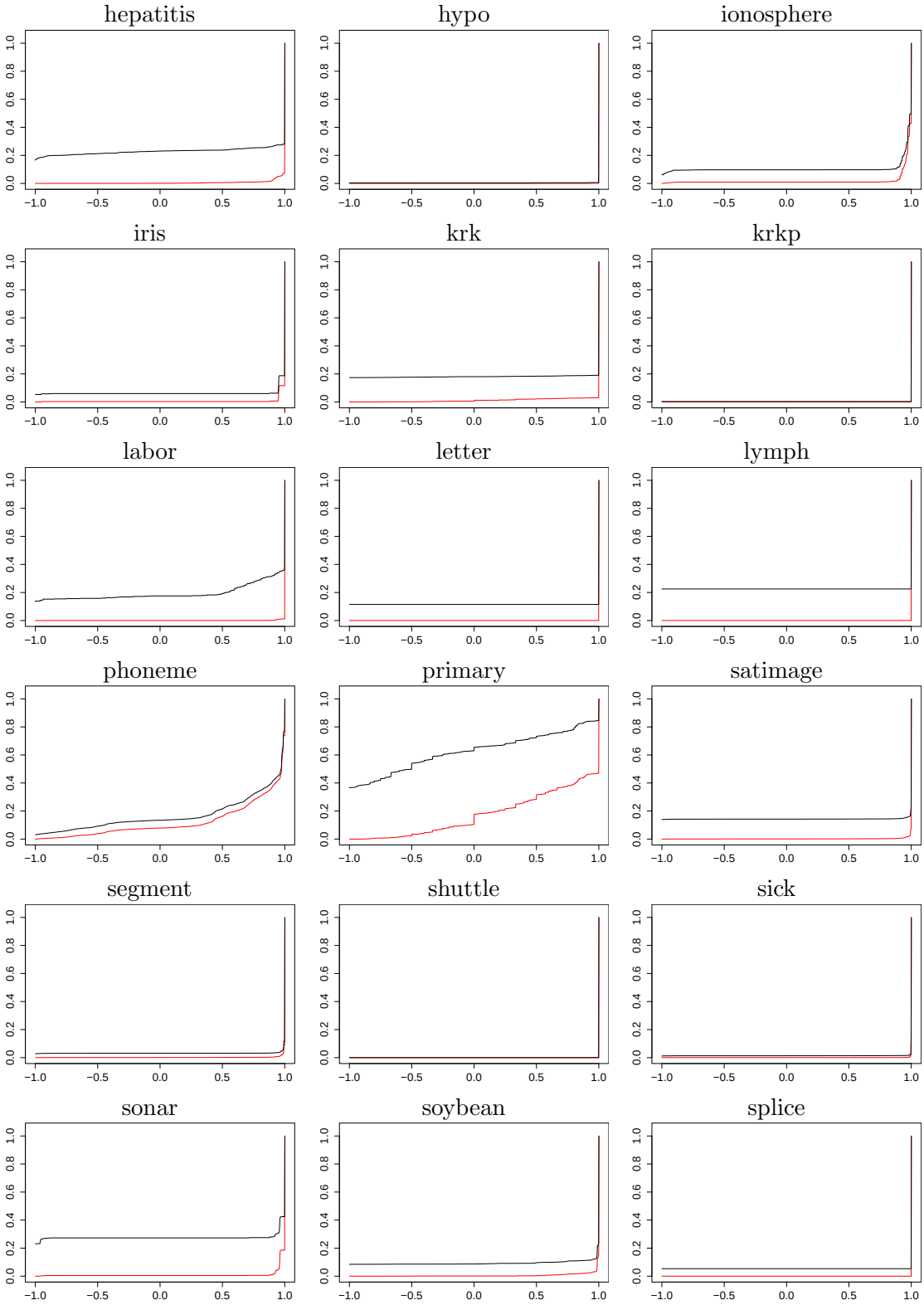

Figure E.7: Cumulative margin distributions on training (—) and test (—) data for the base classifier *(continued on next page)*

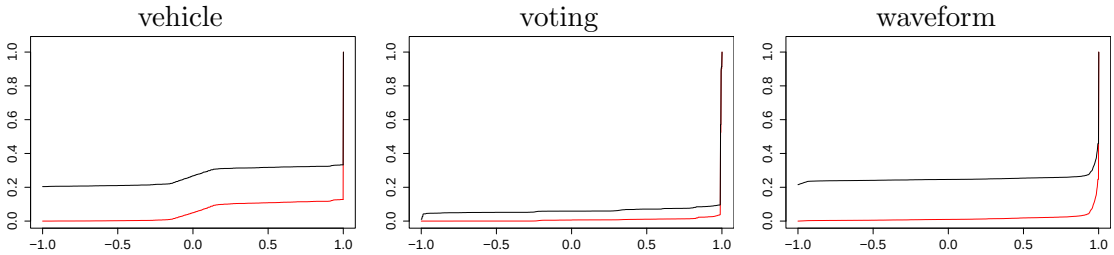

Figure E.7: Cumulative margin distributions on training (—) and test (—) data for the base classifier

F. Bias-Variance Decomposition Results

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	0.79±0.28	0.33±0.27	0.45±0.25	0.54±0.19	0.09±0.09
audiology	24.12±4.10	19.45±4.61	4.67±1.53	7.08±1.64	2.41±0.75
autos	24.27±3.69	16.05±3.84	8.22±2.13	10.14±1.81	1.92±0.64
balance	19.46±1.86	17.42±1.93	2.03±0.66	4.92±0.70	2.89±0.38
breastc	33.55±3.74	30.83±5.30	2.72±2.48	9.15±1.69	6.43±1.15
breastw	4.53±1.04	4.29±1.12	0.24±0.32	1.01±0.20	0.77±0.30
colic	15.61±2.05	14.40±2.30	1.21±1.12	2.88±0.70	1.67±0.62
credita	15.23±1.99	13.48±2.65	1.75±1.05	4.02±0.64	2.27±0.51
creditg	26.98±1.70	23.30±1.84	3.68±0.70	8.26±0.57	4.58±0.35
diabetes	25.72±1.74	23.44±1.86	2.28±0.92	6.25±0.66	3.96±0.39
glass	29.54±4.37	24.68±3.78	4.87±1.58	9.44±1.07	4.58±1.25
heartc	20.90±3.38	18.52±3.81	2.38±1.20	6.33±1.04	3.94±0.80
hearth	21.60±2.83	21.78±2.60	-1.81±1.66	4.60±1.39	4.79±1.04
hearts	65.26±3.74	62.50±6.50	2.76±3.85	12.14±2.64	9.38±1.81
heartv	21.22±2.89	18.89±3.54	2.33±1.79	6.43±0.93	4.10±1.24
hepatitis	18.65±4.27	17.33±4.53	1.32±1.46	4.77±1.17	3.45±1.24
hypo	0.46±0.15	0.34±0.17	0.11±0.07	0.17±0.05	0.05±0.04
ionosphere	8.17±2.27	6.56±2.34	1.62±0.61	2.28±0.59	0.66±0.33
iris	6.04±3.31	5.33±3.44	0.71±0.67	1.00±0.54	0.29±0.23
krk	22.60±0.20	16.58±0.32	6.02±0.24	9.46±0.17	3.44±0.13
krkp	0.73±0.14	0.44±0.18	0.29±0.14	0.42±0.10	0.13±0.06
labor	15.76±4.43	14.00±6.70	1.76±2.56	4.80±1.30	3.04±1.64
letter	7.92±0.26	5.19±0.26	2.73±0.17	3.70±0.16	0.97±0.05
lymph	20.19±3.28	16.81±4.18	3.38±1.90	7.42±1.09	4.03±0.98
phoneme	11.27±0.56	9.81±0.69	1.46±0.29	3.44±0.17	1.98±0.21
primary	63.13±3.26	56.93±3.86	6.19±1.50	10.14±1.45	3.95±0.57
satimage	10.04±0.65	8.78±0.57	1.26±0.18	2.65±0.18	1.39±0.07
segment	3.43±0.39	2.64±0.44	0.79±0.23	1.41±0.21	0.62±0.13
shuttle	0.04±0.01	0.03±0.01	0.01±0.01	0.02±0.01	0.01±0.00
sick	1.47±0.25	1.22±0.25	0.25±0.11	0.54±0.11	0.29±0.09
sonar	24.75±2.72	21.12±4.75	3.63±2.64	9.63±1.61	6.00±1.70
soybean	9.81±2.03	8.05±2.80	1.75±1.18	3.39±0.70	1.63±0.74
splice	6.56±0.41	5.89±0.40	0.67±0.31	1.24±0.30	0.57±0.12
vehicle	27.00±2.20	25.55±2.96	1.45±1.44	7.58±0.97	6.13±0.73
voting	5.10±1.82	3.22±1.55	1.88±0.52	2.10±0.65	0.21±0.20
waveform	17.84±1.07	15.60±1.26	2.24±0.32	5.58±0.24	3.34±0.23
Mean	17.49	15.30	2.19	4.86	2.67

Table F.1: Results of bias-variance decomposition for $Bagging(1;30)$: absolute values.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	0.93	1.50	0.73	0.79	1.53
audiology	0.97	0.96	1.02	0.93	0.79
autos	0.90	1.06	0.69	0.71	0.83
balance	0.89	0.91	0.77	0.80	0.82
breastc	0.92	0.98	0.56	0.68	0.75
breastw	0.71	0.97	0.12	0.32	0.64
colic	0.87	0.96	0.41	0.52	0.65
credita	0.78	0.93	0.35	0.47	0.66
creditg	0.83	0.87	0.63	0.57	0.53
diabetes	0.87	1.01	0.37	0.49	0.59
glass	0.85	0.96	0.52	0.64	0.84
heartc	0.83	0.95	0.43	0.56	0.68
hearth	0.95	1.10	-1.94	0.60	1.00
hearts	0.95	1.07	0.27	0.61	1.00
heartv	0.82	0.91	0.44	0.57	0.69
hepatitis	0.83	0.90	0.43	0.60	0.71
hypo	0.83	1.00	0.54	0.64	1.02
ionosphere	0.76	0.92	0.44	0.43	0.41
iris	0.94	1.00	0.65	0.66	0.68
krk	0.91	1.02	0.70	0.78	0.95
krkp	0.93	1.27	0.66	0.79	1.42
labor	0.88	1.00	0.46	0.59	0.70
letter	0.56	0.89	0.32	0.39	0.85
lymph	0.82	0.86	0.66	0.66	0.67
phoneme	0.78	0.97	0.34	0.48	0.69
primary	0.95	0.97	0.78	0.83	0.92
satimage	0.65	0.92	0.21	0.32	0.57
segment	0.78	1.17	0.37	0.52	1.14
shuttle	1.04	1.13	0.86	0.92	1.06
sick	0.93	1.07	0.56	0.69	0.86
sonar	0.84	1.13	0.34	0.56	0.91
soybean	0.87	1.02	0.51	0.64	0.88
splice	1.06	1.01	1.88	1.17	0.81
vehicle	0.90	1.00	0.31	0.61	0.79
voting	0.81	0.78	0.87	0.77	0.37
waveform	0.69	0.92	0.25	0.37	0.56
Mean	0.86	1.00	0.54	0.63	0.80

Table F.2: Results of bias-variance decomposition for $Bagging(1;30)$: ratios relative to the base classifier.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	1.15±0.33	0.78±0.46	0.37±0.24	0.57±0.13	0.20±0.16
audiology	25.57±4.10	19.45±5.55	6.12±2.34	7.96±1.74	1.84±1.03
autos	27.21±3.15	18.52±3.24	8.68±1.74	11.14±1.49	2.46±0.65
balance	17.66±1.86	13.90±2.27	3.76±1.38	5.77±0.99	2.01±0.57
breastc	32.73±3.31	28.71±4.52	4.02±1.70	10.04±1.47	6.02±1.14
breastw	4.50±1.02	4.01±1.11	0.49±0.25	1.03±0.20	0.54±0.28
colic	15.35±1.94	13.87±2.61	1.47±0.93	2.74±0.57	1.26±0.50
credita	14.86±1.80	13.77±2.14	1.09±0.70	3.15±0.49	2.06±0.40
creditg	26.96±1.84	23.80±2.50	3.16±1.20	7.64±0.82	4.48±0.56
diabetes	24.96±1.77	23.30±1.81	1.66±0.74	5.37±0.57	3.72±0.35
glass	29.58±4.32	25.09±5.32	4.49±1.69	8.56±1.34	4.07±1.33
heartc	19.14±3.02	16.86±3.57	2.28±1.37	5.46±1.03	3.17±0.85
hearth	20.28±3.14	20.45±3.07	-1.83±1.73	4.13±1.40	4.30±1.04
hearts	63.57±4.24	60.77±6.53	2.80±3.36	12.05±2.73	9.24±1.59
heartv	19.69±2.56	15.19±2.54	4.51±0.98	6.91±0.84	2.41±0.79
hepatitis	18.29±4.84	17.33±6.07	0.96±1.68	4.20±1.20	3.24±1.45
hypo	0.51±0.18	0.40±0.20	0.11±0.06	0.18±0.04	0.07±0.05
ionosphere	8.16±2.50	7.13±2.72	1.04±0.72	1.87±0.51	0.83±0.44
iris	6.24±3.21	6.00±3.31	0.24±0.84	0.84±0.46	0.60±0.51
krk	24.63±0.20	19.50±0.30	5.13±0.20	9.22±0.14	4.10±0.13
krkp	0.85±0.19	0.66±0.30	0.19±0.16	0.36±0.10	0.17±0.07
labor	16.08±4.55	12.33±7.03	3.74±2.95	6.13±1.78	2.39±1.68
letter	7.94±0.27	5.52±0.33	2.43±0.18	3.43±0.15	1.01±0.07
lymph	19.67±3.58	14.86±3.43	4.82±1.64	7.71±1.36	2.89±0.83
phoneme	12.27±0.57	10.99±0.71	1.27±0.30	3.24±0.20	1.96±0.19
primary	61.97±3.14	56.63±3.55	5.34±1.23	9.41±1.16	4.07±0.78
satimage	10.23±0.68	9.26±0.62	0.97±0.26	2.38±0.20	1.41±0.12
segment	3.74±0.31	3.03±0.29	0.71±0.21	1.39±0.22	0.68±0.09
shuttle	0.05±0.01	0.05±0.01	0.00±0.00	0.01±0.00	0.01±0.00
sick	1.56±0.28	1.51±0.31	0.05±0.10	0.40±0.08	0.35±0.10
sonar	24.54±3.19	24.02±5.48	0.52±2.90	7.30±1.25	6.78±2.04
soybean	10.01±2.03	8.06±2.43	1.95±1.01	3.35±0.76	1.40±0.42
splice	7.01±0.49	6.77±0.47	0.24±0.15	1.01±0.15	0.77±0.10
vehicle	26.96±1.88	26.02±2.10	0.95±1.50	6.91±1.19	5.96±0.49
voting	4.32±1.55	3.22±1.23	1.10±0.55	1.44±0.63	0.35±0.26
waveform	17.25±1.01	15.44±1.15	1.81±0.36	5.05±0.23	3.24±0.27
Mean	17.37	15.20	2.18	4.68	2.50

Table F.3: Results of bias-variance decomposition for $Bagging(0.5; 30)$: absolute values.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	1.36	3.50	0.60	0.84	3.53
audiology	1.03	0.96	1.34	1.04	0.60
autos	1.01	1.23	0.73	0.78	1.06
balance	0.81	0.72	1.42	0.94	0.57
breastc	0.90	0.91	0.82	0.75	0.70
breastw	0.70	0.90	0.25	0.33	0.45
colic	0.86	0.93	0.50	0.50	0.49
credita	0.76	0.95	0.22	0.37	0.59
creditg	0.83	0.89	0.54	0.53	0.52
diabetes	0.85	1.00	0.27	0.42	0.56
glass	0.85	0.98	0.48	0.58	0.75
heartc	0.76	0.87	0.41	0.48	0.54
hearth	0.90	1.04	-1.94	0.54	0.90
hearts	0.92	1.04	0.27	0.61	0.99
heartv	0.76	0.73	0.85	0.62	0.41
hepatitis	0.82	0.90	0.31	0.53	0.66
hypo	0.92	1.15	0.52	0.68	1.28
ionosphere	0.76	1.00	0.28	0.35	0.52
iris	0.97	1.12	0.22	0.56	1.42
krk	0.99	1.20	0.60	0.76	1.13
krkp	1.08	1.91	0.43	0.68	1.93
labor	0.90	0.88	0.98	0.75	0.55
letter	0.56	0.95	0.29	0.36	0.88
lymph	0.80	0.76	0.94	0.69	0.48
phoneme	0.85	1.08	0.29	0.45	0.69
primary	0.93	0.97	0.67	0.77	0.95
satimage	0.66	0.97	0.16	0.28	0.58
segment	0.85	1.35	0.33	0.52	1.25
shuttle	1.33	1.87	0.26	0.77	1.81
sick	0.99	1.33	0.12	0.51	1.03
sonar	0.83	1.28	0.05	0.42	1.03
soybean	0.88	1.02	0.57	0.64	0.76
splice	1.13	1.16	0.68	0.95	1.09
vehicle	0.89	1.02	0.21	0.56	0.77
voting	0.69	0.78	0.51	0.53	0.60
waveform	0.66	0.91	0.20	0.34	0.55
Mean	0.88	1.12	0.48	0.59	0.91

Table F.4: Results of bias-variance decomposition for $Bagging(0.5; 30)$: ratios relative to the base classifier.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	0.82±0.27	0.22±0.23	0.60±0.25	0.64±0.22	0.04±0.06
audiology	24.36±4.03	20.30±5.21	4.06±2.12	7.04±1.61	2.98±1.31
autos	24.13±3.72	15.07±3.54	9.05±2.31	11.05±2.01	2.00±0.76
balance	20.55±1.79	18.70±1.92	1.85±0.43	4.95±0.45	3.11±0.43
breastc	34.27±3.63	31.49±4.50	2.78±1.76	9.74±1.42	6.96±0.84
breastw	4.89±0.89	4.01±0.88	0.89±0.43	1.62±0.34	0.74±0.23
colic	15.90±1.95	14.13±2.36	1.77±0.97	3.52±0.67	1.76±0.53
credita	16.06±1.96	14.06±2.51	2.00±0.91	4.66±0.48	2.66±0.56
creditg	27.64±1.66	24.00±2.22	3.64±1.07	8.78±0.65	5.15±0.61
diabetes	26.33±1.94	23.57±1.85	2.76±0.73	7.04±0.43	4.28±0.49
glass	30.73±4.13	25.13±4.82	5.60±2.10	10.37±1.43	4.78±1.27
heartc	21.85±3.39	19.51±4.57	2.35±1.54	6.77±1.07	4.42±1.11
hearth	22.09±2.90	20.09±3.00	2.00±2.05	6.18±1.73	4.18±0.91
hearts	66.66±3.97	63.27±6.41	3.39±3.19	13.37±2.05	9.98±1.65
heartv	23.14±2.81	20.74±3.40	2.40±1.39	7.12±0.94	4.73±0.90
hepatitis	19.84±4.46	16.67±5.21	3.17±0.97	6.28±1.17	3.11±1.15
hypo	0.50±0.14	0.34±0.17	0.15±0.06	0.20±0.03	0.05±0.04
ionosphere	7.99±2.06	6.27±2.00	1.72±0.48	2.59±0.60	0.87±0.40
iris	6.02±3.26	5.33±3.44	0.69±0.74	1.07±0.54	0.38±0.34
krk	22.47±0.20	15.81±0.27	6.65±0.20	9.93±0.18	3.28±0.09
krkp	0.73±0.13	0.34±0.16	0.38±0.14	0.47±0.11	0.09±0.05
labor	15.66±4.62	14.00±6.70	1.66±2.71	4.87±1.36	3.21±1.84
letter	8.69±0.23	5.07±0.25	3.62±0.18	4.60±0.17	0.98±0.05
lymph	21.72±3.11	18.86±4.37	2.86±2.11	7.81±1.34	4.95±0.93
phoneme	11.08±0.54	8.94±0.48	2.14±0.24	4.05±0.20	1.90±0.14
primary	64.36±3.16	57.51±3.83	6.85±2.06	10.78±1.80	3.93±0.86
satimage	10.35±0.63	8.81±0.60	1.53±0.13	3.04±0.13	1.50±0.11
segment	3.40±0.39	2.38±0.42	1.02±0.25	1.58±0.23	0.56±0.10
shuttle	0.04±0.01	0.02±0.01	0.02±0.00	0.02±0.01	0.00±0.00
sick	1.47±0.24	1.17±0.32	0.31±0.20	0.61±0.14	0.31±0.11
sonar	25.04±2.91	20.14±4.41	4.89±2.74	10.75±2.00	5.86±1.39
soybean	10.34±2.06	8.20±2.57	2.14±1.28	3.92±0.90	1.78±0.74
splice	7.29±0.57	5.92±0.40	1.37±0.50	2.11±0.51	0.75±0.15
vehicle	26.94±2.35	24.01±3.12	2.93±1.12	8.75±0.77	5.82±0.79
voting	5.61±1.96	3.91±1.71	1.71±0.76	2.20±0.75	0.50±0.26
waveform	18.46±1.03	16.12±1.22	2.34±0.28	6.10±0.18	3.76±0.27
Mean	17.98	15.39	2.59	5.41	2.81

Table F.5: Results of bias-variance decomposition for $Bagging(2; 30)$: absolute values.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	0.97	1.00	0.96	0.94	0.73
audiology	0.98	1.00	0.89	0.92	0.98
autos	0.89	1.00	0.76	0.77	0.86
balance	0.94	0.98	0.70	0.80	0.89
breastc	0.94	1.00	0.57	0.72	0.81
breastw	0.77	0.90	0.46	0.52	0.61
colic	0.89	0.94	0.60	0.64	0.68
credita	0.82	0.97	0.40	0.55	0.77
creditg	0.85	0.90	0.62	0.61	0.60
diabetes	0.89	1.01	0.45	0.55	0.64
glass	0.88	0.98	0.60	0.70	0.88
heartc	0.87	1.00	0.42	0.59	0.76
hearth	0.98	1.02	0.69	0.80	0.87
hearts	0.97	1.08	0.33	0.68	1.07
heartv	0.89	1.00	0.45	0.63	0.80
hepatitis	0.89	0.87	1.02	0.79	0.64
hypo	0.90	1.00	0.73	0.76	0.88
ionosphere	0.74	0.88	0.47	0.49	0.55
iris	0.94	1.00	0.63	0.71	0.89
krk	0.90	0.97	0.78	0.81	0.91
krkp	0.93	1.00	0.87	0.89	0.97
labor	0.88	1.00	0.43	0.59	0.74
letter	0.61	0.87	0.43	0.48	0.86
lymph	0.88	0.97	0.56	0.70	0.82
phoneme	0.76	0.88	0.49	0.56	0.66
primary	0.97	0.98	0.86	0.88	0.92
satimage	0.67	0.92	0.26	0.36	0.61
segment	0.77	1.06	0.47	0.58	1.03
shuttle	0.99	0.87	1.23	1.07	0.74
sick	0.93	1.02	0.69	0.79	0.91
sonar	0.85	1.07	0.46	0.62	0.89
soybean	0.91	1.04	0.63	0.74	0.96
splice	1.17	1.01	3.85	2.00	1.06
vehicle	0.89	0.94	0.64	0.71	0.75
voting	0.89	0.95	0.79	0.81	0.86
waveform	0.71	0.95	0.26	0.41	0.63
Mean	0.88	0.97	0.71	0.73	0.81

Table F.6: Results of bias-variance decomposition for $Bagging(2; 30)$: ratios relative to the base classifier.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	0.90±0.29	0.33±0.27	0.57±0.18	0.64±0.16	0.07±0.07
audiology	23.95±4.03	19.01±5.04	4.94±1.73	7.01±1.47	2.07±1.04
autos	25.28±3.25	17.05±3.72	8.23±2.11	10.55±1.72	2.32±0.74
balance	18.41±1.86	16.94±2.20	1.47±0.95	4.51±0.72	3.04±0.51
breastc	33.24±3.30	28.72±4.62	4.52±2.41	10.43±1.85	5.91±1.15
breastw	4.57±0.99	4.15±1.19	0.42±0.49	1.11±0.32	0.69±0.29
colic	15.58±1.91	13.86±2.34	1.72±0.88	2.97±0.70	1.25±0.41
credita	15.27±1.92	13.91±2.40	1.35±0.78	3.65±0.52	2.29±0.35
creditg	27.37±1.47	24.50±2.06	2.87±1.05	7.84±0.69	4.97±0.56
diabetes	24.79±1.72	23.57±1.48	1.22±0.64	5.14±0.41	3.92±0.55
glass	29.58±4.16	23.68±5.42	5.90±1.99	9.91±1.48	4.01±1.33
heartc	19.46±3.10	16.54±3.54	2.92±1.30	6.13±1.21	3.21±0.62
hearth	21.29±3.08	21.79±2.86	-1.49±1.42	4.21±1.28	4.71±1.16
hearts	65.79±3.48	61.60±6.86	4.19±4.53	12.62±2.86	8.43±2.25
heartv	20.54±2.64	17.04±3.05	3.51±1.68	6.75±1.16	3.25±1.07
hepatitis	18.73±4.89	17.33±6.07	1.40±1.62	4.26±1.18	2.86±1.32
hypo	0.47±0.15	0.32±0.16	0.15±0.06	0.19±0.06	0.04±0.03
ionosphere	8.49±2.52	7.13±2.26	1.36±0.82	2.18±0.77	0.82±0.40
iris	5.96±3.28	5.33±3.44	0.62±0.67	0.89±0.54	0.27±0.19
krk	23.52±0.20	18.15±0.36	5.37±0.22	9.17±0.15	3.80±0.15
krkp	0.81±0.18	0.56±0.25	0.24±0.13	0.38±0.09	0.14±0.06
labor	16.41±4.79	12.33±7.03	4.08±2.71	6.52±1.78	2.44±1.47
letter	8.06±0.25	5.50±0.30	2.56±0.18	3.59±0.15	1.03±0.07
lymph	20.14±3.36	15.48±4.42	4.67±1.99	7.85±1.12	3.18±1.16
phoneme	12.56±0.59	11.38±0.74	1.18±0.29	3.23±0.17	2.06±0.19
primary	62.48±3.01	56.64±3.09	5.84±1.21	9.81±1.20	3.97±0.67
satimage	10.20±0.65	9.03±0.61	1.17±0.13	2.56±0.13	1.39±0.11
segment	3.58±0.34	2.73±0.37	0.85±0.24	1.44±0.21	0.59±0.11
shuttle	0.05±0.01	0.04±0.01	0.00±0.00	0.01±0.00	0.01±0.00
sick	1.49±0.26	1.35±0.30	0.14±0.13	0.46±0.10	0.32±0.11
sonar	23.55±2.89	20.67±4.04	2.88±2.26	8.91±1.46	6.03±1.44
soybean	9.66±2.03	8.05±2.78	1.61±1.08	3.20±0.70	1.60±0.63
splice	6.33±0.35	6.24±0.57	0.09±0.34	0.77±0.26	0.67±0.17
vehicle	26.71±2.06	25.55±2.45	1.17±1.31	7.23±1.03	6.06±0.63
voting	4.48±1.70	2.99±1.54	1.49±0.58	1.64±0.64	0.15±0.21
waveform	17.64±0.99	16.06±1.09	1.58±0.18	5.05±0.16	3.47±0.24
Mean	17.43	15.15	2.27	4.80	2.53

Table F.7: Results of bias-variance decomposition for $Cragging(2; 15)$: absolute values.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	1.07	1.50	0.92	0.94	1.20
audiology	0.96	0.94	1.08	0.92	0.68
autos	0.94	1.13	0.69	0.74	1.01
balance	0.84	0.88	0.55	0.73	0.87
breastc	0.91	0.91	0.92	0.77	0.69
breastw	0.72	0.94	0.21	0.35	0.57
colic	0.87	0.93	0.59	0.54	0.48
credita	0.78	0.96	0.27	0.43	0.66
creditg	0.84	0.92	0.49	0.54	0.58
diabetes	0.84	1.01	0.20	0.40	0.59
glass	0.85	0.92	0.64	0.67	0.74
heartc	0.78	0.85	0.53	0.54	0.55
hearth	0.94	1.10	-1.83	0.55	0.98
hearts	0.96	1.05	0.40	0.64	0.90
heartv	0.79	0.82	0.66	0.60	0.55
hepatitis	0.84	0.90	0.45	0.53	0.58
hypo	0.84	0.92	0.71	0.71	0.72
ionosphere	0.79	1.00	0.37	0.41	0.52
iris	0.93	1.00	0.57	0.59	0.63
krk	0.95	1.11	0.63	0.75	1.05
krkp	1.03	1.64	0.55	0.72	1.51
labor	0.92	0.88	1.06	0.80	0.56
letter	0.57	0.95	0.30	0.37	0.90
lymph	0.82	0.79	0.91	0.70	0.53
phoneme	0.87	1.12	0.27	0.45	0.72
primary	0.94	0.97	0.73	0.80	0.93
satimage	0.66	0.95	0.20	0.30	0.57
segment	0.81	1.21	0.40	0.53	1.09
shuttle	1.20	1.60	0.38	0.73	1.45
sick	0.94	1.19	0.30	0.59	0.96
sonar	0.80	1.10	0.27	0.52	0.91
soybean	0.85	1.02	0.47	0.61	0.87
splice	1.02	1.06	0.26	0.73	0.96
vehicle	0.89	1.00	0.25	0.58	0.78
voting	0.71	0.72	0.69	0.60	0.27
waveform	0.68	0.94	0.18	0.34	0.58
Mean	0.86	1.03	0.50	0.60	0.78

Table F.8: Results of bias-variance decomposition for *Cragging*$(2; 15)$: ratios relative to the base classifier.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	0.77±0.29	0.22±0.35	0.55±0.22	0.60±0.18	0.06±0.09
audiology	24.19±4.21	18.99±4.58	5.20±1.35	7.38±1.59	2.18±0.74
autos	24.64±3.46	15.07±3.33	9.57±1.91	11.28±1.83	1.71±0.62
balance	19.63±1.81	18.06±2.41	1.57±0.75	4.64±0.58	3.07±0.54
breastc	33.71±3.57	30.46±4.63	3.25±1.47	9.83±1.39	6.58±0.59
breastw	4.73±0.91	4.00±1.00	0.72±0.32	1.41±0.20	0.69±0.29
colic	15.84±1.97	13.86±2.25	1.98±1.08	3.39±0.73	1.41±0.58
credita	16.02±1.88	14.06±2.56	1.97±0.93	4.40±0.58	2.43±0.48
creditg	28.00±1.71	24.50±2.02	3.50±0.96	8.74±0.50	5.24±0.66
diabetes	25.64±1.94	24.48±1.78	1.16±0.76	5.73±0.56	4.57±0.53
glass	30.08±4.27	25.13±5.39	4.95±2.88	9.62±2.07	4.67±1.34
heartc	20.68±3.27	16.20±3.29	4.48±1.99	7.48±1.68	3.00±0.71
hearth	21.57±2.93	21.78±2.60	-1.79±1.21	4.78±1.44	4.99±0.85
hearts	66.57±3.20	60.77±7.09	5.81±5.02	14.31±3.20	8.50±2.27
heartv	21.78±2.52	17.78±3.00	4.00±1.66	7.46±1.08	3.46±0.91
hepatitis	19.84±4.61	18.00±5.62	1.84±1.35	5.15±1.34	3.31±1.34
hypo	0.48±0.14	0.32±0.16	0.17±0.08	0.20±0.07	0.04±0.03
ionosphere	8.73±2.35	7.13±2.54	1.61±0.66	2.49±0.52	0.89±0.44
iris	6.20±3.25	5.33±3.44	0.87±0.77	1.16±0.62	0.29±0.26
krk	22.75±0.20	16.61±0.23	6.14±0.12	9.61±0.13	3.47±0.10
krkp	0.76±0.14	0.44±0.18	0.32±0.14	0.44±0.11	0.12±0.06
labor	16.23±4.60	14.00±6.70	2.23±2.68	5.70±1.50	3.47±1.66
letter	8.54±0.26	5.31±0.23	3.23±0.17	4.23±0.16	1.00±0.04
lymph	20.55±3.19	15.48±4.13	5.07±1.46	8.51±0.78	3.44±0.86
phoneme	12.16±0.58	10.49±0.68	1.67±0.28	3.66±0.20	1.99±0.18
primary	63.57±3.26	57.51±4.02	6.05±1.66	10.08±1.43	4.02±0.82
satimage	10.33±0.65	8.86±0.57	1.47±0.23	2.93±0.20	1.46±0.12
segment	3.47±0.38	2.55±0.47	0.92±0.25	1.51±0.20	0.59±0.16
shuttle	0.04±0.01	0.03±0.01	0.01±0.01	0.02±0.00	0.01±0.00
sick	1.50±0.25	1.30±0.30	0.20±0.13	0.53±0.08	0.33±0.12
sonar	23.63±2.94	19.24±5.16	4.40±3.10	10.05±2.08	5.65±1.68
soybean	10.02±2.06	8.20±2.77	1.82±1.19	3.61±0.75	1.78±0.79
splice	6.17±0.33	5.86±0.39	0.30±0.26	0.91±0.21	0.60±0.14
vehicle	26.95±2.21	25.90±2.51	1.04±0.96	7.56±0.75	6.51±0.60
voting	5.31±1.91	3.45±1.65	1.87±0.71	2.15±0.72	0.28±0.22
waveform	18.14±1.02	16.20±1.22	1.94±0.27	5.63±0.23	3.69±0.21
Mean	17.76	15.21	2.55	5.20	2.65

Table F.9: Results of bias-variance decomposition for *Cragging*$(3; 10)$: absolute values.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	0.91	1.00	0.88	0.89	1.00
audiology	0.97	0.94	1.14	0.97	0.71
autos	0.91	1.00	0.80	0.79	0.74
balance	0.90	0.94	0.59	0.75	0.88
breastc	0.93	0.97	0.66	0.73	0.77
breastw	0.74	0.90	0.37	0.45	0.57
colic	0.89	0.93	0.68	0.62	0.55
credita	0.82	0.97	0.39	0.52	0.70
creditg	0.86	0.92	0.60	0.60	0.61
diabetes	0.87	1.05	0.19	0.45	0.68
glass	0.86	0.98	0.53	0.65	0.86
heartc	0.83	0.83	0.81	0.66	0.52
hearth	0.95	1.10	-1.93	0.62	1.04
hearts	0.97	1.04	0.56	0.72	0.91
heartv	0.84	0.86	0.76	0.66	0.58
hepatitis	0.89	0.94	0.59	0.64	0.68
hypo	0.88	0.92	0.80	0.78	0.70
ionosphere	0.81	1.00	0.44	0.47	0.56
iris	0.97	1.00	0.80	0.76	0.68
krk	0.91	1.02	0.72	0.79	0.96
krkp	0.97	1.27	0.73	0.83	1.35
labor	0.91	1.00	0.58	0.70	0.80
letter	0.60	0.92	0.38	0.44	0.88
lymph	0.83	0.79	0.99	0.76	0.57
phoneme	0.84	1.03	0.38	0.51	0.70
primary	0.96	0.98	0.76	0.82	0.94
satimage	0.67	0.93	0.25	0.35	0.60
segment	0.79	1.13	0.43	0.56	1.09
shuttle	1.03	1.20	0.67	0.81	1.10
sick	0.95	1.14	0.45	0.67	0.96
sonar	0.80	1.03	0.41	0.58	0.86
soybean	0.88	1.04	0.53	0.68	0.97
splice	0.99	1.00	0.85	0.86	0.86
vehicle	0.89	1.01	0.23	0.61	0.84
voting	0.85	0.83	0.87	0.79	0.49
waveform	0.70	0.95	0.22	0.38	0.62
Mean	0.87	0.99	0.58	0.66	0.79

Table F.10: Results of bias-variance decomposition for *Cragging*$(3; 10)$: ratios relative to the base classifier.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	0.86±0.32	0.11±0.18	0.75±0.23	0.75±0.23	0.00±0.00
audiology	24.77±3.88	20.30±5.21	4.48±2.17	7.58±1.66	3.10±1.26
autos	26.21±3.37	14.60±3.47	11.62±2.59	13.61±2.33	2.00±0.71
balance	21.73±1.64	18.70±2.23	3.02±0.97	6.22±0.74	3.19±0.38
breastc	35.98±3.22	33.57±4.26	2.41±2.27	11.39±1.68	8.97±0.86
breastw	6.11±0.83	4.29±1.01	1.82±0.43	2.90±0.34	1.09±0.30
colic	17.52±1.88	15.77±2.63	1.75±1.18	4.65±0.66	2.90±0.73
credita	19.03±1.53	15.22±2.67	3.82±1.40	7.38±0.85	3.56±0.67
creditg	30.86±1.37	25.50±2.70	5.36±1.45	12.46±0.85	7.10±0.82
diabetes	27.97±1.69	23.82±2.13	4.14±0.83	10.16±0.42	6.02±0.65
glass	34.14±3.52	26.13±3.75	8.01±1.81	13.46±1.19	5.44±1.02
heartc	24.36±2.90	19.47±3.26	4.89±1.88	10.49±1.31	5.60±1.11
hearth	22.68±2.83	20.40±2.27	2.28±1.73	7.21±1.80	4.93±0.77
hearts	68.44±3.30	60.77±5.91	7.68±4.82	17.24±3.56	9.57±1.88
heartv	25.37±2.35	20.37±2.66	5.00±1.40	10.56±0.93	5.56±0.80
hepatitis	21.92±4.16	19.88±6.90	2.04±3.07	7.26±1.53	5.21±2.20
hypo	0.55±0.14	0.32±0.16	0.23±0.08	0.27±0.06	0.04±0.04
ionosphere	10.14±2.07	6.84±2.26	3.30±0.90	4.59±0.72	1.29±0.49
iris	6.47±3.23	5.33±3.44	1.13±0.74	1.51±0.57	0.38±0.34
krk	24.48±0.19	16.27±0.28	8.21±0.19	11.80±0.16	3.59±0.10
krkp	0.78±0.13	0.34±0.17	0.44±0.16	0.53±0.12	0.09±0.06
labor	17.57±4.11	14.00±6.70	3.57±3.59	7.62±1.96	4.06±2.10
letter	12.73±0.31	5.60±0.26	7.13±0.32	8.24±0.30	1.10±0.07
lymph	24.33±3.03	18.81±4.32	5.52±2.51	11.17±1.62	5.65±1.09
phoneme	13.22±0.53	9.72±0.68	3.50±0.43	5.91±0.26	2.41±0.25
primary	65.74±3.48	57.81±4.35	7.93±1.67	11.98±1.33	4.04±1.02
satimage	12.81±0.62	9.21±0.73	3.60±0.22	5.62±0.15	2.02±0.18
segment	4.11±0.41	2.47±0.60	1.64±0.52	2.28±0.38	0.64±0.22
shuttle	0.04±0.01	0.03±0.01	0.01±0.01	0.02±0.01	0.01±0.00
sick	1.55±0.25	1.14±0.29	0.41±0.15	0.74±0.13	0.33±0.09
sonar	27.70±2.37	18.31±4.51	9.39±2.90	15.52±1.82	6.13±1.79
soybean	11.08±1.98	8.19±2.52	2.89±1.26	4.82±0.88	1.94±0.76
splice	6.19±0.29	5.86±0.34	0.33±0.19	1.02±0.19	0.69±0.10
vehicle	29.41±2.03	25.55±2.48	3.86±1.69	11.24±1.30	7.38±0.75
voting	6.24±2.01	4.13±1.77	2.11±0.73	2.67±0.70	0.56±0.31
waveform	21.65±0.90	16.50±1.19	5.15±0.33	9.99±0.15	4.84±0.33
Mean	19.58	15.70	3.87	7.25	3.37

Table F.11: Results of bias-variance decomposition for *Cragging*$(30; 1)$: absolute values.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	1.01	0.50	1.20	1.10	0.00
audiology	1.00	1.00	0.98	0.99	1.01
autos	0.97	0.97	0.97	0.95	0.87
balance	1.00	0.97	1.14	1.01	0.91
breastc	0.99	1.07	0.49	0.85	1.05
breastw	0.96	0.97	0.93	0.92	0.90
colic	0.98	1.05	0.60	0.84	1.12
credita	0.97	1.05	0.76	0.87	1.03
creditg	0.95	0.96	0.92	0.86	0.82
diabetes	0.95	1.02	0.68	0.79	0.90
glass	0.98	1.02	0.86	0.91	1.00
heartc	0.97	1.00	0.88	0.92	0.96
hearth	1.00	1.03	0.78	0.94	1.03
hearts	0.99	1.04	0.74	0.87	1.02
heartv	0.97	0.98	0.94	0.94	0.94
hepatitis	0.98	1.03	0.66	0.91	1.06
hypo	0.99	0.92	1.10	1.04	0.80
ionosphere	0.94	0.96	0.89	0.87	0.81
iris	1.01	1.00	1.04	1.00	0.89
krk	0.98	1.00	0.96	0.97	0.99
krkp	1.00	1.00	1.00	1.00	1.00
labor	0.99	1.00	0.93	0.93	0.93
letter	0.89	0.97	0.85	0.86	0.97
lymph	0.99	0.97	1.07	1.00	0.93
phoneme	0.91	0.96	0.81	0.82	0.84
primary	0.99	0.99	1.00	0.98	0.94
satimage	0.83	0.97	0.60	0.67	0.82
segment	0.93	1.10	0.76	0.85	1.18
shuttle	1.00	1.07	0.87	0.96	1.15
sick	0.98	1.00	0.92	0.95	0.98
sonar	0.94	0.98	0.88	0.90	0.93
soybean	0.98	1.04	0.84	0.92	1.05
splice	1.00	1.00	0.93	0.97	0.98
vehicle	0.98	1.00	0.84	0.91	0.95
voting	0.99	1.00	0.98	0.98	0.97
waveform	0.83	0.97	0.57	0.67	0.81
Mean	0.97	0.99	0.87	0.91	0.93

Table F.12: Results of bias-variance decomposition for *Cragging*$(30; 1)$: ratios relative to the base classifier.

Dataset	*Loss*	*Bias*	*Var*	Var_U	Var_B
anneal	0.85±0.31	0.22±0.23	0.62±0.28	0.68±0.23	0.06±0.09
audiology	24.88±3.94	20.30±5.21	4.58±2.12	7.64±1.68	3.06±1.22
autos	27.03±3.42	15.07±3.33	11.96±2.42	14.27±2.22	2.31±0.73
balance	21.83±1.60	19.18±2.03	2.65±0.84	6.15±0.74	3.50±0.30
breastc	36.38±3.09	31.48±4.85	4.90±2.50	13.46±1.65	8.56±1.13
breastw	6.38±0.84	4.44±1.14	1.94±0.46	3.15±0.29	1.20±0.36
colic	17.88±2.04	14.95±2.74	2.92±0.93	5.51±0.56	2.59±0.62
credita	19.53±1.47	14.49±2.46	5.03±1.46	8.49±0.89	3.46±0.67
creditg	32.55±1.12	26.70±2.32	5.85±1.47	14.48±0.76	8.64±0.90
diabetes	29.43±1.49	23.31±2.30	6.12±1.07	12.80±0.59	6.68±0.77
glass	34.92±3.44	25.63±4.00	9.29±1.94	14.74±1.36	5.45±1.04
heartc	25.04±2.83	19.48±3.19	5.55±1.55	11.38±1.11	5.83±0.92
hearth	22.64±2.80	19.74±2.53	2.90±1.13	7.70±1.55	4.80±0.61
hearts	68.86±3.13	58.46±7.00	10.40±5.91	19.76±4.14	9.36±2.22
heartv	26.04±2.27	20.74±2.92	5.30±1.34	11.22±0.77	5.93±0.78
hepatitis	22.35±4.01	19.25±7.00	3.10±3.28	8.00±1.73	4.90±2.17
hypo	0.55±0.14	0.34±0.17	0.21±0.08	0.26±0.06	0.05±0.04
ionosphere	10.81±1.97	7.13±2.16	3.68±0.70	5.27±0.50	1.59±0.50
iris	6.42±3.23	5.33±3.44	1.09±0.81	1.51±0.58	0.42±0.40
krk	24.87±0.19	16.30±0.29	8.57±0.20	12.20±0.16	3.62±0.10
krkp	0.78±0.13	0.34±0.17	0.44±0.16	0.53±0.12	0.09±0.06
labor	17.83±3.98	14.00±6.70	3.83±3.77	8.19±2.06	4.36±2.26
letter	14.24±0.30	5.80±0.20	8.44±0.30	9.58±0.28	1.14±0.06
lymph	24.62±3.08	19.48±4.50	5.14±2.61	11.19±1.65	6.05±1.19
phoneme	14.50±0.42	10.16±0.56	4.34±0.33	7.20±0.19	2.86±0.23
primary	66.36±3.54	58.40±4.60	7.97±1.94	12.26±1.51	4.29±1.05
satimage	15.49±0.56	9.54±0.63	5.95±0.31	8.40±0.26	2.45±0.17
segment	4.41±0.40	2.25±0.48	2.16±0.42	2.70±0.36	0.54±0.16
shuttle	0.04±0.01	0.03±0.01	0.01±0.00	0.02±0.00	0.01±0.00
sick	1.58±0.24	1.14±0.29	0.44±0.16	0.78±0.13	0.34±0.10
sonar	29.41±1.97	18.76±3.99	10.65±3.01	17.26±2.13	6.61±1.42
soybean	11.33±1.93	7.90±2.45	3.42±1.25	5.27±0.91	1.84±0.72
splice	6.22±0.29	5.86±0.34	0.36±0.19	1.06±0.19	0.70±0.09
vehicle	30.16±1.98	25.55±2.66	4.61±1.72	12.36±1.30	7.75±0.81
voting	6.29±2.02	4.13±1.77	2.15±0.75	2.73±0.72	0.58±0.32
waveform	26.00±0.74	17.00±1.28	9.00±0.62	14.95±0.31	5.95±0.41
Mean	20.24	15.64	4.60	8.14	3.54

Table F.13: Results of bias-variance decomposition for the base classifier: absolute values.

G. Proofs

G.1 Proof of Theorem 5.1

Theorem 5.1 (Page 56): Let $C := \langle n, \mathbf{c}, \mathbf{w}, V \rangle$ be an ensemble of classifiers such that $Y = \widehat{Y} = \mathcal{R}$, $\widehat{Y}_c = \{P(\mathcal{R})\}$ for all $c \in \{1, \ldots, n\}$, and $V = V_{\text{dem}}$. Then, for squared loss, the ensemble loss $L(\mathbf{x}, y)$ can be written as

$$L(\mathbf{x}, y) = f_l(\overline{L}(\mathbf{x}, y), \overline{D}(\mathbf{x})) = \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}).$$

Proof.

$$
\begin{aligned}
& L(\mathbf{x}, y) && \text{(G.1)}\\
&= l_2(\widehat{y}(\mathbf{x}), y) && \text{(G.2)}\\
&= (y - \widehat{y}(\mathbf{x}))^2 && \text{(G.3)}\\
&= (y - \widehat{y}(\mathbf{x}))^2 * 1 && \text{(G.4)}\\
&= (y - \widehat{y}(\mathbf{x}))^2 * \sum_{c=1}^{n} w_c && \text{(G.5)}\\
&= (y - \widehat{y}(\mathbf{x}))^2 * \sum_{c=1}^{n} w_c * 1 && \text{(G.6)}\\
&= (y - \widehat{y}(\mathbf{x}))^2 \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})\, dy' && \text{(G.7)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(\widehat{y}(\mathbf{x}) - y)^2\, dy' && \text{(G.8)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(\widehat{y}(\mathbf{x}) - y)^2\, dy' - 0 && \text{(G.9)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(\widehat{y}(\mathbf{x}) - y)^2\, dy' - 2(\widehat{y}(\mathbf{x}) - y) * 0 && \text{(G.10)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(\widehat{y}(\mathbf{x}) - y)^2\, dy' \\
&\quad - 2(\widehat{y}(\mathbf{x}) - y) \left[\widehat{y}(\mathbf{x}) - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) y'\, dy' \right] && \text{(G.11)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(\widehat{y}(\mathbf{x}) - y)^2\, dy' \\
&\quad - 2(\widehat{y}(\mathbf{x}) - y)\widehat{y}(\mathbf{x}) + 2(\widehat{y}(\mathbf{x}) - y) \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) y'\, dy' && \text{(G.12)}
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(\widehat{y}(\mathbf{x}) - y)^2 \, dy' \\
&\quad - 2(\widehat{y}(\mathbf{x}) - y)\widehat{y}(\mathbf{x}) * 1 + 2(\widehat{y}(\mathbf{x}) - y) \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) y' \, dy' && \text{(G.13)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(\widehat{y}(\mathbf{x}) - y)^2 \, dy' \\
&\quad - 2(\widehat{y}(\mathbf{x}) - y)\widehat{y}(\mathbf{x}) \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \, dy' \\
&\quad + \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) 2(\widehat{y}(\mathbf{x}) - y) y' \, dy' && \text{(G.14)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(\widehat{y}(\mathbf{x}) - y)^2 \, dy' \\
&\quad - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) 2(\widehat{y}(\mathbf{x}) - y)\widehat{y}(\mathbf{x}) \, dy' \\
&\quad + \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) 2(\widehat{y}(\mathbf{x}) - y) y' \, dy' && \text{(G.15)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(\widehat{y}(\mathbf{x}) - y)^2 \, dy' \\
&\quad - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) 2(\widehat{y}(\mathbf{x}) - y)(\widehat{y}(\mathbf{x}) - y') \, dy' && \text{(G.16)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \left[(\widehat{y}(\mathbf{x}) - y)^2 - 2(\widehat{y}(\mathbf{x}) - y)(\widehat{y}(\mathbf{x}) - y') \right] dy' && \text{(G.17)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \left[(\widehat{y}(\mathbf{x}) - y)(\widehat{y}(\mathbf{x}) - y - 2\widehat{y}(\mathbf{x}) + 2y') \right] dy' && \text{(G.18)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \left[(\widehat{y}(\mathbf{x}) - y)(2y' - \widehat{y}(\mathbf{x}) - y) \right] dy' && \text{(G.19)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \left[(y' - y)^2 - (y' - \widehat{y}(\mathbf{x}))^2 \right] dy' && \text{(G.20)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(y' - y)^2 \, dy' \\
&\quad - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(y' - \widehat{y}(\mathbf{x}))^2 \, dy' && \text{(G.21)} \\
&= \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}). && \text{(G.22)}
\end{aligned}
$$

G.11 equals G.10 because $\widehat{y}(\mathbf{x}) = \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) y' \, dy'$ from Equation 5.11. □

G.2 Proof of Theorem 5.2

Theorem 5.2 (Page 56): Let $C := \langle n, \mathbf{c}, \mathbf{w}, V \rangle$ be an ensemble of classifiers such that $Y = \widehat{Y} = \mathcal{R}$, $\widehat{Y}_c = \mathcal{R}$ for all $c \in \{1, \ldots, n\}$, and $V = V_{\text{dem}}$. Then, for squared loss, the ensemble loss $L(\mathbf{x}, y)$ can be written as

$$L(\mathbf{x}, y) = f_l(\overline{L}(\mathbf{x}, y), \overline{D}(\mathbf{x})) = \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}).$$

Proof.

$$\begin{aligned}
& L(\mathbf{x}, y) && \text{(G.23)}\\
&= l_2(\widehat{y}(\mathbf{x}), y) && \text{(G.24)}\\
&= (y - \widehat{y}(\mathbf{x}))^2 && \text{(G.25)}\\
&= (y - \widehat{y}(\mathbf{x}))^2 \sum_{c=1}^{n} w_c && \text{(G.26)}\\
&= \sum_{c=1}^{n} w_c (y - \widehat{y}(\mathbf{x}))^2 && \text{(G.27)}\\
&= \left[\sum_{c=1}^{n} w_c (\widehat{y}(\mathbf{x}) - y)^2\right] - 2(\widehat{y}(\mathbf{x}) - y) * 0 && \text{(G.28)}\\
&= \left[\sum_{c=1}^{n} w_c (\widehat{y}(\mathbf{x}) - y)^2\right] - 2(\widehat{y}(\mathbf{x}) - y)\left[\widehat{y}(\mathbf{x}) - \sum_{c=1}^{n} w_c \widehat{y}_c(\mathbf{x})\right] && \text{(G.29)}\\
&= \sum_{c=1}^{n} w_c \left[(\widehat{y}(\mathbf{x}) - y)^2 - 2(\widehat{y}(\mathbf{x}) - y)(\widehat{y}(\mathbf{x}) - \widehat{y}_c(\mathbf{x}))\right] && \text{(G.30)}\\
&= \sum_{c=1}^{n} w_c \left[(\widehat{y}(\mathbf{x}) - y)(\widehat{y}(\mathbf{x}) - y - 2\widehat{y}(\mathbf{x}) + 2\widehat{y}_c(\mathbf{x}))\right] && \text{(G.31)}\\
&= \sum_{c=1}^{n} w_c \left[(\widehat{y}(\mathbf{x}) - y)(2\widehat{y}_c(\mathbf{x}) - \widehat{y}(\mathbf{x}) - y)\right] && \text{(G.32)}\\
&= \sum_{c=1}^{n} w_c \left[(\widehat{y}_c(\mathbf{x}) - y)^2 - (\widehat{y}_c(\mathbf{x}) - \widehat{y}(\mathbf{x}))^2\right] && \text{(G.33)}\\
&= \sum_{c=1}^{n} w_c (\widehat{y}_c(\mathbf{x}) - y)^2 - \sum_{c=1}^{n} w_c (\widehat{y}_c(\mathbf{x}) - \widehat{y}(\mathbf{x}))^2 && \text{(G.34)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(y' - y)^2 \, dy' \\
&\quad - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})(y' - \widehat{y}(\mathbf{x}))^2 \, dy' && \text{(G.35)}\\
&= \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}). && \text{(G.36)}
\end{aligned}$$

G.26 equals G.25 because $\sum_{c=1}^{n} w_c = 1$.
G.29 equals G.28 because $\widehat{y}(\mathbf{x}) = \sum_{c=1}^{n} w_c \widehat{y}_c(\mathbf{x})$ from Equation 5.7 and therefore $\widehat{y}(\mathbf{x}) - \sum_{c=1}^{n} w_c \widehat{y}_c(\mathbf{x}) = 0$.
G.35 equals G.34 because, with $\widehat{Y}_c = \mathcal{R}$, $\widehat{p}_c(y'|\mathbf{x}) := I(y' = \widehat{y}_c(\mathbf{x}))$ and therefore $\widehat{y}_c(\mathbf{x})$ can be written as $\widehat{y}_c(\mathbf{x}) = \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) y' \, dy'$. □

G.3 Proof of Theorem 5.3

Theorem 5.3 (Page 56): For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and squared loss, the expected ensemble loss over the domain is

$$L = \overline{L} - \overline{D}.$$

Proof.

$$\begin{aligned}
L &= E_{P(\mathbf{X},Y)}\left[L(\mathbf{x}, y)\right] && \text{(G.37)}\\
&= \int_{\mathbf{X}\times Y} L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.38)}\\
&= \int_{\mathbf{X}\times Y} (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.39)}\\
&= \int_{\mathbf{X}\times Y} \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle - \int_{\mathbf{X}\times Y} \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.40)}\\
&= \int_{\mathbf{X}\times Y} \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle - \int_{\mathbf{X}} \overline{D}(\mathbf{x})p(\mathbf{x})\, d\mathbf{x} && \text{(G.41)}\\
&= E_{P(\mathbf{X},Y)}\left[L(\mathbf{x}, y)\right] - E_{P(\mathbf{X})}\left[\overline{D}(\mathbf{x})\right] && \text{(G.42)}\\
&= \overline{L} - \overline{D}. && \text{(G.43)}
\end{aligned}$$

Equation G.41 equals Equation G.40 because

$$\begin{aligned}
\int_{\mathbf{X}\times Y} \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle &= \int_{\mathbf{X}}\int_{Y} \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, dy\, d\mathbf{x} && \text{(G.44)}\\
&= \int_{\mathbf{X}}\int_{Y} \overline{D}(\mathbf{x})p(y|\mathbf{x})p(\mathbf{x})\, dy\, d\mathbf{x} && \text{(G.45)}\\
&= \int_{\mathbf{X}} \overline{D}(\mathbf{x})p(\mathbf{x}) \int_{Y} p(y|\mathbf{x})\, dy\, d\mathbf{x} && \text{(G.46)}\\
&= \int_{\mathbf{X}} \overline{D}(\mathbf{x})p(\mathbf{x}) * 1\, d\mathbf{x} && \text{(G.47)}\\
&= \int_{\mathbf{X}} \overline{D}(\mathbf{x})p(\mathbf{x})\, d\mathbf{x}. && \text{(G.48)}
\end{aligned}$$

□

G.4 Proof of Theorem 5.4

Theorem 5.4 (Page 57): For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and 0-1 loss in two-class problems, the ensemble loss $L(\mathbf{x}, y)$ can be written as

$$L(\mathbf{x}, y) = f_l(\overline{L}(\mathbf{x}, y), \overline{D}(\mathbf{x})) = \overline{L}(\mathbf{x}, y) + z\overline{D}(\mathbf{x})$$

with $z = -1$ iff $\widehat{y}(\mathbf{x}) = y$, and $z = 1$ iff $\widehat{y}(\mathbf{x}) \neq y$.

Proof. Case 1: $\widehat{y}(\mathbf{x}) = y$. To show: $L(\mathbf{x}, y) = \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x})$.

$$\begin{aligned}
& L(\mathbf{x}, y) \\
&= l_{01}(\widehat{y}(\mathbf{x}), y) = 0 && \text{(G.49)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', y)\, dy' \\
&\quad - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', \widehat{y}(\mathbf{x}))\, dy' && \text{(G.50)} \\
&= \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}). && \text{(G.51)}
\end{aligned}$$

Case 2: $\widehat{y}(\mathbf{x}) \neq y$. To show: $L(\mathbf{x}, y) = \overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x})$.

$$\begin{aligned}
& L(\mathbf{x}, y) \\
&= l_{01}(\widehat{y}(\mathbf{x}), y) = 1 && \text{(G.52)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})\, dy' && \text{(G.53)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' \neq y)\, dy' \\
&\quad + \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' = y)\, dy' && \text{(G.54)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' \neq y)\, dy' \\
&\quad + \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' \neq \widehat{y}(\mathbf{x}))\, dy' && \text{(G.55)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', y)\, dy' \\
&\quad + \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', \widehat{y}(\mathbf{x}))\, dy' && \text{(G.56)} \\
&= \overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x}). && \text{(G.57)}
\end{aligned}$$

G.55 equals G.54 because there are only two classes and $\widehat{y}(\mathbf{x}) \neq y$, therefore it holds $y' \neq \widehat{y}(\mathbf{x}) \Leftrightarrow y' = y$.

□

G.5 Proof of Theorem 5.5

Theorem 5.5 (Page 57): Under 0-1 loss, the expected diversity $\overline{D}$ can be written as

$$\overline{D} = (1 - L)\overline{D}_T + L\overline{D}_F.$$

Proof.

$$\begin{aligned}
\overline{D} &= E_{P(\mathbf{X})}\left[\overline{D}(\mathbf{x})\right] && \text{(G.58)}\\
&= \int_{\mathbf{X}} \overline{D}(\mathbf{x})p(\mathbf{x})\, d\mathbf{x} && \text{(G.59)}\\
&= \int_{\mathbf{X}\times Y} \overline{D}(\mathbf{x})p(\mathbf{x},y)\, d\langle \mathbf{x},y\rangle && \text{(G.60)}\\
&= \int_{T} \overline{D}(\mathbf{x})p(\mathbf{x},y)\, d\langle \mathbf{x},y\rangle + \int_{F} \overline{D}(\mathbf{x})p(\mathbf{x},y)\, d\langle \mathbf{x},y\rangle && \text{(G.61)}\\
&= \int_{T} \overline{D}(\mathbf{x})p(\mathbf{x},y)p(\langle \mathbf{x},y\rangle \in T)\, d\langle \mathbf{x},y\rangle \\
&\quad + \int_{F} \overline{D}(\mathbf{x})p(\mathbf{x},y)p(\langle \mathbf{x},y\rangle \in F)\, d\langle \mathbf{x},y\rangle && \text{(G.62)}\\
&= \int_{T} \overline{D}(\mathbf{x})p(\langle \mathbf{x},y\rangle \wedge \langle \mathbf{x},y\rangle \in T)\, d\langle \mathbf{x},y\rangle \\
&\quad + \int_{F} \overline{D}(\mathbf{x})p(\langle \mathbf{x},y\rangle \wedge \langle \mathbf{x},y\rangle \in F)\, d\langle \mathbf{x},y\rangle && \text{(G.63)}\\
&= p(\langle \mathbf{x},y\rangle \in T) \int_{T} \overline{D}(\mathbf{x}) \frac{p(\langle \mathbf{x},y\rangle \wedge \langle \mathbf{x},y\rangle \in T)}{p(\langle \mathbf{x},y\rangle \in T)}\, d\langle \mathbf{x},y\rangle \\
&\quad + p(\langle \mathbf{x},y\rangle \in F) \int_{F} \overline{D}(\mathbf{x}) \frac{p(\langle \mathbf{x},y\rangle \wedge \langle \mathbf{x},y\rangle \in F)}{p(\langle \mathbf{x},y\rangle \in F)}\, d\langle \mathbf{x},y\rangle && \text{(G.64)}\\
&= p(\langle \mathbf{x},y\rangle \in T) \int_{T} \overline{D}(\mathbf{x})p(\langle \mathbf{x},y\rangle | \langle \mathbf{x},y\rangle \in T)\, d\langle \mathbf{x},y\rangle \\
&\quad + p(\langle \mathbf{x},y\rangle \in F) \int_{F} \overline{D}(\mathbf{x})p(\langle \mathbf{x},y\rangle | \langle \mathbf{x},y\rangle \in F)\, d\langle \mathbf{x},y\rangle && \text{(G.65)}\\
&= (1 - L)\overline{D}_T + L\overline{D}_F. && \text{(G.66)}
\end{aligned}$$

□

G.6 Proof of Theorem 5.6

Theorem 5.6 (Page 58): For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and 0-1 loss in two-class problems, the expected ensemble loss over the domain can be written as

$$L = \frac{\overline{L} - \overline{D}}{1 - 2\overline{D}_F}. \tag{G.67}$$

Proof. We will show that $L(1-2\overline{D}_F) = \overline{L} - \overline{D}$, from which Equation G.67 can be obtained by a simple rearrangement of the terms.

$$\begin{aligned}
L(1 - 2\overline{D}_F) &= L - 2L\overline{D}_F && \text{(G.68)}\\
&= \int_{\mathbf{X}\times Y} L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - 2L \int_F \overline{D}(\mathbf{x})p(\langle \mathbf{x}, y\rangle \mid \langle \mathbf{x}, y\rangle \in F)\, d\langle \mathbf{x}, y\rangle && \text{(G.69)}\\
&= \int_{\mathbf{X}\times Y} L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - 2p(\langle \mathbf{x}, y\rangle \in F) \int_F \overline{D}(\mathbf{x})p(\langle \mathbf{x}, y\rangle \mid \langle \mathbf{x}, y\rangle \in F)\, d\langle \mathbf{x}, y\rangle && \text{(G.70)}\\
&= \int_T L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - 2p(\langle \mathbf{x}, y\rangle \in F) \int_F \overline{D}(\mathbf{x})p(\langle \mathbf{x}, y\rangle \mid \langle \mathbf{x}, y\rangle \in F)\, d\langle \mathbf{x}, y\rangle && \text{(G.71)}\\
&= \int_T L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - 2p(\langle \mathbf{x}, y\rangle \in F) \int_F \overline{D}(\mathbf{x}) \frac{p(\langle \mathbf{x}, y\rangle \wedge \langle \mathbf{x}, y\rangle \in F)}{p(\langle \mathbf{x}, y\rangle \in F)}\, d\langle \mathbf{x}, y\rangle && \text{(G.72)}\\
&= \int_T L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - 2 \int_F \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.73)}\\
&= \int_T (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad + \int_F (\overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - 2 \int_F \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.74)}\\
&= \int_T \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - \int_T \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle - \int_F \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.75)}\\
&= \int_{\mathbf{X}\times Y} \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle - \int_{\mathbf{X}\times Y} \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.76)}\\
&= \overline{L} - \overline{D}. && \text{(G.77)}
\end{aligned}$$

□

G.7 Proof of Theorem 5.7

Theorem 5.7 (Page 58): For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and 0-1 loss in two-class problems, the expected ensemble loss over the domain can be written as

$$L = \frac{\overline{L} - \overline{D}_T}{1 - \overline{D}_T - \overline{D}_F} \tag{G.78}$$

Proof. We will show that $L = \overline{L} - (1 - L)\overline{D}_T + L\overline{D}_F$, from which Equation G.78 can be obtained by a simple rearrangement of the terms.

$$\begin{aligned}
L &= E_{P(\mathbf{X},Y)}\left[L(\mathbf{x}, y)\right] && \text{(G.79)}\\
&= \int_{XxY} L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.80)}\\
&= \int_T (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F (\overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.81)}\\
&= \int_T \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - \int_T \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.82)}\\
&= \int_{\mathbf{X}\times Y} \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - p(\langle \mathbf{x}, y\rangle \in T) \int_T \overline{D}(\mathbf{x})p(\langle \mathbf{x}, y\rangle \mid \langle \mathbf{x}, y\rangle \in T)\, d\langle \mathbf{x}, y\rangle \\
&\quad + p(\langle \mathbf{x}, y\rangle \in F) \int_F \overline{D}(\mathbf{x})p(\langle \mathbf{x}, y\rangle \mid \langle \mathbf{x}, y\rangle \in F)\, d\langle \mathbf{x}, y\rangle && \text{(G.83)}\\
&= \overline{L} - p(\langle \mathbf{x}, y\rangle \in T)\overline{D}_T + p(\langle \mathbf{x}, y\rangle \in F)\overline{D}_F && \text{(G.84)}\\
&= \overline{L} - (1 - L)\overline{D}_T + L\overline{D}_F. && \text{(G.85)}
\end{aligned}$$

□

G.8 Proof of Theorem 5.8

Theorem 5.8 (Page 58): For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and 0-1 loss, the ensemble loss $L(\mathbf{x}, y)$ can be written as

$$L(\mathbf{x}, y) = f_l(\overline{L}(\mathbf{x}, y), \overline{D}(\mathbf{x})) = \overline{L}(\mathbf{x}, y) + z\overline{D}(\mathbf{x}),$$

with $z = -1$ iff $\widehat{y}(\mathbf{x}) = y$, and $z = \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|x)}$ iff $\widehat{y}(\mathbf{x}) \neq y$.

Proof. Case 1: $\widehat{y}(\mathbf{x}) = y$. To show: $L(\mathbf{x}, y) = \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x})$.

$$\begin{aligned}
& L(\mathbf{x}, y) \\
&= l_{01}(\widehat{y}(\mathbf{x}), y) = 0 && \text{(G.86)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', y)\, dy' \\
&\quad - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', \widehat{y}(\mathbf{x}))\, dy' && \text{(G.87)} \\
&= \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}). && \text{(G.88)}
\end{aligned}$$

Case 2: $\widehat{y}(\mathbf{x}) \neq y$. To show: $L(\mathbf{x}, y) = \overline{L}(\mathbf{x}, y) + \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|\mathbf{x})} \overline{D}(\mathbf{x})$.

$$\begin{aligned}
& L(\mathbf{x}, y) \\
&= l_{01}(\widehat{y}(\mathbf{x}), y) = 1 && \text{(G.89)} \\
&= 1 - \sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x}) + \sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x}) && \text{(G.90)} \\
&= 1 - \sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x}) \\
&\quad + \frac{(\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x}))}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|\mathbf{x})} \left[1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|\mathbf{x}) \right] && \text{(G.91)} \\
&= 1 - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' = y)\, dy' \\
&\quad + \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|\mathbf{x})} \left[1 - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' = \widehat{y}(\mathbf{x}))\, dy' \right] && \text{(G.92)} \\
&= 1 - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \left[1 - I(y' \neq y) \right]\, dy' \\
&\quad + \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|\mathbf{x})} * \\
&\quad \left[1 - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \left[1 - I(y' \neq \widehat{y}(\mathbf{x})) \right]\, dy' \right] && \text{(G.93)}
\end{aligned}$$

$$
\begin{aligned}
&= 1 - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})\, dy' + \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' \neq y)\, dy' \\
&\quad + \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|\mathbf{x})} * \\
&\quad \left[1 - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})\, dy' + \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' \neq \widehat{y}(\mathbf{x}))\, dy'\right] \quad \text{(G.94)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' \neq y)\, dy' \\
&\quad + \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|\mathbf{x})} (\sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) I(y' \neq \widehat{y}(\mathbf{x}))\, dy') \quad \text{(G.95)} \\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', y)\, dy' \\
&\quad + \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|\mathbf{x})} (\sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', \widehat{y}(\mathbf{x}))\, dy') \quad \text{(G.96)} \\
&= \overline{L}(\mathbf{x}, y) + \frac{\sum_{c=1}^{n} w_c \widehat{p}_c(y|\mathbf{x})}{1 - \sum_{c=1}^{n} w_c \widehat{p}_c(\widehat{y}(\mathbf{x})|\mathbf{x})} \overline{D}(\mathbf{x}). \quad \text{(G.97)}
\end{aligned}
$$

G.95 equals G.94 because $\int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})\, dy' = 1$ for all $\mathbf{x} \in \mathbf{X}$ and $\sum_{c=1}^{n} w_c = 1$, therefore $\sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})\, dy' = 1$ and therefore $\left[1 - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})\, dy'\right] = 0$ for all $\mathbf{x} \in \mathbf{X}$. □

G.9 Proof of Theorem 5.9

Theorem 5.9 (Page 59): For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and 0-1 loss, the expected ensemble loss over the domain can be written as

$$L = \frac{\overline{L} - \overline{D}}{1 - \overline{D}_P - \overline{D}_F} \tag{G.98}$$

Proof. We will show that $L(1 - \overline{D}_P - \overline{D}_F) = \overline{L} - \overline{D}$, from which Equation G.98 can be obtained by a simple rearrangement of the terms.

$$\begin{aligned}
&L(1 - \overline{D}_P - \overline{D}_F) \\
&= L - L\overline{D}_P - L\overline{D}_F && \text{(G.99)} \\
&= \int_{\mathbf{X}\times Y} L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - p(\langle \mathbf{x}, y\rangle \in F) \int_F (1 - \overline{L}(\mathbf{x}, y))p(\langle \mathbf{x}, y\rangle | \langle \mathbf{x}, y\rangle \in F)\, d\langle \mathbf{x}, y\rangle \\
&\quad - p(\langle \mathbf{x}, y\rangle \in F) \int_F \overline{D}(\mathbf{x})p(\langle \mathbf{x}, y\rangle | \langle \mathbf{x}, y\rangle \in F)\, d\langle \mathbf{x}, y\rangle && \text{(G.100)} \\
&= \int_T L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - p(\langle \mathbf{x}, y\rangle \in F) \int_F (1 - \overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x}))p(\langle \mathbf{x}, y\rangle | \langle \mathbf{x}, y\rangle \in F)\, d\langle \mathbf{x}, y\rangle && \text{(G.101)} \\
&= \int_T (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - p(\langle \mathbf{x}, y\rangle \in F) \int_F (1 - \overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x})) \frac{p(\langle \mathbf{x}, y\rangle \wedge \langle \mathbf{x}, y\rangle \in F)}{p(\langle \mathbf{x}, y\rangle \in F)}\, d\langle \mathbf{x}, y\rangle && \text{(G.102)} \\
&= \int_T (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - \int_F (1 - \overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.103)} \\
&= \int_T (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad + \int_F (L(\mathbf{x}, y) - 1 + \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.104)} \\
&= \int_T (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad + \int_F (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.105)} \\
&= \int_T \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - \int_T \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle - \int_F \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.106)} \\
&= \int_{\mathbf{X}\times Y} \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle - \int_{\mathbf{X}\times Y} \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.107)} \\
&= \overline{L} - \overline{D}. && \text{(G.108)}
\end{aligned}$$

G.102 equals G.101 because of Theorem 5.8. □

G.10 Proof of Theorem 5.10

Theorem 5.10 (Page 59): For ensembles $\langle n, \mathbf{c}, \mathbf{w}, V_{\text{dem}} \rangle$ and 0-1 loss, the expected ensemble loss over the domain can be written as

$$L = \frac{\overline{L} - \overline{D}_T}{1 - \overline{D}_T - \overline{D}_P} \tag{G.109}$$

Proof. We will show that $L = \overline{L} - (1 - L)\overline{D}_T + L\overline{D}_P$, from which Equation 5.43 can be obtained by a simple rearrangement of the terms.

$$\begin{aligned}
L &= E_{P(\mathbf{X},Y)}\left[L(\mathbf{x}, y)\right] && \text{(G.110)}\\
&= \int_{\mathbf{X}\times Y} L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.111)}\\
&= \int_T L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F L(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.112)}\\
&= \int_T (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F 1 * p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.113)}\\
&= \int_T (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad + \int_F \left[\overline{L}(\mathbf{x}, y) + (1 - \overline{L}(\mathbf{x}, y))\right] p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.114)}\\
&= \int_T \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - \int_T \overline{D}(\mathbf{x})p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle + \int_F (1 - \overline{L}(\mathbf{x}, y))p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle && \text{(G.115)}\\
&= \int_{\mathbf{X}\times Y} \overline{L}(\mathbf{x}, y)p(\mathbf{x}, y)\, d\langle \mathbf{x}, y\rangle \\
&\quad - p(\langle \mathbf{x}, y\rangle \in T) \int_T \overline{D}(\mathbf{x})p(\langle \mathbf{x}, y\rangle \mid \langle \mathbf{x}, y\rangle \in T)\, d\langle \mathbf{x}, y\rangle \\
&\quad + p(\langle \mathbf{x}, y\rangle \in F) \int_F (1 - \overline{L}(\mathbf{x}, y))p(\langle \mathbf{x}, y\rangle \mid \langle \mathbf{x}, y\rangle \in F)\, d\langle \mathbf{x}, y\rangle && \text{(G.116)}\\
&= \overline{L} - p(\langle \mathbf{x}, y\rangle \in T)\overline{D}_T + p(\langle \mathbf{x}, y\rangle \in F)\overline{D}_P && \text{(G.117)}\\
&= \overline{L} - (1 - L)\overline{D}_T + L\overline{D}_P. && \text{(G.118)}
\end{aligned}$$

G.113 equals G.112 because of Theorem 5.8. □

G.11 Proof of Theorem 5.11

Theorem 5.11 (Page 63): Let l be any triangular loss function. Then, for any given democratic ensemble,

$$L(\mathbf{x}, y) \geq \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}) \tag{G.119}$$

and

$$L \geq \overline{L} - \overline{D}. \tag{G.120}$$

Proof. From the triangle inequality (5.44) follows that

$$\forall y', \widehat{y}(\mathbf{x}), y \in Y : l(y', \widehat{y}(\mathbf{x})) + l(\widehat{y}(\mathbf{x}), y) \geq l(y', y)$$

and therefore

$$\forall y', \widehat{y}(\mathbf{x}), y \in Y : l(\widehat{y}(\mathbf{x}), y) \geq l(y', y) - l(y', \widehat{y}(\mathbf{x})).$$

Hence, $\forall \langle \mathbf{x}, y \rangle \in \mathbf{X} \times Y$:

$$\begin{aligned}
L(\mathbf{x}, y) &= l(\widehat{y}(\mathbf{x}), y) && \text{(G.121)}\\
&= l(\widehat{y}(\mathbf{x}), y) \sum_{c=1}^{n} w_c && \text{(G.122)}\\
&= l(\widehat{y}(\mathbf{x}), y) \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})\, dy' && \text{(G.123)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(\widehat{y}(\mathbf{x}), y)\, dy' && \text{(G.124)}\\
&\geq \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \left[l(y', y) - l(y', \widehat{y}(\mathbf{x})) \right] dy' && \text{(G.125)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', y)\, dy' \\
&\quad - \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', \widehat{y}(\mathbf{x}))\, dy' && \text{(G.126)}\\
&= \overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x}), && \text{(G.127)}
\end{aligned}$$

which proves Equation G.119.

Integrating Equation G.119 over $\mathbf{X} \times Y$ gives Equation G.120:

$$\begin{aligned}
L &= \int_{\mathbf{X} \times Y} L(\mathbf{x}, y) p(\mathbf{x}, y)\, d\langle \mathbf{x}, y \rangle && \text{(G.128)}\\
&\geq \int_{\mathbf{X} \times Y} (\overline{L}(\mathbf{x}, y) - \overline{D}(\mathbf{x})) p(\mathbf{x}, y)\, d\langle \mathbf{x}, y \rangle && \text{(G.129)}\\
&= \int_{\mathbf{X} \times Y} \overline{L}(\mathbf{x}, y) p(\mathbf{x}, y)\, d\langle \mathbf{x}, y \rangle - \int_{\mathbf{X} \times Y} \overline{D}(\mathbf{x}) p(\mathbf{x}, y)\, d\langle \mathbf{x}, y \rangle && \text{(G.130)}\\
&= \overline{L} - \overline{D}. && \text{(G.131)}
\end{aligned}$$

□

G.12 Proof of Theorem 5.12

Theorem 5.12 (Page 63): Let l be any loss function which is both triangular and symmetric. Then, for any given democratic ensemble,

$$L(\mathbf{x}, y) \leq \overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x}) \tag{G.132}$$

and

$$L \leq \overline{L} + \overline{D}. \tag{G.133}$$

Proof. From the triangle inequality (5.44) follows that

$$\forall y, \widehat{y}(\mathbf{x}), y' \in Y : l(y, \widehat{y}(\mathbf{x})) \leq l(y, y') + l(y', \widehat{y}(\mathbf{x})).$$

From the symmetry property (5.45) follows that

$$\forall y, y' \in Y : l(y, y') = l(y', y).$$

Hence, $\forall \langle \mathbf{x}, y \rangle \in \mathbf{X} \times Y$:

$$\begin{aligned}
L(\mathbf{x}, y) &= l(\widehat{y}(\mathbf{x}), y) && \text{(G.134)}\\
&= l(y, \widehat{y}(\mathbf{x})) && \text{(G.135)}\\
&= l(y, \widehat{y}(\mathbf{x})) \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x})\, dy' && \text{(G.136)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y, \widehat{y}(\mathbf{x}))\, dy' && \text{(G.137)}\\
&\leq \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \left[l(y, y') + l(y', \widehat{y}(\mathbf{x})) \right]\, dy' && \text{(G.138)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) \left[l(y', y) + l(y', \widehat{y}(\mathbf{x})) \right]\, dy' && \text{(G.139)}\\
&= \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', y)\, dy' \\
&\quad + \sum_{c=1}^{n} w_c \int_{y' \in Y} \widehat{p}_c(y'|\mathbf{x}) l(y', \widehat{y}(\mathbf{x}))\, dy' && \text{(G.140)}\\
&= \overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x}), && \text{(G.141)}
\end{aligned}$$

which proves (Equation G.132).

Integrating Equation G.132 over $\mathbf{X} \times Y$ gives Equation G.133:

$$\begin{aligned}
L &= \int_{\mathbf{X} \times Y} L(\mathbf{x}, y) p(\mathbf{x}, y)\, d\langle \mathbf{x}, y \rangle && \text{(G.142)}\\
&\leq \int_{\mathbf{X} \times Y} (\overline{L}(\mathbf{x}, y) + \overline{D}(\mathbf{x})) p(\mathbf{x}, y)\, d\langle \mathbf{x}, y \rangle && \text{(G.143)}\\
&= \int_{\mathbf{X} \times Y} \overline{L}(\mathbf{x}, y) p(\mathbf{x}, y)\, d\langle \mathbf{x}, y \rangle + \int_{\mathbf{X} \times Y} \overline{D}(\mathbf{x}) p(\mathbf{x}, y)\, d\langle \mathbf{x}, y \rangle && \text{(G.144)}\\
&= \overline{L} + \overline{D}. && \text{(G.145)}
\end{aligned}$$

□

H. Loss Decomposition Results by Method

Dataset	$\overline{L}$	$\overline{D}$	$\overline{D}_T$	$\overline{D}_F$	$\overline{D}_P$	L	L^*
anneal	0.89±0.10	0.67±0.08	0.59±0.08	5.88±4.01	5.88±4.01	0.39±0.20	0.33
audiology	24.04±0.93	14.61±0.48	10.31±0.61	32.86±2.98	11.85±2.68	17.74±1.52	17.63
autos	26.79±0.78	19.61±0.68	16.13±0.72	36.72±2.71	17.51±1.96	16.07±1.12	16.07
balance	22.14±0.31	13.66±0.33	7.19±0.40	39.56±1.13	17.94±0.79	19.96±0.50	19.97
breastc	36.67±1.07	22.11±0.47	19.82±0.59	27.11±0.76	27.11±0.76	31.58±1.80	31.75
breastw	6.25±0.17	4.25±0.15	3.38±0.15	23.32±3.91	23.32±3.91	4.00±0.20	3.91
colic	17.96±0.26	7.74±0.29	6.43±0.34	14.94±1.23	14.94±1.23	14.67±0.56	14.66
credita	19.15±0.35	11.90±0.37	9.66±0.42	25.09±1.12	25.09±1.12	14.54±0.60	14.54
creditg	32.80±0.34	23.27±0.30	20.51±0.45	31.33±0.64	31.33±0.64	25.44±1.09	25.52
diabetes	29.55±0.34	19.44±0.19	16.44±0.34	28.85±0.70	28.85±0.70	23.96±1.07	23.96
glass	34.73±0.43	24.02±0.58	19.73±0.78	36.46±1.33	20.71±1.33	25.05±1.44	25.18
heartc	25.44±0.62	17.41±0.43	14.49±0.64	29.38±1.70	29.38±1.70	19.49±1.15	19.50
hearth	23.52±0.45	12.86±0.48	8.99±0.62	26.73±1.53	26.73±1.53	22.49±0.81	22.60
hearts	68.86±0.80	45.19±0.90	45.66±1.84	45.17±1.03	17.86±1.04	63.68±2.82	63.61
heartv	25.93±0.66	17.59±0.44	14.62±0.37	28.71±1.99	28.71±1.99	20.07±0.92	19.95
hepatitis	21.81±0.73	12.98±0.61	10.03±0.99	24.73±2.55	24.73±2.55	18.23±1.65	18.06
hypo	0.56±0.03	0.33±0.02	0.27±0.02	12.16±3.89	11.65±3.67	0.34±0.06	0.32
ionosphere	11.29±0.45	7.53±0.43	6.11±0.43	22.65±3.16	22.65±3.16	7.50±0.57	7.27
iris	6.56±0.55	2.74±0.52	1.93±0.46	10.29±4.20	10.25±4.25	5.67±0.79	5.27
krk	24.90±0.07	18.37±0.05	14.60±0.08	37.92±0.15	21.74±0.17	16.18±0.18	16.18
krkp	0.76±0.04	0.57±0.05	0.47±0.06	17.15±3.51	17.15±3.51	0.39±0.07	0.35
labor	19.81±1.73	11.69±1.60	9.54±1.43	14.41±2.97	14.41±2.97	15.10±1.73	13.51
letter	14.27±0.05	12.74±0.09	10.23±0.12	54.00±0.40	19.40±0.11	5.74±0.12	5.74
lymph	23.84±0.79	17.03±0.47	13.61±0.84	30.95±3.30	28.05±2.81	17.65±1.95	17.54
phoneme	14.53±0.12	10.17±0.06	8.08±0.09	28.53±0.39	28.53±0.39	10.17±0.16	10.17
primary	66.83±0.68	38.36±0.66	28.82±1.86	44.54±0.67	8.69±0.66	60.77±1.58	60.83
satimage	15.49±0.05	12.08±0.05	9.53±0.07	37.41±0.36	25.41±0.33	9.17±0.15	9.16
segment	4.31±0.09	3.39±0.07	2.67±0.07	34.36±1.75	24.32±1.58	2.26±0.18	2.24
shuttle	0.04±0.00	0.03±0.00	0.02±0.00	16.20±3.31	14.66±3.09	0.02±0.00	0.02
sick	1.55±0.05	1.07±0.04	0.75±0.05	28.37±2.54	28.37±2.54	1.15±0.13	1.13
sonar	30.23±0.67	24.57±0.50	21.40±0.87	36.53±2.00	36.53±2.00	20.90±2.52	20.98
soybean	11.03±0.35	7.50±0.20	5.42±0.19	30.53±2.19	24.27±2.08	7.99±0.46	7.98
splice	6.25±0.08	1.90±0.10	1.15±0.09	13.90±0.90	11.84±1.01	5.86±0.06	5.86
vehicle	29.87±0.31	22.18±0.33	16.62±0.48	39.06±1.05	29.95±0.67	24.85±0.71	24.80
voting	6.02±0.25	3.21±0.14	2.61±0.17	13.91±2.68	13.91±2.68	4.23±0.36	4.09
waveform	26.01±0.15	20.93±0.09	17.91±0.14	35.57±0.31	34.73±0.31	17.10±0.37	17.11
G. Mean	11.50	7.43	5.86	25.67	20.04	8.19	8.03

Table H.1: Loss decomposition results for *Bagging*$(1;30)$ with Majority Vote.

Dataset	$\overline{L}$	$\overline{D}$	$\overline{D}_T$	$\overline{D}_F$	$\overline{D}_P$	L	L^*
anneal	0.89±0.10	0.67±0.08	0.59±0.08	5.88±4.01	5.88±4.01	0.39±0.20	0.33
audiology	24.11±0.92	14.71±0.48	10.51±0.68	32.92±3.01	11.55±2.68	17.56±1.61	17.45
autos	26.87±0.81	19.75±0.65	16.29±0.69	36.67±2.73	17.57±2.02	16.02±1.13	16.00
balance	22.15±0.31	13.67±0.34	7.20±0.40	39.57±1.12	17.94±0.79	19.96±0.50	19.97
breastc	36.69±1.05	22.26±0.47	19.87±0.62	27.48±0.70	27.48±0.70	31.79±1.87	31.95
breastw	6.33±0.18	4.36±0.15	3.49±0.16	23.39±3.94	23.39±3.94	3.98±0.21	3.88
colic	19.27±0.29	9.77±0.24	8.37±0.30	17.61±0.69	17.61±0.69	14.70±0.44	14.72
credita	19.18±0.35	11.97±0.38	9.76±0.37	25.07±1.24	25.07±1.24	14.46±0.53	14.46
creditg	32.80±0.34	23.27±0.30	20.51±0.45	31.33±0.64	31.33±0.64	25.44±1.09	25.52
diabetes	30.44±0.29	20.85±0.19	17.89±0.44	30.04±0.87	30.04±0.87	24.09±1.20	24.10
glass	34.93±0.42	24.31±0.56	20.03±0.81	36.71±1.43	20.89±1.37	25.09±1.46	25.23
heartc	25.45±0.61	17.44±0.44	14.55±0.67	29.33±1.74	29.33±1.74	19.40±1.20	19.41
hearth	23.97±0.41	14.18±0.41	10.02±0.67	28.89±1.50	28.89±1.50	22.69±0.90	22.83
hearts	69.01±0.81	45.85±0.92	46.40±1.63	45.71±1.17	18.05±1.08	63.70±2.94	63.59
heartv	25.93±0.66	17.59±0.44	14.64±0.40	28.70±1.98	28.70±1.98	20.04±0.93	19.93
hepatitis	21.98±0.68	13.62±0.59	10.59±0.92	26.81±2.83	26.81±2.83	18.11±1.36	18.20
hypo	0.58±0.03	0.35±0.03	0.30±0.02	11.76±6.10	11.22±5.82	0.33±0.07	0.31
ionosphere	11.60±0.46	7.89±0.44	6.50±0.40	22.92±3.13	22.92±3.13	7.44±0.59	7.23
iris	6.66±0.55	2.84±0.54	2.13±0.61	9.07±4.21	9.04±4.24	5.47±0.88	5.09
krk	24.90±0.07	18.38±0.05	14.61±0.09	37.93±0.15	21.74±0.19	16.16±0.18	16.16
krkp	0.76±0.04	0.57±0.05	0.47±0.06	17.15±3.51	17.15±3.51	0.39±0.07	0.35
labor	21.15±1.57	13.25±1.38	11.15±1.24	14.95±3.47	14.95±3.47	15.27±1.73	13.52
letter	14.27±0.05	12.74±0.09	10.23±0.12	54.00±0.40	19.40±0.11	5.74±0.12	5.74
lymph	23.84±0.79	17.03±0.47	13.61±0.84	30.95±3.30	28.05±2.81	17.65±1.95	17.54
phoneme	15.68±0.11	11.77±0.08	9.71±0.07	30.38±0.43	30.38±0.43	9.96±0.19	9.96
primary	68.29±0.64	42.05±0.64	33.23±1.60	47.85±0.76	9.03±0.53	60.68±1.29	60.73
satimage	15.52±0.05	12.11±0.05	9.55±0.07	37.47±0.44	25.48±0.39	9.19±0.17	9.19
segment	4.36±0.09	3.44±0.07	2.75±0.07	33.94±1.95	23.84±1.67	2.21±0.18	2.19
shuttle	0.04±0.00	0.03±0.00	0.02±0.00	15.97±3.54	14.43±3.29	0.02±0.00	0.02
sick	1.58±0.05	1.11±0.04	0.79±0.05	28.04±2.27	28.04±2.27	1.13±0.11	1.11
sonar	30.39±0.66	24.77±0.49	21.84±0.91	36.21±2.11	36.21±2.11	20.33±2.57	20.38
soybean	11.46±0.35	7.93±0.20	5.86±0.21	30.79±2.26	24.50±2.05	8.05±0.49	8.04
splice	6.25±0.08	1.90±0.10	1.15±0.09	13.90±0.90	11.84±1.01	5.86±0.06	5.86
vehicle	29.88±0.31	22.23±0.32	16.70±0.47	39.10±1.10	29.92±0.76	24.74±0.57	24.68
voting	6.55±0.25	4.27±0.17	3.43±0.19	19.53±2.98	19.53±2.98	4.25±0.38	4.05
waveform	26.12±0.15	21.06±0.09	18.05±0.12	35.64±0.32	34.80±0.32	17.12±0.35	17.13
G. Mean	11.68	7.74	6.17	26.20	20.41	8.16	8.00

Table H.2: Loss decomposition results for $Bagging(1;30)$.

Dataset	$\overline{L}$	$\overline{D}$	$\overline{D}_T$	$\overline{D}_F$	$\overline{D}_P$	L	L^*
anneal	1.62±0.09	1.28±0.12	1.09±0.13	13.23±3.74	12.63±3.48	0.72±0.15	0.62
audiology	30.42±0.72	22.38±0.65	17.21±0.57	43.49±2.99	14.79±1.86	19.41±1.23	19.43
autos	36.28±0.72	29.45±0.92	25.64±0.80	44.27±1.97	20.79±1.95	19.88±1.03	19.86
balance	23.11±0.23	18.02±0.13	12.32±0.45	47.74±0.80	20.33±1.05	16.05±0.99	16.01
breastc	37.42±0.62	25.67±0.59	23.36±0.55	30.96±1.16	30.96±1.16	30.54±1.19	30.78
breastw	6.48±0.23	4.70±0.17	3.77±0.22	23.93±3.74	23.93±3.74	3.93±0.43	3.75
colic	20.15±0.28	11.04±0.37	9.60±0.36	19.07±1.81	19.07±1.81	14.77±0.39	14.79
credita	19.53±0.31	12.89±0.19	10.70±0.49	26.17±1.38	26.17±1.38	13.99±0.98	13.98
creditg	33.32±0.28	24.57±0.23	21.83±0.27	32.64±0.43	32.64±0.43	25.21±0.56	25.24
diabetes	31.10±0.32	22.42±0.34	19.52±0.39	31.68±0.67	31.68±0.67	23.73±0.79	23.73
glass	39.02±0.95	29.95±0.92	25.74±0.99	41.58±1.87	22.41±1.76	25.39±1.88	25.62
heartc	25.86±0.70	18.73±0.45	16.02±0.43	30.83±1.28	30.83±1.28	18.45±1.48	18.52
hearth	23.92±0.31	15.65±0.40	11.81±0.58	30.90±0.99	30.90±0.99	20.89±0.88	21.13
hearts	70.10±0.35	50.45±0.91	50.08±2.16	50.33±0.87	19.55±1.11	64.88±3.27	65.92
heartv	26.17±0.84	19.18±0.59	16.62±0.49	30.63±1.18	30.63±1.18	18.19±1.24	18.11
hepatitis	22.55±0.64	15.30±0.51	12.56±0.80	25.88±4.54	25.88±4.54	16.75±1.81	16.23
hypo	0.81±0.04	0.59±0.05	0.51±0.04	16.20±5.43	14.01±5.44	0.37±0.06	0.35
ionosphere	13.23±0.31	9.40±0.28	8.05±0.19	23.72±3.10	23.72±3.10	7.78±0.49	7.60
iris	7.28±0.41	3.99±0.48	2.86±0.48	13.52±2.64	13.45±2.56	5.87±0.93	5.28
krk	31.84±0.08	25.92±0.06	21.27±0.08	45.05±0.09	24.72±0.12	19.57±0.15	19.58
krkp	1.28±0.08	1.05±0.07	0.84±0.06	27.70±6.09	27.70±6.09	0.65±0.10	0.61
labor	23.99±1.09	18.54±1.36	15.94±1.52	18.29±3.54	18.29±3.54	15.00±2.17	12.24
letter	17.42±0.02	16.01±0.03	13.25±0.07	57.88±0.24	19.42±0.30	6.19±0.08	6.19
lymph	26.16±0.59	20.14±0.72	17.01±1.02	32.31±2.09	29.01±1.58	17.12±1.36	16.95
phoneme	18.77±0.08	14.60±0.13	12.22±0.15	32.38±0.40	32.38±0.40	11.82±0.20	11.82
primary	69.50±0.55	49.94±0.82	40.41±1.75	56.67±1.07	9.98±0.54	58.59±1.57	58.64
satimage	16.50±0.07	13.10±0.07	10.36±0.12	38.86±0.41	25.74±0.31	9.60±0.21	9.60
segment	5.72±0.14	4.69±0.13	3.77±0.18	37.33±1.30	26.92±1.34	2.81±0.18	2.81
shuttle	0.06±0.00	0.04±0.00	0.03±0.00	29.73±2.29	25.40±2.56	0.04±0.00	0.04
sick	1.98±0.05	1.43±0.05	1.06±0.08	28.85±1.53	28.85±1.53	1.31±0.12	1.30
sonar	32.73±0.74	27.11±0.83	24.22±0.98	36.81±1.67	36.81±1.67	22.15±1.83	21.84
soybean	15.31±0.37	12.44±0.36	10.25±0.29	38.24±0.98	27.20±1.36	7.98±0.39	8.08
splice	6.97±0.12	2.89±0.10	1.89±0.12	18.16±1.07	15.53±1.34	6.15±0.29	6.15
vehicle	32.31±0.34	25.61±0.33	19.84±0.58	42.54±0.70	31.26±0.63	25.50±1.28	25.50
voting	6.79±0.26	4.84±0.23	4.11±0.23	17.27±3.93	17.27±3.93	3.73±0.44	3.41
waveform	26.61±0.12	21.90±0.13	19.05±0.15	36.21±0.26	35.29±0.24	16.55±0.37	16.55
G. Mean	13.43	9.79	8.03	30.44	23.24	8.73	8.55

Table H.3: Loss decomposition results for *Bagging*$(0.5; 30)$.

Dataset	$\overline{L}$	$\overline{D}$	$\overline{D}_T$	$\overline{D}_F$	$\overline{D}_P$	L	L^*
anneal	0.57±0.15	0.35±0.07	0.32±0.08	2.27±2.10	2.27±2.10	0.28±0.14	0.25
audiology	20.63±1.05	8.55±0.59	4.82±0.68	24.01±2.89	10.36±1.88	18.71±1.65	18.64
autos	20.61±0.99	12.51±0.61	9.13±0.39	29.15±3.39	14.05±2.03	15.15±1.42	14.94
balance	22.08±0.39	9.45±0.32	4.45±0.38	28.18±1.26	11.85±1.20	21.08±0.74	21.07
breastc	35.60±0.99	18.20±0.56	16.26±0.76	22.30±1.05	22.30±1.05	31.44±1.23	31.47
breastw	6.22±0.21	4.09±0.16	3.14±0.13	24.26±3.93	24.26±3.93	4.31±0.30	4.25
colic	18.64±0.25	8.86±0.29	7.55±0.28	16.37±1.18	16.37±1.18	14.50±0.35	14.57
credita	19.20±0.38	10.98±0.38	8.77±0.29	22.91±1.34	22.91±1.34	15.32±0.63	15.27
creditg	32.25±0.31	21.48±0.18	18.62±0.33	29.48±0.32	29.48±0.32	26.21±0.88	26.25
diabetes	30.22±0.32	19.57±0.20	16.60±0.42	28.51±0.61	28.51±0.61	24.82±0.76	24.81
glass	32.18±0.77	19.88±0.54	15.51±0.72	32.28±0.75	19.59±1.12	25.46±0.94	25.69
heartc	26.22±0.77	16.46±0.49	13.33±0.46	27.65±1.60	27.65±1.60	21.85±1.69	21.85
hearth	24.21±0.54	12.06±0.32	8.99±0.31	23.17±1.49	23.17±1.49	22.34±1.07	22.44
hearts	68.78±1.02	40.54±1.07	39.82±2.18	41.28±1.43	16.16±0.86	65.95±2.60	65.78
heartv	26.31±0.59	16.43±0.47	13.21±0.80	27.33±2.13	27.33±2.13	22.07±1.64	22.04
hepatitis	21.89±0.82	12.08±0.66	9.55±0.86	23.14±2.55	23.14±2.55	18.24±0.70	18.33
hypo	0.47±0.04	0.21±0.03	0.18±0.03	7.90±1.90	7.86±1.86	0.34±0.05	0.32
ionosphere	10.60±0.28	6.95±0.29	5.48±0.37	22.71±2.41	22.71±2.41	7.38±0.45	7.14
iris	6.09±0.62	1.73±0.35	1.31±0.18	5.59±3.47	5.59±3.47	5.33±0.77	5.13
krk	20.34±0.13	12.02±0.05	8.88±0.07	29.06±0.18	17.46±0.19	15.55±0.19	15.55
krkp	0.46±0.04	0.24±0.04	0.20±0.04	7.48±3.73	7.48±3.73	0.31±0.08	0.29
labor	19.57±1.83	10.14±1.35	8.53±1.53	11.09±1.83	11.09±1.83	14.77±2.22	13.73
letter	12.46±0.05	10.59±0.06	8.05±0.09	50.05±0.32	19.15±0.24	6.04±0.09	6.04
lymph	22.41±1.07	13.82±0.65	10.66±1.34	26.14±3.48	24.52±3.35	18.31±2.12	18.12
phoneme	13.54±0.09	9.46±0.10	7.63±0.09	27.68±0.71	27.68±0.71	9.15±0.20	9.14
primary	67.82±0.86	34.14±0.70	27.50±1.42	38.28±1.23	7.47±0.52	61.94±1.74	62.01
satimage	14.87±0.13	11.43±0.11	8.93±0.05	36.51±0.60	25.24±0.50	9.02±0.18	9.02
segment	3.52±0.11	2.55±0.10	1.98±0.11	30.69±2.64	21.17±2.20	2.00±0.21	2.00
shuttle	0.03±0.00	0.01±0.00	0.01±0.00	13.20±3.12	12.43±2.47	0.02±0.00	0.02
sick	1.35±0.07	0.81±0.03	0.55±0.03	22.64±2.90	22.64±2.90	1.07±0.11	1.04
sonar	29.31±0.78	22.97±0.77	19.65±1.02	34.84±2.28	34.84±2.28	21.25±2.01	21.22
soybean	10.01±0.23	5.41±0.21	3.59±0.24	25.20±3.01	20.51±2.48	8.45±0.48	8.46
splice	6.73±0.13	2.48±0.18	1.89±0.18	12.45±0.53	10.75±0.67	5.55±0.14	5.54
vehicle	28.23±0.49	19.14±0.43	13.97±0.50	35.14±1.00	27.45±0.83	24.39±1.56	24.35
voting	6.47±0.22	3.64±0.14	2.82±0.11	16.85±3.57	16.85±3.57	4.73±0.41	4.54
waveform	25.98±0.12	20.55±0.06	17.48±0.13	34.96±0.27	34.13±0.29	17.56±0.38	17.57
G. Mean	10.63	6.07	4.70	21.57	17.05	8.13	8.02

Table H.4: Loss decomposition results for *Bagging*$(2; 30)$.

Dataset	$\overline{L}$	$\overline{D}$	$\overline{D}_T$	$\overline{D}_F$	$\overline{D}_P$	L	L^*
anneal	1.23±0.07	0.99±0.10	0.86±0.08	10.55±4.31	10.38±4.50	0.51±0.13	0.42
audiology	26.86±0.56	18.61±0.42	14.52±0.65	37.93±3.84	11.79±1.74	16.74±0.87	16.76
autos	31.29±0.69	24.18±0.58	20.82±0.87	39.34±2.49	19.00±2.69	17.20±1.65	17.39
balance	22.41±0.34	16.10±0.29	9.58±0.41	45.47±0.71	20.15±0.72	18.23±0.80	18.26
breastc	37.12±0.79	24.49±0.50	22.36±0.74	29.43±0.85	29.43±0.85	30.49±1.31	30.63
breastw	6.52±0.17	4.55±0.14	3.73±0.18	22.85±4.53	22.85±4.53	3.91±0.35	3.80
colic	19.56±0.23	10.34±0.19	9.03±0.28	17.67±1.43	17.67±1.43	14.40±0.62	14.37
credita	19.45±0.28	12.27±0.22	10.10±0.28	25.12±1.07	25.12±1.07	14.43±0.68	14.43
creditg	33.34±0.32	23.97±0.10	21.08±0.16	32.10±0.35	32.10±0.35	26.16±0.53	26.18
diabetes	31.70±0.27	23.28±0.27	20.29±0.30	32.61±0.49	32.61±0.49	24.18±0.56	24.22
glass	37.14±0.68	27.34±0.64	23.58±0.68	37.86±2.04	21.11±1.48	24.44±1.27	24.51
heartc	25.24±0.61	17.39±0.45	14.85±0.52	28.36±2.07	28.36±2.07	18.32±1.14	18.30
hearth	23.75±0.39	15.36±0.33	10.88±0.91	31.65±1.66	31.65±1.66	22.18±1.72	22.40
hearts	70.19±1.15	48.14±0.71	48.88±1.90	47.51±1.03	18.39±0.85	64.38±3.16	65.10
heartv	25.36±0.87	17.93±0.70	15.34±1.00	29.17±2.17	29.17±2.17	18.11±1.73	18.05
hepatitis	22.07±0.86	14.09±0.50	11.01±0.53	26.42±2.65	26.42±2.65	17.99±0.98	17.67
hypo	0.68±0.04	0.47±0.03	0.41±0.02	14.88±4.96	13.90±5.01	0.34±0.04	0.32
ionosphere	12.48±0.40	8.49±0.31	7.13±0.45	23.46±2.55	23.46±2.55	7.78±0.76	7.71
iris	7.02±0.44	3.48±0.28	2.58±0.33	10.63±3.26	10.53±3.24	5.60±0.64	5.10
krk	28.10±0.07	22.06±0.03	17.78±0.06	41.98±0.10	23.83±0.14	17.69±0.14	17.69
krkp	0.97±0.05	0.76±0.05	0.60±0.05	23.16±5.34	23.16±5.34	0.54±0.07	0.48
labor	23.28±1.34	16.56±1.09	14.21±1.90	17.85±5.20	17.85±5.20	15.63±2.15	13.36
letter	15.63±0.04	14.14±0.07	11.50±0.11	55.74±0.28	19.25±0.22	5.96±0.14	5.96
lymph	24.86±0.88	18.54±0.76	15.44±0.81	32.57±2.20	29.25±2.54	16.92±1.00	17.04
phoneme	19.88±0.07	15.36±0.08	12.87±0.08	32.32±0.23	32.32±0.23	12.80±0.17	12.80
primary	68.95±0.42	46.96±0.49	37.45±1.00	53.45±0.67	9.79±0.54	59.73±1.64	59.71
satimage	15.85±0.08	12.43±0.08	9.78±0.10	38.00±0.41	25.70±0.41	9.41±0.17	9.41
segment	5.03±0.11	4.09±0.08	3.28±0.09	36.26±1.90	26.19±1.73	2.48±0.18	2.47
shuttle	0.05±0.00	0.03±0.00	0.02±0.00	22.26±2.59	19.36±3.22	0.03±0.00	0.03
sick	1.78±0.04	1.29±0.03	0.93±0.05	29.39±1.82	29.39±1.82	1.23±0.13	1.22
sonar	30.75±0.66	25.43±0.59	22.48±0.83	37.51±1.26	37.51±1.26	20.31±1.78	20.67
soybean	13.03±0.27	9.99±0.22	7.91±0.30	34.88±2.43	26.23±1.97	7.71±0.56	7.76
splice	6.03±0.06	2.04±0.06	1.40±0.10	13.19±1.07	10.86±0.99	5.29±0.15	5.28
vehicle	30.99±0.25	24.27±0.39	18.32±0.56	41.64±1.06	32.02±0.61	25.51±0.89	25.51
voting	6.64±0.17	4.66±0.15	4.02±0.15	17.06±4.24	17.06±4.24	3.54±0.27	3.32
waveform	25.97±0.12	20.90±0.11	17.96±0.13	35.39±0.28	34.55±0.27	16.86±0.25	16.87
G. Mean	12.53	8.77	7.14	28.44	21.91	8.37	8.21

Table H.5: Loss decomposition results for *Cragging*$(2; 15)$.

Dataset	$\overline{L}$	$\overline{D}$	$\overline{D}_T$	$\overline{D}_F$	$\overline{D}_P$	L	L^*
anneal	0.80±0.12	0.57±0.07	0.56±0.07	1.00±1.76	1.00±1.76	0.26±0.16	0.24
audiology	23.26±0.84	13.49±0.27	9.60±0.46	30.45±3.13	11.47±2.00	17.38±1.20	17.31
autos	25.44±0.69	17.47±0.35	14.57±0.40	31.55±2.82	15.41±1.88	15.49±1.27	15.53
balance	21.80±0.34	12.63±0.30	6.36±0.30	37.66±0.73	16.80±0.72	20.09±0.44	20.09
breastc	36.32±0.74	21.05±0.52	19.11±0.89	25.22±1.34	25.22±1.34	30.86±1.02	30.91
breastw	6.36±0.13	4.28±0.07	3.43±0.18	22.85±2.59	22.85±2.59	4.05±0.28	3.97
colic	19.23±0.31	9.49±0.29	8.36±0.53	15.69±1.37	15.69±1.37	14.34±0.79	14.32
credita	19.28±0.36	11.17±0.37	9.12±0.44	22.79±1.33	22.79±1.33	14.93±1.10	14.93
creditg	33.13±0.17	22.68±0.15	19.71±0.30	30.70±0.40	30.70±0.40	27.02±0.73	27.06
diabetes	31.33±0.24	22.58±0.21	19.65±0.29	31.84±0.30	31.84±0.30	24.03±0.45	24.07
glass	34.95±0.94	24.13±0.59	19.74±0.59	36.48±2.14	21.35±1.77	25.74±1.88	25.81
heartc	25.06±0.41	16.21±0.43	13.66±0.35	26.96±1.27	26.96±1.27	19.18±1.04	19.20
hearth	24.09±0.60	14.14±0.40	9.90±0.68	28.84±1.34	28.84±1.34	23.03±1.47	23.17
hearts	69.95±0.67	43.85±0.98	44.95±1.93	42.98±1.09	16.69±0.83	64.56±1.74	65.18
heartv	24.93±0.51	16.15±0.49	13.37±0.74	27.41±1.15	27.41±1.15	19.56±1.21	19.52
hepatitis	21.87±0.84	12.63±0.62	9.49±0.91	24.67±2.83	24.67±2.83	18.93±1.53	18.80
hypo	0.55±0.04	0.32±0.02	0.27±0.03	11.93±3.54	11.41±3.61	0.33±0.05	0.32
ionosphere	11.82±0.25	7.65±0.27	6.17±0.30	23.59±2.93	23.59±2.93	8.12±0.60	8.04
iris	6.90±0.44	2.84±0.33	1.98±0.37	10.33±3.30	10.27±3.32	6.00±0.44	5.60
krk	23.83±0.10	16.79±0.05	13.17±0.08	35.67±0.20	20.69±0.14	16.11±0.16	16.11
krkp	0.68±0.05	0.48±0.04	0.40±0.03	14.57±3.92	14.57±3.92	0.37±0.06	0.34
labor	21.50±1.95	13.10±1.10	10.85±1.26	16.34±3.11	16.34±3.11	16.13±2.38	14.63
letter	13.75±0.05	12.03±0.05	9.41±0.07	52.46±0.24	19.16±0.23	6.07±0.11	6.08
lymph	23.76±0.93	15.92±0.84	12.57±0.95	29.65±2.39	27.11±2.40	18.49±1.67	18.55
phoneme	18.69±0.09	14.08±0.10	11.72±0.11	31.11±0.20	31.11±0.20	12.19±0.15	12.19
primary	68.26±0.70	40.74±0.62	32.47±0.83	46.11±1.02	8.82±0.53	60.91±1.57	60.97
satimage	15.18±0.07	11.71±0.08	9.13±0.11	36.97±0.48	25.62±0.47	9.27±0.17	9.27
segment	4.21±0.09	3.25±0.09	2.59±0.11	32.51±2.62	22.66±2.43	2.18±0.22	2.16
shuttle	0.04±0.00	0.02±0.00	0.01±0.00	15.42±1.68	14.26±1.82	0.03±0.00	0.03
sick	1.54±0.04	1.03±0.04	0.72±0.07	27.57±2.83	27.57±2.83	1.16±0.16	1.15
sonar	29.27±0.69	23.43±0.62	20.43±0.88	34.64±2.25	34.64±2.25	19.85±2.03	19.66
soybean	11.19±0.29	7.45±0.23	5.28±0.24	30.80±2.35	24.98±2.22	8.45±0.51	8.47
splice	5.53±0.05	1.34±0.09	0.83±0.09	10.62±0.54	8.66±0.60	5.20±0.08	5.20
vehicle	29.29±0.40	22.14±0.22	16.29±0.27	39.80±0.80	31.80±0.82	25.05±1.08	25.05
voting	6.55±0.23	4.27±0.17	3.39±0.18	19.10±4.18	19.10±4.18	4.32±0.48	4.07
waveform	25.59±0.15	20.15±0.09	17.13±0.11	34.63±0.18	33.81±0.19	17.25±0.38	17.25
G. Mean	11.50	7.29	5.77	24.02	18.82	8.22	8.11

Table H.6: Loss decomposition results for *Cragging*(3; 10).

Dataset	$\overline{L}$	$\overline{D}$	$\overline{D}_T$	$\overline{D}_F$	$\overline{D}_P$	L	L^*
anneal	0.27±0.16	0.06±0.03	0.06±0.03	0.10±0.32	0.10±0.32	0.21±0.15	0.21
audiology	19.30±1.19	2.63±0.33	1.27±0.24	8.11±1.40	3.28±0.63	18.88±1.41	18.89
autos	17.87±1.34	5.05±0.46	3.31±0.40	13.66±2.06	8.11±1.67	16.50±1.65	16.44
balance	22.07±0.44	4.10±0.35	1.48±0.20	13.33±1.19	5.15±1.01	22.06±0.55	22.05
breastc	34.98±1.13	10.45±0.97	8.55±0.92	13.88±1.85	13.88±1.85	34.02±1.89	34.07
breastw	6.26±0.35	2.51±0.15	1.67±0.15	17.15±2.28	17.15±2.28	5.61±0.61	5.66
colic	18.67±0.30	6.66±0.45	5.46±0.47	12.73±1.71	12.73±1.71	16.14±0.64	16.14
credita	19.53±0.73	5.91±0.23	4.10±0.32	13.81±0.96	13.81±0.96	18.75±1.02	18.79
creditg	32.86±0.59	16.13±0.43	13.22±0.41	22.74±1.20	22.74±1.20	30.63±1.26	30.67
diabetes	30.69±0.26	19.32±0.43	16.74±0.45	27.14±0.80	27.14±0.80	24.82±0.80	24.86
glass	33.41±1.20	12.69±0.68	9.27±0.79	20.23±1.40	14.39±1.30	31.50±1.76	31.63
heartc	25.03±1.18	8.37±1.05	6.29±0.89	14.91±2.01	14.91±2.01	23.78±1.63	23.78
hearth	24.27±0.79	10.22±0.44	7.38±0.87	19.70±2.10	19.70±2.10	23.09±1.65	23.15
hearts	70.22±1.32	27.64±1.44	28.65±2.43	27.21±1.68	10.97±1.20	68.79±2.35	68.85
heartv	25.94±1.00	8.21±0.89	5.97±0.93	14.57±1.94	14.57±1.94	25.11±1.72	25.13
hepatitis	22.55±1.28	6.15±0.89	3.81±0.82	14.73±3.79	14.73±3.79	22.83±1.38	23.00
hypo	0.42±0.04	0.08±0.01	0.06±0.01	3.85±2.64	3.85±2.64	0.38±0.05	0.38
ionosphere	10.87±0.62	4.66±0.59	3.56±0.51	14.98±3.68	14.98±3.68	9.03±1.06	8.98
iris	6.49±0.95	0.96±0.26	0.76±0.21	1.96±1.12	1.96±1.12	6.00±1.18	5.90
krk	18.49±0.19	4.88±0.06	3.06±0.04	13.46±0.27	8.95±0.19	17.54±0.21	17.54
krkp	0.36±0.05	0.06±0.01	0.05±0.01	2.36±1.20	2.36±1.20	0.33±0.06	0.32
labor	19.66±2.36	5.93±0.82	5.28±0.98	5.54±1.52	5.54±1.52	16.63±2.52	16.13
letter	11.68±0.06	7.05±0.26	4.13±0.21	35.40±1.35	15.88±0.56	9.42±0.22	9.44
lymph	22.74±1.90	5.79±0.84	3.80±0.85	12.32±3.40	11.73±3.20	22.40±2.86	22.42
phoneme	17.09±0.28	10.87±0.20	8.84±0.18	25.13±0.50	25.13±0.50	12.48±0.32	12.49
primary	67.59±0.99	23.99±0.60	20.29±1.64	26.24±1.33	5.51±0.43	63.80±1.29	63.76
satimage	14.36±0.16	9.37±0.10	6.73±0.14	31.29±0.68	22.36±0.53	10.76±0.29	10.77
segment	3.17±0.13	1.51±0.10	0.96±0.11	21.87±2.77	16.45±1.99	2.68±0.23	2.68
shuttle	0.02±0.00	0.00±0.00	0.00±0.00	4.11±2.51	4.03±2.54	0.02±0.00	0.02
sick	1.37±0.09	0.31±0.02	0.19±0.02	9.53±1.50	9.53±1.50	1.31±0.10	1.31
sonar	27.60±1.32	12.94±0.99	9.94±0.73	22.08±3.27	22.08±3.27	25.65±2.60	25.98
soybean	9.47±0.41	2.47±0.20	1.72±0.11	10.81±2.64	8.71±2.18	8.62±0.41	8.66
splice	5.34±0.07	0.40±0.05	0.19±0.05	4.00±0.53	3.51±0.56	5.35±0.11	5.35
vehicle	26.88±0.81	13.01±0.53	8.99±0.58	24.70±1.36	21.29±1.12	25.69±1.25	25.66
voting	6.45±0.37	2.16±0.17	1.50±0.11	11.34±2.68	11.34±2.68	5.79±0.41	5.68
waveform	25.29±0.13	17.40±0.26	14.16±0.30	30.51±0.63	29.75±0.65	19.81±0.42	19.83
G. Mean	10.09	3.16	2.24	11.53	9.33	9.14	9.12

Table H.7: Loss decomposition results for *Cragging*(30; 1).

I. Loss Decomposition Results by Variable

Dataset	B(.5;30)	B(1;30)	B(2;30)	C(2;15)	C(3;10)	C(30;1)
anneal	0.72±0.15	0.39±0.20	0.28±0.14	0.51±0.13	0.26±0.16	**0.21**±0.15
audiology	19.41±1.23	17.56±1.61	18.71±1.65	**16.74**±0.87	17.38±1.20	18.88±1.41
autos	19.88±1.03	16.02±1.13	**15.15**±1.42	17.20±1.65	15.49±1.27	16.50±1.65
balance	**16.05**±0.99	19.96±0.50	21.08±0.74	18.23±0.80	20.09±0.44	22.06±0.55
breastc	30.54±1.19	31.79±1.87	31.44±1.23	**30.49**±1.31	30.86±1.02	34.02±1.89
breastw	3.93±0.43	3.98±0.21	4.31±0.30	**3.91**±0.35	4.05±0.28	5.61±0.61
colic	14.77±0.39	14.70±0.44	14.50±0.35	14.40±0.62	**14.34**±0.79	16.14±0.64
credita	**13.99**±0.98	14.46±0.53	15.32±0.63	14.43±0.68	14.93±1.10	18.75±1.02
creditg	**25.21**±0.56	25.44±1.09	26.21±0.88	26.16±0.53	27.02±0.73	30.63±1.26
diabetes	**23.73**±0.79	24.09±1.20	24.82±0.76	24.18±0.56	24.03±0.45	24.82±0.80
glass	25.39±1.88	25.09±1.46	25.46±0.94	**24.44**±1.27	25.74±1.88	31.50±1.76
heartc	18.45±1.48	19.40±1.20	21.85±1.69	**18.32**±1.14	19.18±1.04	23.78±1.63
hearth	**20.89**±0.88	22.69±0.90	22.34±1.07	22.18±1.72	23.03±1.47	23.09±1.65
hearts	64.88±3.27	**63.70**±2.94	65.95±2.60	64.38±3.16	64.56±1.74	68.79±2.35
heartv	18.19±1.24	20.04±0.93	22.07±1.64	**18.11**±1.73	19.56±1.21	25.11±1.72
hepatitis	**16.75**±1.81	18.11±1.36	18.24±0.70	17.99±0.98	18.93±1.53	22.83±1.38
hypo	0.37±0.06	**0.33**±0.07	0.34±0.05	0.34±0.04	0.33±0.05	0.38±0.05
ionosphere	7.78±0.49	7.44±0.59	**7.38**±0.45	7.78±0.76	8.12±0.60	9.03±1.06
iris	5.87±0.93	5.47±0.88	**5.33**±0.77	5.60±0.64	6.00±0.44	6.00±1.18
krk	19.57±0.15	16.16±0.18	**15.55**±0.19	17.69±0.14	16.11±0.16	17.54±0.21
krkp	0.65±0.10	0.39±0.07	**0.31**±0.08	0.54±0.07	0.37±0.06	0.33±0.06
labor	15.00±2.17	15.27±1.73	**14.77**±2.22	15.63±2.15	16.13±2.38	16.63±2.52
letter	6.19±0.08	**5.74**±0.12	6.04±0.09	5.96±0.14	6.07±0.11	9.42±0.22
lymph	17.12±1.36	17.65±1.95	18.31±2.12	**16.92**±1.00	18.49±1.67	22.40±2.86
phoneme	11.82±0.20	9.96±0.19	**9.15**±0.20	12.80±0.17	12.19±0.15	12.48±0.32
primary	**58.59**±1.57	60.68±1.29	61.94±1.74	59.73±1.64	60.91±1.57	63.80±1.29
satimage	9.60±0.21	9.19±0.17	**9.02**±0.18	9.41±0.17	9.27±0.17	10.76±0.29
segment	2.81±0.18	2.21±0.18	**2.00**±0.21	2.48±0.18	2.18±0.22	2.68±0.23
shuttle	0.04±0.00	0.02±0.00	0.02±0.00	0.03±0.00	0.03±0.00	**0.02**±0.00
sick	1.31±0.12	1.13±0.11	**1.07**±0.11	1.23±0.13	1.16±0.16	1.31±0.10
sonar	22.15±1.83	20.33±2.57	21.25±2.01	20.31±1.78	**19.85**±2.03	25.65±2.60
soybean	7.98±0.39	8.05±0.49	8.45±0.48	**7.71**±0.56	8.45±0.51	8.62±0.41
splice	6.15±0.29	5.86±0.06	5.55±0.14	5.29±0.15	**5.20**±0.08	5.35±0.11
vehicle	25.50±1.28	24.74±0.57	**24.39**±1.56	25.51±0.89	25.05±1.08	25.69±1.25
voting	3.73±0.44	4.25±0.38	4.73±0.41	**3.54**±0.27	4.32±0.48	5.79±0.41
waveform	**16.55**±0.37	17.12±0.35	17.56±0.38	16.86±0.25	17.25±0.38	19.81±0.42
G. Mean	8.73	8.16	8.13	8.37	8.22	9.14

Table I.1: Comparison of L.

Dataset	B(.5;30)	B(1;30)	B(2;30)	C(2;15)	C(3;10)	C(30;1)
anneal	1.62±0.09	0.89±0.10	0.57±0.15	1.23±0.07	0.80±0.12	**0.27**±0.16
audiology	30.42±0.72	24.11±0.92	20.63±1.05	**26.86**±0.56	23.26±0.84	19.30±1.19
autos	36.28±0.72	26.87±0.81	**20.61**±0.99	31.29±0.69	25.44±0.69	17.87±1.34
balance	**23.11**±0.23	22.15±0.31	22.08±0.39	22.41±0.34	21.80±0.34	22.07±0.44
breastc	37.42±0.62	36.69±1.05	35.60±0.99	**37.12**±0.79	36.32±0.74	34.98±1.13
breastw	6.48±0.23	6.33±0.18	6.22±0.21	**6.52**±0.17	6.36±0.13	6.26±0.35
colic	20.15±0.28	19.27±0.29	18.64±0.25	19.56±0.23	**19.23**±0.31	18.67±0.30
credita	**19.53**±0.31	19.18±0.35	19.20±0.38	19.45±0.28	19.28±0.36	19.53±0.73
creditg	**33.32**±0.28	32.80±0.34	32.25±0.31	33.34±0.32	33.13±0.17	32.86±0.59
diabetes	**31.10**±0.32	30.44±0.29	30.22±0.32	31.70±0.27	31.33±0.24	30.69±0.26
glass	39.02±0.95	34.93±0.42	32.18±0.77	**37.14**±0.68	34.95±0.94	33.41±1.20
heartc	25.86±0.70	25.45±0.61	26.22±0.77	**25.24**±0.61	25.06±0.41	25.03±1.18
hearth	**23.92**±0.31	23.97±0.41	24.21±0.54	23.75±0.39	24.09±0.60	24.27±0.79
hearts	70.10±0.35	**69.01**±0.81	68.78±1.02	70.19±1.15	69.95±0.67	70.22±1.32
heartv	26.17±0.84	25.93±0.66	26.31±0.59	**25.36**±0.87	24.93±0.51	25.94±1.00
hepatitis	**22.55**±0.64	21.98±0.68	21.89±0.82	22.07±0.86	21.87±0.84	22.55±1.28
hypo	0.81±0.04	**0.58**±0.03	0.47±0.04	0.68±0.04	0.55±0.04	0.42±0.04
ionosphere	13.23±0.31	11.60±0.46	**10.60**±0.28	12.48±0.40	11.82±0.25	10.87±0.62
iris	7.28±0.41	6.66±0.55	**6.09**±0.62	7.02±0.44	6.90±0.44	6.49±0.95
krk	31.84±0.08	24.90±0.07	**20.34**±0.13	28.10±0.07	23.83±0.10	18.49±0.19
krkp	1.28±0.08	0.76±0.04	**0.46**±0.04	0.97±0.05	0.68±0.05	0.36±0.05
labor	23.99±1.09	21.15±1.57	**19.57**±1.83	23.28±1.34	21.50±1.95	19.66±2.36
letter	17.42±0.02	**14.27**±0.05	12.46±0.05	15.63±0.04	13.75±0.05	11.68±0.06
lymph	26.16±0.59	23.84±0.79	22.41±1.07	**24.86**±0.88	23.76±0.93	22.74±1.90
phoneme	18.77±0.08	15.68±0.11	**13.54**±0.09	19.88±0.07	18.69±0.09	17.09±0.28
primary	**69.50**±0.55	68.29±0.64	67.82±0.86	68.95±0.42	68.26±0.70	67.59±0.99
satimage	16.50±0.07	15.52±0.05	**14.87**±0.13	15.85±0.08	15.18±0.07	14.36±0.16
segment	5.72±0.14	4.36±0.09	**3.52**±0.11	5.03±0.11	4.21±0.09	3.17±0.13
shuttle	0.06±0.00	0.04±0.00	0.03±0.00	0.05±0.00	0.04±0.00	**0.02**±0.00
sick	1.98±0.05	1.58±0.05	**1.35**±0.07	1.78±0.04	1.54±0.04	1.37±0.09
sonar	32.73±0.74	30.39±0.66	29.31±0.78	30.75±0.66	**29.27**±0.69	27.60±1.32
soybean	15.31±0.37	11.46±0.35	10.01±0.23	**13.03**±0.27	11.19±0.29	9.47±0.41
splice	6.97±0.12	6.25±0.08	6.73±0.13	6.03±0.06	**5.53**±0.05	5.34±0.07
vehicle	32.31±0.34	29.88±0.31	**28.23**±0.49	30.99±0.25	29.29±0.40	26.88±0.81
voting	6.79±0.26	6.55±0.25	6.47±0.22	**6.64**±0.17	6.55±0.23	6.45±0.37
waveform	**26.61**±0.12	26.12±0.15	25.98±0.12	25.97±0.12	25.59±0.15	25.29±0.13
G. Mean	13.43	11.68	10.63	12.53	11.50	10.09

Table I.2: Comparison of $\overline{L}$.

Dataset	B(.5;30)	B(1;30)	B(2;30)	C(2;15)	C(3;10)	C(30;1)
anneal	1.28±0.12	0.67±0.08	0.35±0.07	0.99±0.10	0.57±0.07	**0.06**±0.03
audiology	22.38±0.65	14.71±0.48	8.55±0.59	**18.61**±0.42	13.49±0.27	2.63±0.33
autos	29.45±0.92	19.75±0.65	**12.51**±0.61	24.18±0.58	17.47±0.35	5.05±0.46
balance	**18.02**±0.13	13.67±0.34	9.45±0.32	16.10±0.29	12.63±0.30	4.10±0.35
breastc	25.67±0.59	22.26±0.47	18.20±0.56	**24.49**±0.50	21.05±0.52	10.45±0.97
breastw	4.70±0.17	4.36±0.15	4.09±0.16	**4.55**±0.14	4.28±0.07	2.51±0.15
colic	11.04±0.37	9.77±0.24	8.86±0.29	10.34±0.19	**9.49**±0.29	6.66±0.45
credita	**12.89**±0.19	11.97±0.38	10.98±0.38	12.27±0.22	11.17±0.37	5.91±0.23
creditg	**24.57**±0.23	23.27±0.30	21.48±0.18	23.97±0.10	22.68±0.15	16.13±0.43
diabetes	**22.42**±0.34	20.85±0.19	19.57±0.20	23.28±0.27	22.58±0.21	19.32±0.43
glass	29.95±0.92	24.31±0.56	19.88±0.54	**27.34**±0.64	24.13±0.59	12.69±0.68
heartc	18.73±0.45	17.44±0.44	16.46±0.49	**17.39**±0.45	16.21±0.43	8.37±1.05
hearth	**15.65**±0.40	14.18±0.41	12.06±0.32	15.36±0.33	14.14±0.40	10.22±0.44
hearts	50.45±0.91	**45.85**±0.92	40.54±1.07	48.14±0.71	43.85±0.98	27.64±1.44
heartv	19.18±0.59	17.59±0.44	16.43±0.47	**17.93**±0.70	16.15±0.49	8.21±0.89
hepatitis	**15.30**±0.51	13.62±0.59	12.08±0.66	14.09±0.50	12.63±0.62	6.15±0.89
hypo	0.59±0.05	**0.35**±0.03	0.21±0.03	0.47±0.03	0.32±0.02	0.08±0.01
ionosphere	9.40±0.28	7.89±0.44	**6.95**±0.29	8.49±0.31	7.65±0.27	4.66±0.59
iris	3.99±0.48	2.84±0.54	**1.73**±0.35	3.48±0.28	2.84±0.33	0.96±0.26
krk	25.92±0.06	18.38±0.05	**12.02**±0.05	22.06±0.03	16.79±0.05	4.88±0.06
krkp	1.05±0.07	0.57±0.05	**0.24**±0.04	0.76±0.05	0.48±0.04	0.06±0.01
labor	18.54±1.36	13.25±1.38	**10.14**±1.35	16.56±1.09	13.10±1.10	5.93±0.82
letter	16.01±0.03	**12.74**±0.09	10.59±0.06	14.14±0.07	12.03±0.05	7.05±0.26
lymph	20.14±0.72	17.03±0.47	13.82±0.65	**18.54**±0.76	15.92±0.84	5.79±0.84
phoneme	14.60±0.13	11.77±0.08	**9.46**±0.10	15.36±0.08	14.08±0.10	10.87±0.20
primary	**49.94**±0.82	42.05±0.64	34.14±0.70	46.96±0.49	40.74±0.62	23.99±0.60
satimage	13.10±0.07	12.11±0.05	**11.43**±0.11	12.43±0.08	11.71±0.08	9.37±0.10
segment	4.69±0.13	3.44±0.07	**2.55**±0.10	4.09±0.08	3.25±0.09	1.51±0.10
shuttle	0.04±0.00	0.03±0.00	0.01±0.00	0.03±0.00	0.02±0.00	**0.00**±0.00
sick	1.43±0.05	1.11±0.04	**0.81**±0.03	1.29±0.03	1.03±0.04	0.31±0.02
sonar	27.11±0.83	24.77±0.49	22.97±0.77	25.43±0.59	**23.43**±0.62	12.94±0.99
soybean	12.44±0.36	7.93±0.20	5.41±0.21	**9.99**±0.22	7.45±0.23	2.47±0.20
splice	2.89±0.10	1.90±0.10	2.48±0.18	2.04±0.06	**1.34**±0.09	0.40±0.05
vehicle	25.61±0.33	22.23±0.32	**19.14**±0.43	24.27±0.39	22.14±0.22	13.01±0.53
voting	4.84±0.23	4.27±0.17	3.64±0.14	**4.66**±0.15	4.27±0.17	2.16±0.17
waveform	**21.90**±0.13	21.06±0.09	20.55±0.06	20.90±0.11	20.15±0.09	17.40±0.26
G. Mean	9.79	7.74	6.07	8.77	7.29	3.16

Table I.3: Comparison of $\overline{D}$.

Dataset	B(.5;30)	B(1;30)	B(2;30)	C(2;15)	C(3;10)	C(30;1)
anneal	1.09±0.13	0.59±0.08	0.32±0.08	0.86±0.08	0.56±0.07	**0.06**±0.03
audiology	17.21±0.57	10.51±0.68	4.82±0.68	**14.52**±0.65	9.60±0.46	1.27±0.24
autos	25.64±0.80	16.29±0.69	**9.13**±0.39	20.82±0.87	14.57±0.40	3.31±0.40
balance	**12.32**±0.45	7.20±0.40	4.45±0.38	9.58±0.41	6.36±0.30	1.48±0.20
breastc	23.36±0.55	19.87±0.62	16.26±0.76	**22.36**±0.74	19.11±0.89	8.55±0.92
breastw	3.77±0.22	3.49±0.16	3.14±0.13	**3.73**±0.18	3.43±0.18	1.67±0.15
colic	9.60±0.36	8.37±0.30	7.55±0.28	9.03±0.28	**8.36**±0.53	5.46±0.47
credita	**10.70**±0.49	9.76±0.37	8.77±0.29	10.10±0.28	9.12±0.44	4.10±0.32
creditg	**21.83**±0.27	20.51±0.45	18.62±0.33	21.08±0.16	19.71±0.30	13.22±0.41
diabetes	**19.52**±0.39	17.89±0.44	16.60±0.42	20.29±0.30	19.65±0.29	16.74±0.45
glass	25.74±0.99	20.03±0.81	15.51±0.72	**23.58**±0.68	19.74±0.59	9.27±0.79
heartc	16.02±0.43	14.55±0.67	13.33±0.46	**14.85**±0.52	13.66±0.35	6.29±0.89
hearth	**11.81**±0.58	10.02±0.67	8.99±0.31	10.88±0.91	9.90±0.68	7.38±0.87
hearts	50.08±2.16	**46.40**±1.63	39.82±2.18	48.88±1.90	44.95±1.93	28.65±2.43
heartv	16.62±0.49	14.64±0.40	13.21±0.80	**15.34**±1.00	13.37±0.74	5.97±0.93
hepatitis	**12.56**±0.80	10.59±0.92	9.55±0.86	11.01±0.53	9.49±0.91	3.81±0.82
hypo	0.51±0.04	**0.30**±0.02	0.18±0.03	0.41±0.02	0.27±0.03	0.06±0.01
ionosphere	8.05±0.19	6.50±0.40	**5.48**±0.37	7.13±0.45	6.17±0.30	3.56±0.51
iris	2.86±0.48	2.13±0.61	**1.31**±0.18	2.58±0.33	1.98±0.37	0.76±0.21
krk	21.27±0.08	14.61±0.09	**8.88**±0.07	17.78±0.06	13.17±0.08	3.06±0.04
krkp	0.84±0.06	0.47±0.06	**0.20**±0.04	0.60±0.05	0.40±0.03	0.05±0.01
labor	15.94±1.52	11.15±1.24	**8.53**±1.53	14.21±1.90	10.85±1.26	5.28±0.98
letter	13.25±0.07	**10.23**±0.12	8.05±0.09	11.50±0.11	9.41±0.07	4.13±0.21
lymph	17.01±1.02	13.61±0.84	10.66±1.34	**15.44**±0.81	12.57±0.95	3.80±0.85
phoneme	12.22±0.15	9.71±0.07	**7.63**±0.09	12.87±0.08	11.72±0.11	8.84±0.18
primary	**40.41**±1.75	33.23±1.60	27.50±1.42	37.45±1.00	32.47±0.83	20.29±1.64
satimage	10.36±0.12	9.55±0.07	**8.93**±0.05	9.78±0.10	9.13±0.11	6.73±0.14
segment	3.77±0.18	2.75±0.07	**1.98**±0.11	3.28±0.09	2.59±0.11	0.96±0.11
shuttle	0.03±0.00	0.02±0.00	0.01±0.00	0.02±0.00	0.01±0.00	**0.00**±0.00
sick	1.06±0.08	0.79±0.05	**0.55**±0.03	0.93±0.05	0.72±0.07	0.19±0.02
sonar	24.22±0.98	21.84±0.91	19.65±1.02	22.48±0.83	**20.43**±0.88	9.94±0.73
soybean	10.25±0.29	5.86±0.21	3.59±0.24	**7.91**±0.30	5.28±0.24	1.72±0.11
splice	1.89±0.12	1.15±0.09	1.89±0.18	1.40±0.10	**0.83**±0.09	0.19±0.05
vehicle	19.84±0.58	16.70±0.47	**13.97**±0.50	18.32±0.56	16.29±0.27	8.99±0.58
voting	4.11±0.23	3.43±0.19	2.82±0.11	**4.02**±0.15	3.39±0.18	1.50±0.11
waveform	**19.05**±0.15	18.05±0.12	17.48±0.13	17.96±0.13	17.13±0.11	14.16±0.30
G. Mean	8.03	6.17	4.70	7.14	5.77	2.24

Table I.4: Comparison of $\overline{D}_T$.

Dataset	B(.5;30)	B(1;30)	B(2;30)	C(2;15)	C(3;10)	C(30;1)
anneal	13.23±3.74	5.88±4.01	2.27±2.10	10.55±4.31	1.00±1.76	**0.10**±0.32
audiology	43.49±2.99	32.92±3.01	24.01±2.89	**37.93**±3.84	30.45±3.13	8.11±1.40
autos	44.27±1.97	36.67±2.73	**29.15**±3.39	39.34±2.49	31.55±2.82	13.66±2.06
balance	**47.74**±0.80	39.57±1.12	28.18±1.26	45.47±0.71	37.66±0.73	13.33±1.19
breastc	30.96±1.16	27.48±0.70	22.30±1.05	**29.43**±0.85	25.22±1.34	13.88±1.85
breastw	23.93±3.74	23.39±3.94	24.26±3.93	**22.85**±4.53	22.85±2.59	17.15±2.28
colic	19.07±1.81	17.61±0.69	16.37±1.18	17.67±1.43	**15.69**±1.37	12.73±1.71
credita	**26.17**±1.38	25.07±1.24	22.91±1.34	25.12±1.07	22.79±1.33	13.81±0.96
creditg	**32.64**±0.43	31.33±0.64	29.48±0.32	32.10±0.35	30.70±0.40	22.74±1.20
diabetes	**31.68**±0.67	30.04±0.87	28.51±0.61	32.61±0.49	31.84±0.30	27.14±0.80
glass	41.58±1.87	36.71±1.43	32.28±0.75	**37.86**±2.04	36.48±2.14	20.23±1.40
heartc	30.83±1.28	29.33±1.74	27.65±1.60	**28.36**±2.07	26.96±1.27	14.91±2.01
hearth	**30.90**±0.99	28.89±1.50	23.17±1.49	31.65±1.66	28.84±1.34	19.70±2.10
hearts	50.33±0.87	**45.71**±1.17	41.28±1.43	47.51±1.03	42.98±1.09	27.21±1.68
heartv	30.63±1.18	28.70±1.98	27.33±2.13	**29.17**±2.17	27.41±1.15	14.57±1.94
hepatitis	**25.88**±4.54	26.81±2.83	23.14±2.55	26.42±2.65	24.67±2.83	14.73±3.79
hypo	16.20±5.43	**11.76**±6.10	7.90±1.90	14.88±4.96	11.93±3.54	3.85±2.64
ionosphere	23.72±3.10	22.92±3.13	**22.71**±2.41	23.46±2.55	23.59±2.93	14.98±3.68
iris	13.52±2.64	9.07±4.21	**5.59**±3.47	10.63±3.26	10.33±3.30	1.96±1.12
krk	45.05±0.09	37.93±0.15	**29.06**±0.18	41.98±0.10	35.67±0.20	13.46±0.27
krkp	27.70±6.09	17.15±3.51	**7.48**±3.73	23.16±5.34	14.57±3.92	2.36±1.20
labor	18.29±3.54	14.95±3.47	**11.09**±1.83	17.85±5.20	16.34±3.11	5.54±1.52
letter	57.88±0.24	**54.00**±0.40	50.05±0.32	55.74±0.28	52.46±0.24	35.40±1.35
lymph	32.31±2.09	30.95±3.30	26.14±3.48	**32.57**±2.20	29.65±2.39	12.32±3.40
phoneme	32.38±0.40	30.38±0.43	**27.68**±0.71	32.32±0.23	31.11±0.20	25.13±0.50
primary	**56.67**±1.07	47.85±0.76	38.28±1.23	53.45±0.67	46.11±1.02	26.24±1.33
satimage	38.86±0.41	37.47±0.44	**36.51**±0.60	38.00±0.41	36.97±0.48	31.29±0.68
segment	37.33±1.30	33.94±1.95	**30.69**±2.64	36.26±1.90	32.51±2.62	21.87±2.77
shuttle	29.73±2.29	15.97±3.54	13.20±3.12	22.26±2.59	15.42±1.68	**4.11**±2.51
sick	28.85±1.53	28.04±2.27	**22.64**±2.90	29.39±1.82	27.57±2.83	9.53±1.50
sonar	36.81±1.67	36.21±2.11	34.84±2.28	37.51±1.26	**34.64**±2.25	22.08±3.27
soybean	38.24±0.98	30.79±2.26	25.20±3.01	**34.88**±2.43	30.80±2.35	10.81±2.64
splice	18.16±1.07	13.90±0.90	12.45±0.53	13.19±1.07	**10.62**±0.54	4.00±0.53
vehicle	42.54±0.70	39.10±1.10	**35.14**±1.00	41.64±1.06	39.80±0.80	24.70±1.36
voting	17.27±3.93	19.53±2.98	16.85±3.57	**17.06**±4.24	19.10±4.18	11.34±2.68
waveform	**36.21**±0.26	35.64±0.32	34.96±0.27	35.39±0.28	34.63±0.18	30.51±0.63
G. Mean	30.44	26.20	21.57	28.44	24.02	11.53

Table I.5: Comparison of $\overline{D}_F$.

Dataset	B(.5;30)	B(1;30)	B(2;30)	C(2;15)	C(3;10)	C(30;1)
anneal	12.63±3.48	5.88±4.01	2.27±2.10	10.38±4.50	1.00±1.76	**0.10**±0.32
audiology	14.79±1.86	11.55±2.68	10.36±1.88	**11.79**±1.74	11.47±2.00	3.28±0.63
autos	20.79±1.95	17.57±2.02	**14.05**±2.03	19.00±2.69	15.41±1.88	8.11±1.67
balance	**20.33**±1.05	17.94±0.79	11.85±1.20	20.15±0.72	16.80±0.72	5.15±1.01
breastc	30.96±1.16	27.48±0.70	22.30±1.05	**29.43**±0.85	25.22±1.34	13.88±1.85
breastw	23.93±3.74	23.39±3.94	24.26±3.93	**22.85**±4.53	22.85±2.59	17.15±2.28
colic	19.07±1.81	17.61±0.69	16.37±1.18	17.67±1.43	**15.69**±1.37	12.73±1.71
credita	**26.17**±1.38	25.07±1.24	22.91±1.34	25.12±1.07	22.79±1.33	13.81±0.96
creditg	**32.64**±0.43	31.33±0.64	29.48±0.32	32.10±0.35	30.70±0.40	22.74±1.20
diabetes	**31.68**±0.67	30.04±0.87	28.51±0.61	32.61±0.49	31.84±0.30	27.14±0.80
glass	22.41±1.76	20.89±1.37	19.59±1.12	**21.11**±1.48	21.35±1.77	14.39±1.30
heartc	30.83±1.28	29.33±1.74	27.65±1.60	**28.36**±2.07	26.96±1.27	14.91±2.01
hearth	**30.90**±0.99	28.89±1.50	23.17±1.49	31.65±1.66	28.84±1.34	19.70±2.10
hearts	19.55±1.11	**18.05**±1.08	16.16±0.86	18.39±0.85	16.69±0.83	10.97±1.20
heartv	30.63±1.18	28.70±1.98	27.33±2.13	**29.17**±2.17	27.41±1.15	14.57±1.94
hepatitis	**25.88**±4.54	26.81±2.83	23.14±2.55	26.42±2.65	24.67±2.83	14.73±3.79
hypo	14.01±5.44	**11.22**±5.82	7.86±1.86	13.90±5.01	11.41±3.61	3.85±2.64
ionosphere	23.72±3.10	22.92±3.13	**22.71**±2.41	23.46±2.55	23.59±2.93	14.98±3.68
iris	13.45±2.56	9.04±4.24	**5.59**±3.47	10.53±3.24	10.27±3.32	1.96±1.12
krk	24.72±0.12	21.74±0.19	**17.46**±0.19	23.83±0.14	20.69±0.14	8.95±0.19
krkp	27.70±6.09	17.15±3.51	**7.48**±3.73	23.16±5.34	14.57±3.92	2.36±1.20
labor	18.29±3.54	14.95±3.47	**11.09**±1.83	17.85±5.20	16.34±3.11	5.54±1.52
letter	19.42±0.30	**19.40**±0.11	19.15±0.24	19.25±0.22	19.16±0.23	15.88±0.56
lymph	29.01±1.58	28.05±2.81	24.52±3.35	**29.25**±2.54	27.11±2.40	11.73±3.20
phoneme	32.38±0.40	30.38±0.43	**27.68**±0.71	32.32±0.23	31.11±0.20	25.13±0.50
primary	**9.98**±0.54	9.03±0.53	7.47±0.52	9.79±0.54	8.82±0.53	5.51±0.43
satimage	25.74±0.31	25.48±0.39	**25.24**±0.50	25.70±0.41	25.62±0.47	22.36±0.53
segment	26.92±1.34	23.84±1.67	**21.17**±2.20	26.19±1.73	22.66±2.43	16.45±1.99
shuttle	25.40±2.56	14.43±3.29	12.43±2.47	19.36±3.22	14.26±1.82	**4.03**±2.54
sick	28.85±1.53	28.04±2.27	**22.64**±2.90	29.39±1.82	27.57±2.83	9.53±1.50
sonar	36.81±1.67	36.21±2.11	34.84±2.28	37.51±1.26	**34.64**±2.25	22.08±3.27
soybean	27.20±1.36	24.50±2.05	20.51±2.48	**26.23**±1.97	24.98±2.22	8.71±2.18
splice	15.53±1.34	11.84±1.01	10.75±0.67	10.86±0.99	**8.66**±0.60	3.51±0.56
vehicle	31.26±0.63	29.92±0.76	**27.45**±0.83	32.02±0.61	31.80±0.82	21.29±1.12
voting	17.27±3.93	19.53±2.98	16.85±3.57	**17.06**±4.24	19.10±4.18	11.34±2.68
waveform	**35.29**±0.24	34.80±0.32	34.13±0.29	34.55±0.27	33.81±0.19	29.75±0.65
G. Mean	23.24	20.41	17.05	21.91	18.82	9.33

Table I.6: Comparison of $\overline{D}_P$.

J. Sanity Check for Experimental Results

Throughout this thesis, several analysis methods produce empirical estimates of the expected performance of classifiers by repeatedly measuring their loss on some data subsamples. To ensure that the reported experimental results are indeed correct and consistent with each other, we compare here the loss estimates from the various experiments with each other.

Figure J.1 on pages 165–168 shows the loss estimates for $\mathit{Bagging}(0.5; 30)$, $\mathit{Bagging}(1; 30)$, and $\mathit{Bagging}(2; 30)$ obtained from the various experiments, while Figure J.2 on pages 168–172 shows the loss estimates for $\mathit{Cragging}(2; 15)$, $\mathit{Cragging}(3; 10)$, and $\mathit{Cragging}(30; 1)$.

All the experimental results are consistent with each other as far as the relative ordering of the learning methods according to their performance is concerned.

The loss estimates from the bias-variance decomposition experiments (BVD) are consistently biased upwards, compared to the estimates obtained from the other experiments. This is a direct result of the experimental methodology: in order to estimate bias and variance, a further bootstrap sampling procedure is embedded within the 10x10 fold cross-validation. This results both in an upwards bias as well as a higher variance for the loss estimates resulting from these experiments.

The results from the other experiments (Error Curves, Margins, Decomposition) are consistent with each other as they lead to the same performance estimates.

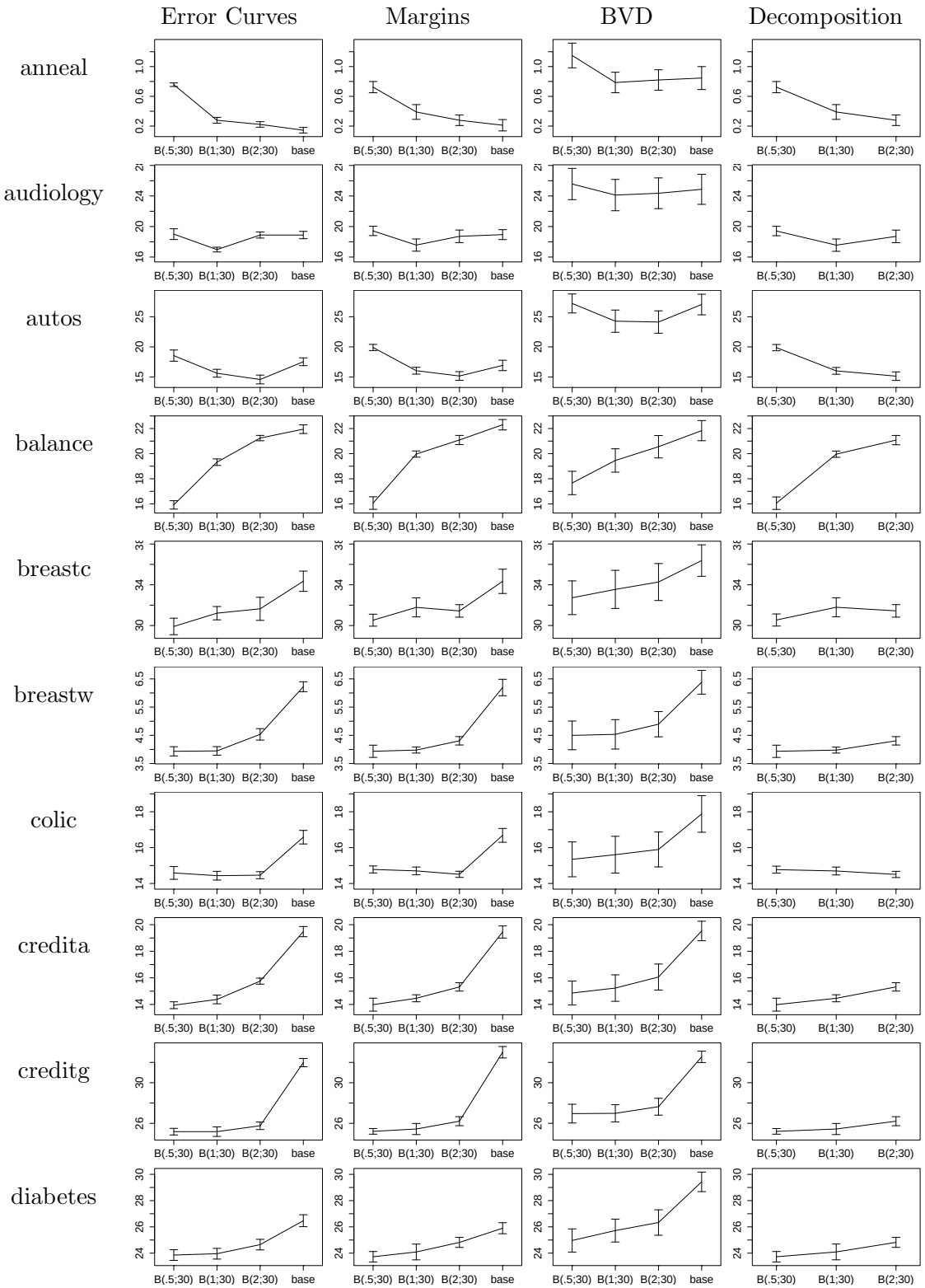

Figure J.1: Loss comparison for *Bagging*$(0.5;30)$, *Bagging*$(1;30)$, *Bagging*$(2;30)$, and the base classifier. *(continued on next page)*

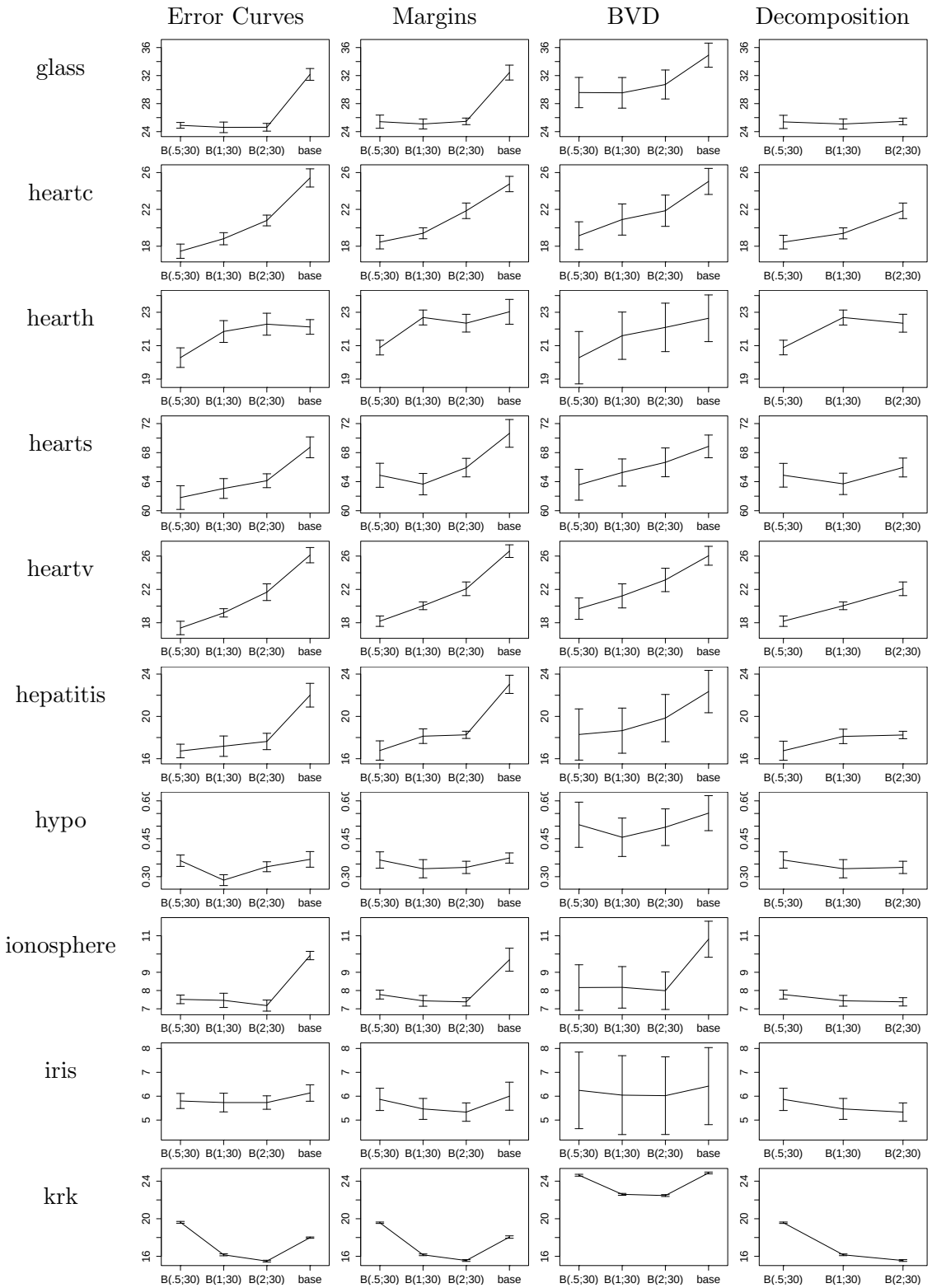

Figure J.1: Loss comparison for *Bagging*$(0.5; 30)$, *Bagging*$(1; 30)$, *Bagging*$(2; 30)$, and the base classifier. *(continued on next page)*

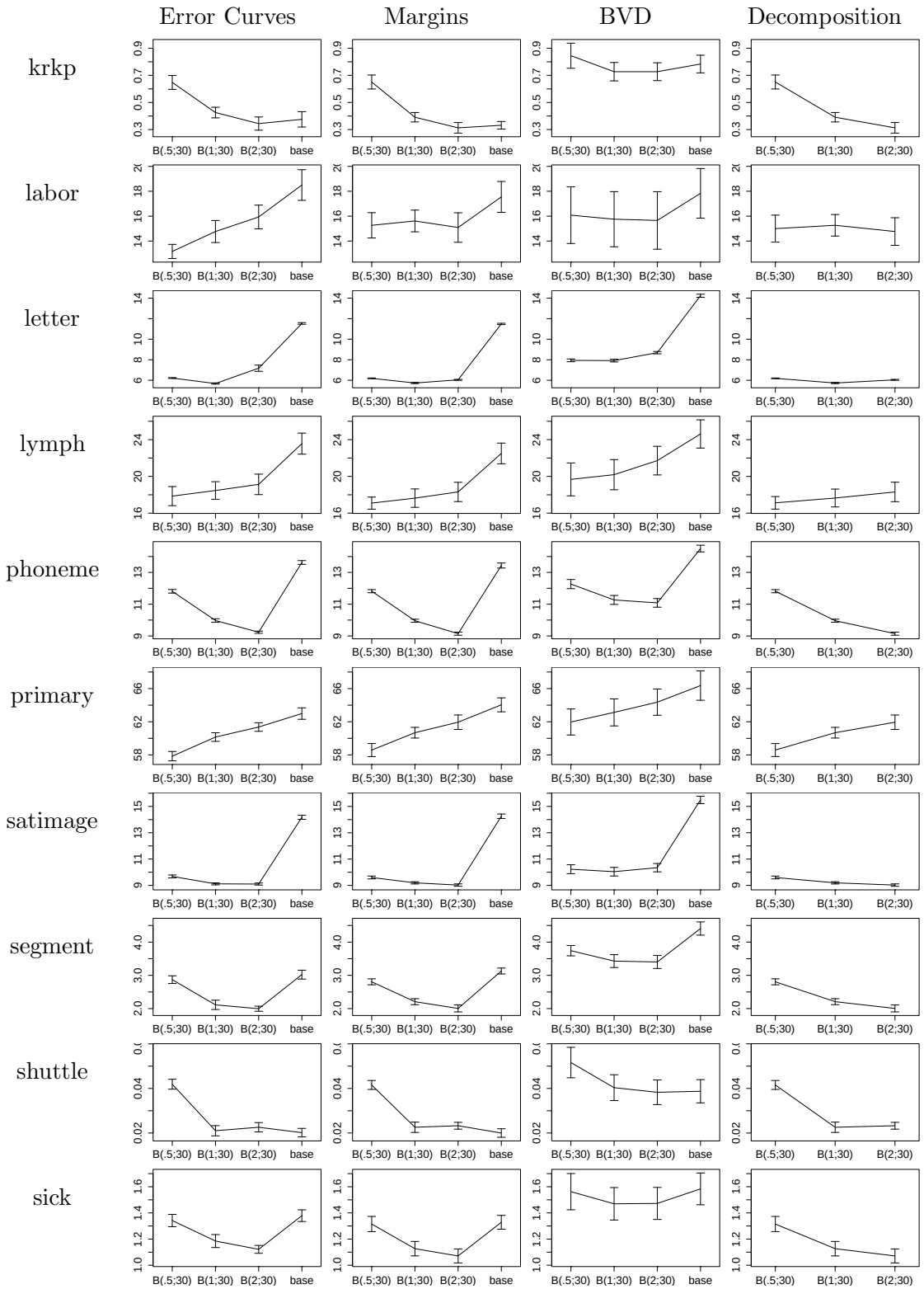

Figure J.1: Loss comparison for *Bagging*$(0.5; 30)$, *Bagging*$(1; 30)$, *Bagging*$(2; 30)$, and the base classifier. *(continued on next page)*

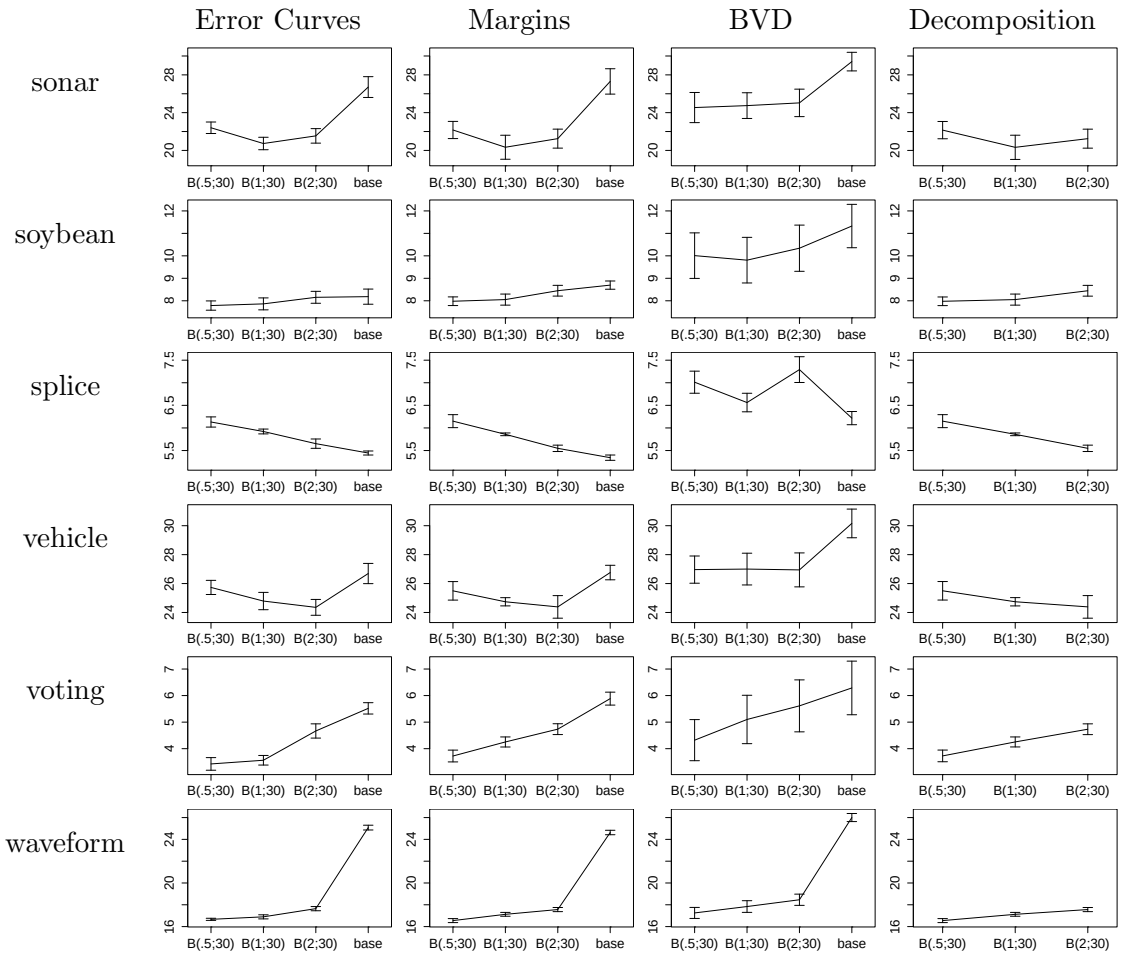

Figure J.1: Loss comparison for *Bagging*$(0.5; 30)$, *Bagging*$(1; 30)$, *Bagging*$(2; 30)$, and the base classifier.

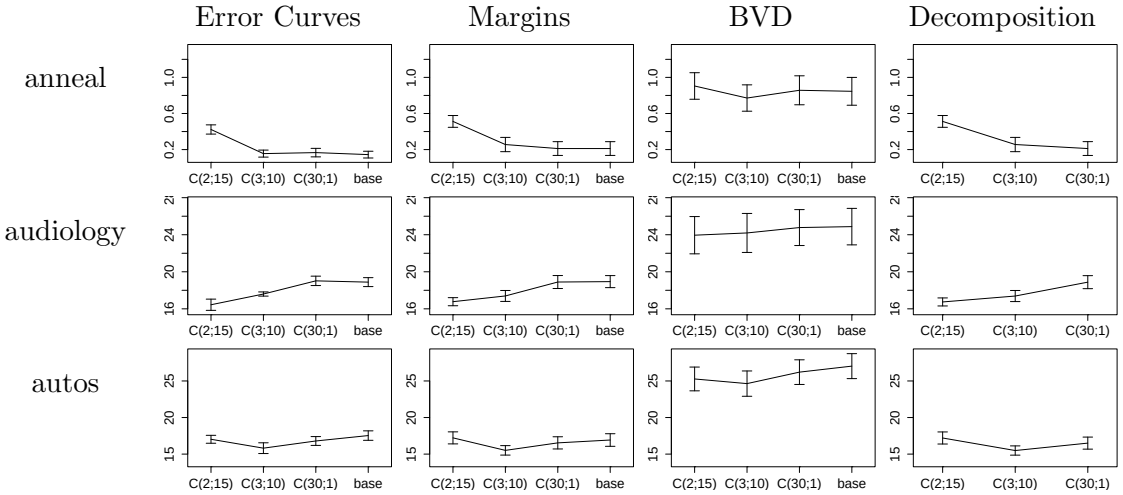

Figure J.2: Loss comparison for *Cragging*$(2; 15)$, *Cragging*$(3; 10)$, *Cragging*$(30; 1)$, and the base classifier. *(continued on next page)*

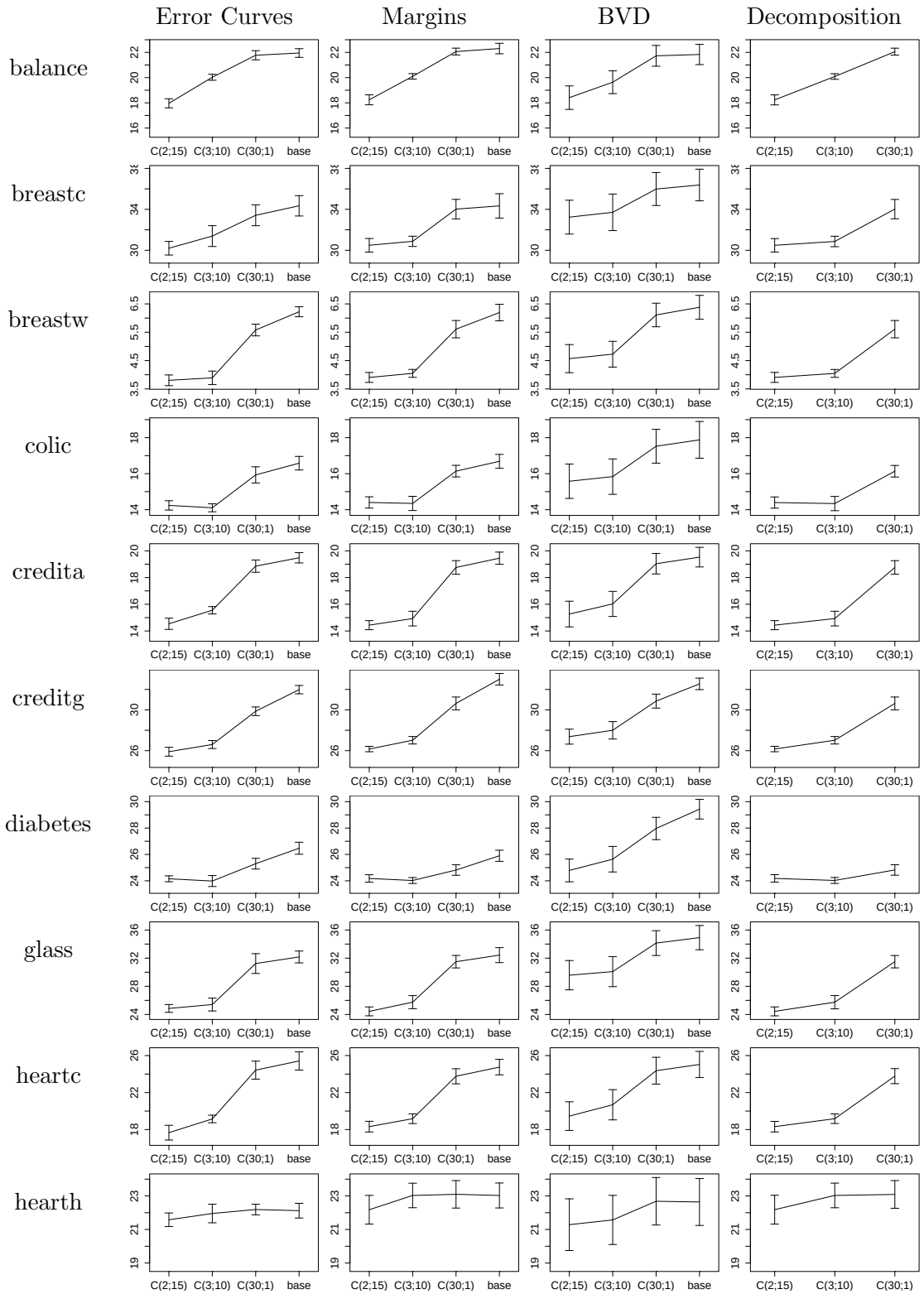

Figure J.2: Loss comparison for *Cragging*$(2;15)$, *Cragging*$(3;10)$, *Cragging*$(30;1)$, and the base classifier. *(continued on next page)*

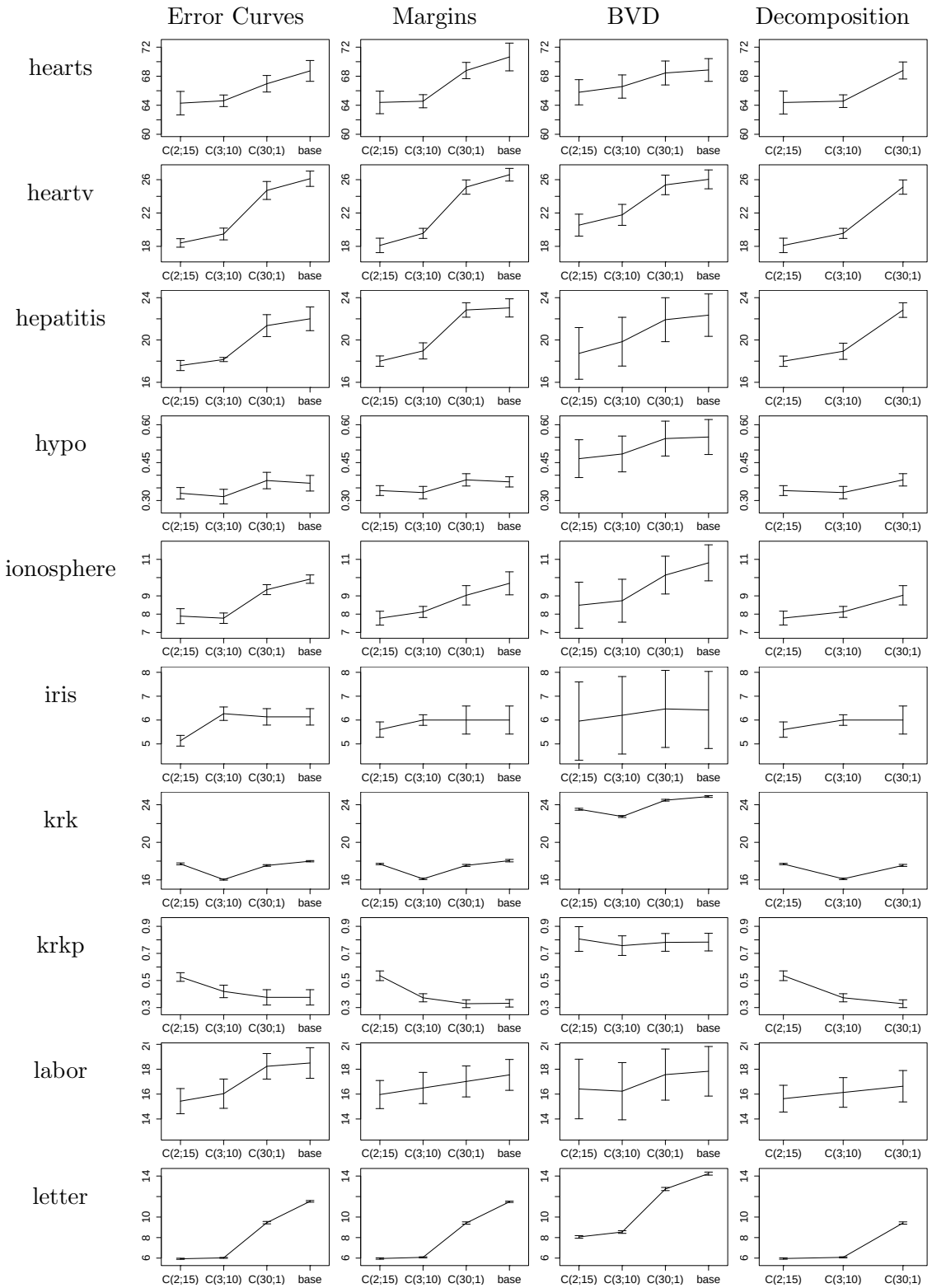

Figure J.2: Loss comparison for *Cragging*$(2;15)$, *Cragging*$(3;10)$, *Cragging*$(30;1)$, and the base classifier. *(continued on next page)*

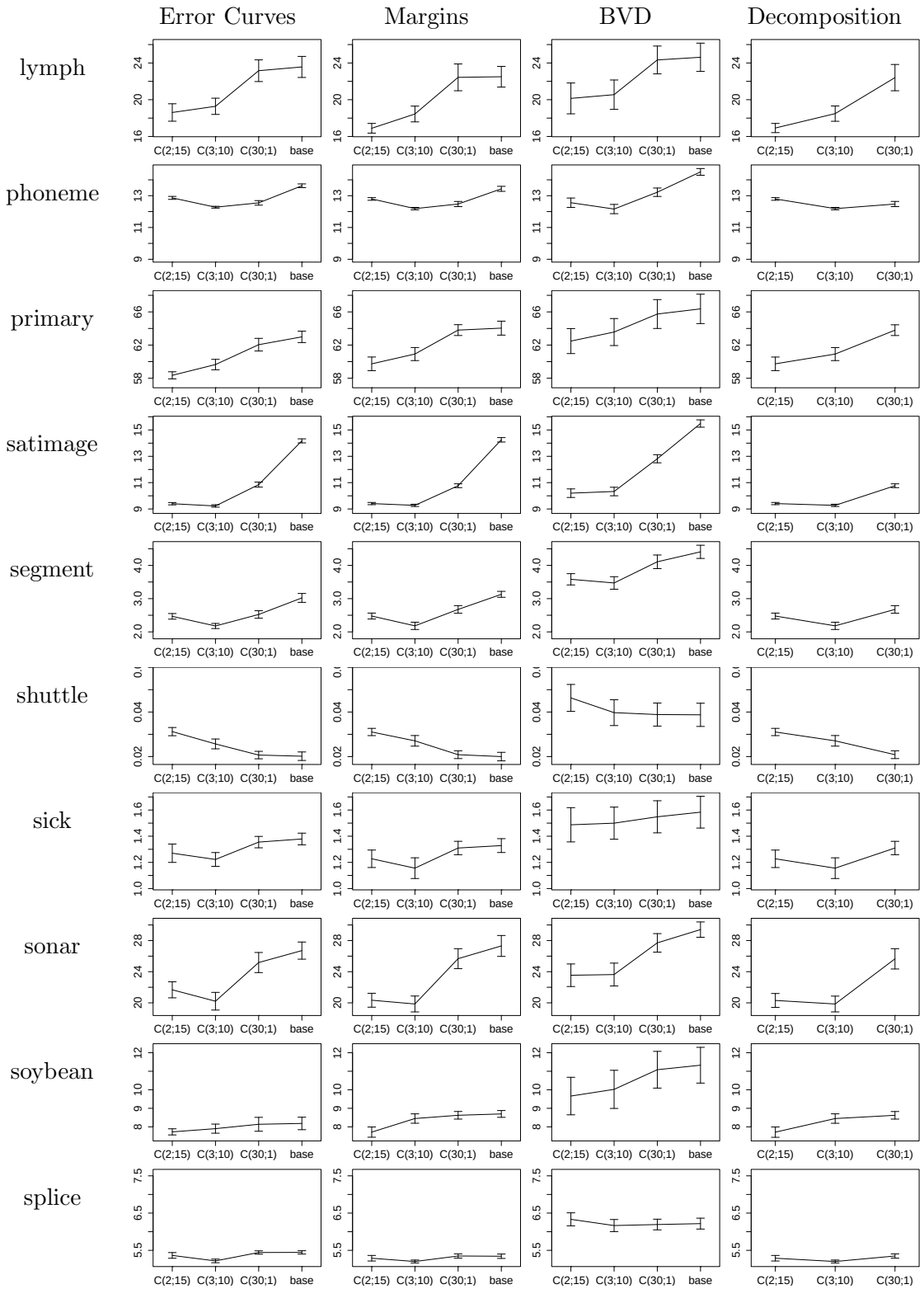

Figure J.2: Loss comparison for *Cragging*$(2;15)$, *Cragging*$(3;10)$, *Cragging*$(30;1)$, and the base classifier. *(continued on next page)*

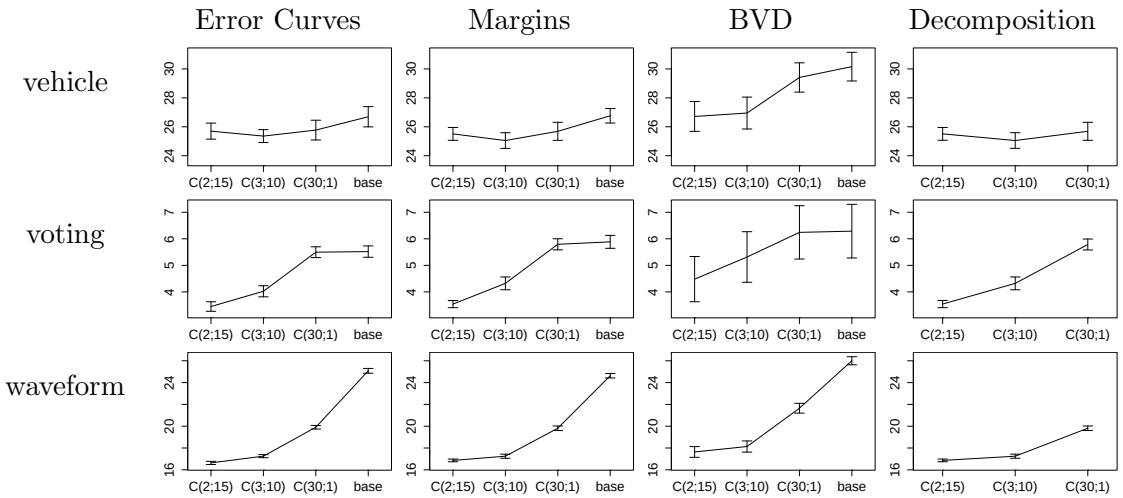

Figure J.2: Loss comparison for *Cragging*$(2;15)$, *Cragging*$(3;10)$, *Cragging*$(30;1)$, and the base classifier.

Bibliography

[1] Yali Amit, Gilles Blanchard, and Kenneth Wilder. Multiple randomized classifiers: Mrcl. Technical Report 496, Department of Statistics, University of Chicago, 2000.

[2] C. Andrieu, J. de Freitas, and A. Doucet. Sequential bayesian estimation and model selection applied to neural networks. Technical Report CUED/F-INFENG/TR 341, Cambridge University, 1999.

[3] Eric Bauer and Ron Kohavi. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine Learning*, 36(1–2):105–139, 1999.

[4] Stephen D. Bay and Michael J. Pazzani. Characterizing model errors and differences. In *Proc. 17th International Conf. on Machine Learning*, pages 49–56. Morgan Kaufmann, San Francisco, CA, 2000.

[5] C.L. Blake and C.J. Merz. UCI repository of machine learning databases, 1998.

[6] Remco R. Bouckaert, Michael Goebel, and Pat Riddle. Generalized unified decomposition of ensemble loss. Technical Report AI-04-01, Computer Science Department, University of Auckland, Auckland, New Zealand, 2004.

[7] R.R. Bouckaert and E. Frank. Choosing learning algorithms based on repeated cross-validation. Technical report, Computer Science Department, University of Waikato, Hamilton, New Zealand, 2002.

[8] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[9] Leo Breiman. Bias, variance and arcing classifiers. Technical Report 460, Department of Statistics, University of California at Berkeley, Berkeley, CA, 1996.

[10] Leo Breiman. Arcing the edge. Technical Report 486, Department of Statistics, University of California at Berkeley, 1997.

[11] Peter Buehlmann and Bin Yu. Analyzing bagging. *The Annals of Statistics*, 30(4):927–961, 2002.

[12] Andreas Buja and Werner Stuetzle. The effect of bagging on variance, bias, and mean squared error. *submitted (available at http://ljsavage.wharton.upenn.edu/~buja/)*, 2000.

[13] Andreas Buja and Werner Stuetzle. Bagging does not always decrease mean squared error. *submitted (available at http://ljsavage.wharton.upenn.edu/~buja/)*, 2001.

[14] J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

[15] William W. Cohen, Robert E. Schapire, and Yoram Singer. Learning to order things. In Michael I. Jordan, Michael J. Kearns, and Sara A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10. The MIT Press, 1998.

[16] Thomas G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 26:1–22, 1998.

[17] Thomas G. Dietterich. Machine-learning research: Four current directions. *The AI Magazine*, 18(4):97–136, 1998.

[18] Thomas G. Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems, First International Workshop, MCS 2000, June 2000*, Cagliari, Italy, 2000.

[19] Thomas G. Dietterich and Ghulum Bakiri. Error-correcting output codes: A general method for improving multiclass inductive learning programs. In *Proceedings of AAAI-91*, 1991.

[20] Pedro Domingos. Why does bagging work? A bayesian account and its implications. In *Knowledge Discovery and Data Mining*, pages 155–158, 1997.

[21] Pedro Domingos. How to get a free lunch: A simple cost model for machine learning applications. In *Proceedings of the AAAI-98/ICML-98 Workshop on the Methodology of Applying Machine Learning*, pages 1–7. AAAI Press, Madison, WI, 1998.

[22] Pedro Domingos. Metacost: A general method for making classifiers cost-sensitive. In *KDD99*, 1999.

[23] Pedro Domingos. Bayesian averaging of classifiers and the overfitting problem. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 223–230, Stanford, CA, 2000. Morgan Kaufmann.

[24] Pedro Domingos. A unified bias-variance decomposition. Technical report, Department of Computer Science and Engineering, University of Washington, Seattle, WA, 2000.

[25] Pedro Domingos. A unified bias-variance decomposition and its applications. In *Proc. 17th International Conf. on Machine Learning*, pages 231–238. Morgan Kaufmann, San Francisco, CA, 2000.

[26] Pedro Domingos. A unified bias-variance decomposition for zero-one and squared loss. In *AAAI/IAAI*, pages 564–569, 2000.

[27] R. Duin, D. Tax, and J. Kittler. Combining multiple classifiers by averaging or by multiplying? *Pattern Recognition*, 33:1475–1485, 2000.

[28] Yoav Freund, Raj Iyer, Robert E. Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. In Jude W. Shavlik, editor, *Proceedings of ICML-98, 15th International Conference on Machine Learning*, pages 170–178, Madison, US, 1998. Morgan Kaufmann Publishers, San Francisco, US.

[29] Yoav Freund and Robert E. Schapire. Experiments with a new boosting algorithm. In *Proc. 13th International Conference on Machine Learning*, pages 148–146. Morgan Kaufmann, 1996.

[30] J. Friedman and P. Hall. On bagging and nonlinear estimation. Technical report, Statistics Department, Stanford University, 2000.

[31] J.H. Friedman. On bias, variance, 0/1 - loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, 1:55–77, 1997.

[32] Nir Friedman and Daphne Koller. Being bayesian about network structure. In C. Boutilier and M. Godszmidt, editors, *Uncertainty in Articial Intelligence*, pages 201–210. Morgan Kaufmann, 2000.

[33] S. Geman, E. Bienenstock, and R. Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4:1–58, 1992.

[34] Y. Grandvalet. Bagging down-weights leverage points. In *Proc. of IJCNN'2000*, volume 4, pages 505–510. IEEE, 2000.

[35] Y. Grandvalet. Bagging can stabilize without reducing variance. In *Proc. of ICANN'01*, Lecture Notes in Computer Science, pages 49–56. Springer, 2001.

[36] Y. Grandvalet. Bagging equalizes influence. *Machine Learning, to appear (available from http://www.iro.umontreal.ca/˜grandvay/)*, 2003.

[37] Adam J. Grove and Dale Schuurmans. Boosting in the limit: Maximizing the margin of learned ensembles. In *Proceedings of AAAI-98*, pages 692–699, Madison, WI, 1998. AAAI Press.

[38] J. Hansen and T.M. Heskes. General bias/variance decomposition with target independent variance of error functions derived from the exponential family of distributions. In A. Sanfeliu, J.J. Villanueva, M. Vanrell, R. Alguezar, A.K. Jain, and J. Kittler, editors, *15th International Conference on Pattern Recognition*, pages 207–210, 2000.

[39] L. Hansen and P. Salamon. Neural network ensembles. *IEEE Trans. Pattern analysis and Machine Intelligence*, 12:993–1003, 1990.

[40] Michael Harries. Boosting a strong learner: evidence against the minimum margin. In *Proc. 16th International Conference on Machine Learning*, pages 171–180. Morgan Kaufmann, San Francisco, CA, 1999.

[41] R. Herbrich and T. Graepel. A PAC-bayesian margin bound for linear classifiers. *IEEE Transactions on Information Theory*, 48(12):3140–3150, 2002.

[42] Tin Kam Ho. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998.

[43] Tin Kam Ho. Data complexity analysis for classifier combinaton. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems 2001*, volume 2096 of *LNCS*, pages 53–67, Berlin, 2001. Springer-Verlag.

[44] Michael I. Jordan and Robert A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6:181–214, 1994.

[45] Kukjin Kang and Jong-Hoon Oh. Statistical mechanics of the mixture of experts. In Michael C. Mozer, Michael I. Jordan, and Thomas Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, page 183. The MIT Press, 1997.

[46] C. Kaynak and E. Alpaydin. Multistage cascading of multiple classifiers: One man's noise is another man's data. In *ICML 2000*, Stanford University, USA, 2000.

[47] Jyrki Kivinen and Manfred K. Warmuth. Averaging expert predictions. *Lecture Notes in Computer Science*, 1572:153–167, 1999.

[48] R. Kohavi and D.H. Wolpert. Bias plus variance decomposition for zero-one loss functions. In *Proceedings of the 13th International Conference on Machine Learning*, pages 275–283, Bari, Italy, 1996. Morgan Kaufmann.

[49] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1137–1145, 1995.

[50] Eun Bae Kong and Thomas G. Dietterich. Error-correcting output coding corrects bias and variance. In *Proc. 12th International Conference on Machine Learning*, pages 313–321. Morgan Kaufmann, 1995.

[51] Eun Bae Kong and Thomas G. Dietterich. Machine learning bias, statistical bias, and statistical variance of decision tree algorithms. Technical report, Department of Computer Science, Oregon State University, Corvallis, Oregon, 1995.

[52] A. Krogh and J. Vedelsby. Neural network ensembles, cross validation, and active learning. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 231–238, Cambridge, CA, 1995. MIT Press.

[53] L.I. Kuncheva et al. Is independence good for combining classifiers? In *Proceedings of ICPR2000, 15th Int. Conference on Pattern Recognition*, volume 2, pages 168–171, Barcelona, Spain, 2000.

[54] Patrce Latinne, Olivier Debeir, and Christine Decaestecker. Different ways of weakening deciosion trees and their impact on classification accuracy of dt combination. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems 2000*, volume 1857 of *LNCS*, pages 200–209, Berlin, 2000. Springer-Verlag.

[55] S. Madhvanath and V. Govindaraju. Serial classifier combination for handwritten word recognition. In *Proceedings of the Third International Conference on Document Analysis and Recognition*, pages 911–914, Montreal, Canada, August 1995.

[56] Dragos D. Margineantu and Thomas G. Dietterich. Pruning adaptive boosting. In *Proc. 14th International Conference on Machine Learning*, pages 211–218. Morgan Kaufmann, 1997.

[57] Thomas P. Minka. Bayesian model averaging is not model combination. Technical Report (MIT Media Lab note 7/6/00), 2000.

[58] Tim Oates and David Jensen. The effects of training set size on decision tree complexity. In *Proceedings of The Fourteenth International Conference on Machine Learning (ICML97)*, pages 254–262, 1997.

[59] David Opitz and Richard Maclin. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198, August 1999.

[60] David Opitz and Jude Shavlik. Actively searching for an effective neural-network ensemble. *Connection Science*, 8 (3–4), 1996.

[61] B. Parmanto, P. W. Munro, and H. R. Doyle. Improving committee diagnosis with resampling techniques. In D. S. Touretzky, M. C. Mozer, and M. E. Hesselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 882–888, Cambridge, MA, 1996. MIT Press.

[62] David M. Pennock, Eric Horvitz, and C. Lee Giles. Social choice theory and recommender systems: Analysis of the axiomatic foundations of collaborative filtering. In *AAAI/IAAI*, pages 729–734, 2000.

[63] David M. Pennock, Pedrito Maynard-Reid II, C. Lee Giles, and Eric Horvitz. A normative examination of ensemble learning algorithms. In *Proc. 17th International Conf. on Machine Learning*, pages 735–742. Morgan Kaufmann, San Francisco, CA, 2000.

[64] Michaelis Petrakos, Jon Atli Benediktsson, and Ioannis Kanellopoulos. The effect of classifier agreement on the accuracy of the combined classifier in decision level fusion. In *IEEE Transactions on Geoscience and Remote Sensing*, volume 39, pages 2539–2546, November 2001.

[65] Bernhard Pfahringer and Ian H. Witten. Improving bagging performance by increasing decision tree diversity. Technical Report TR-97-31, Oesterreichisches Forschungsinstitut fuer Artificial Intelligence, Wien, 1997.

[66] J.R. Quinlan. Bagging, boosting, and c4.5. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 725–730, Cambridge, MA, 1996. AAAI Press/MIT Press.

[67] J.R. Quinlan. Miniboosting decision trees. *Submitted to Journal of Artificial Intelligence Research (available at http://www.cse.unsw.edu.au/~quinlan/miniboost.ps)*, 1999.

[68] Fabio Roli, Olivier Debeir, and Christine Decaestecker. Limiting the number of trees in random forests. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems 2001*, volume 2096 of *LNCS*, pages 178–187, Berlin, 2001. Springer-Verlag.

[69] Fabio Roli, Giorgio Giacinto, and Gianni Vernazza. Methods for designing multiple classifier systems. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems 2001*, volume 2096 of *LNCS*, pages 78–87, Berlin, 2001. Springer-Verlag.

[70] Dymitr Ruta and Bogdan Gabrys. A theoretical analysis of the limits of majority votig errors for multiple classifier systems. *Pattern Analysis and Applications*, 2002(5):333–350, 2002.

[71] Robert E. Schapire, Yoav Freund, Peter Bartlett, and Wee Sun Lee. Boosting the margin: a new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26(5):1651–1686, 1998.

[72] Robert E. Schapire and Yoram Singer. Improved boosting algorithms using confidence-rated predictions. In *Computational Learing Theory*, pages 80–91, 1998.

[73] Ching Y. Suen and Louisa Lam. Multiple classifier combination methodologies for different output levels. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems 2000*, volume 1857 of *LNCS*, pages 52–66, Berlin, 2001. Springer-Verlag.

[74] D. Tax, R. Duin, and M. van Breukelen. Comparison between product and mean classifier combination rules. In *Proceedings of the Workshop on Statistical Pattern Recognition*, 1997.

[75] R. Tibshirani. Bias, variance and prediction error for classification rules. Technical report, Department of Preventive Medicine and Biostatistics and Department of Statistics, University of Toronto, Toronto, Canada, 1996.

[76] Kagan Tumer and Joydeep Ghosh. Error correlation and error reduction in ensemble classifiers. *Connection Science*, 8(3-4):385–403, 1996.

[77] Ian H. Witten and Eibe Frank. *Data mining: Practical machine learning tools and techniques with Java implementaions.* Morgan Kaufmann, San Francisco, 1999.

[78] David H. Wolpert and William G. Macready. No free lunch theorems for search. Technical Report SFI-TR-95-02-010, Santa Fe Institute, Santa Fe, NM, 1995.

[79] David H. Wolpert and William G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, April 1997.

[80] D.H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.

[81] Gabriele Zenobi and Pádraig Cunningham. Using diversity in preparing ensembles of classifiers based on different feature subsets to minimize generalization error. In L. De Readt and P. Flach, editors, *12th European Conference on Machine Learning (ECML 2001)*, volume 2167 of *LNAI*, pages 576–587. Springer Verlag, 2001.