Model Correlation in Random Forests

Master Thesis

presented by

Benjamin Moser

at the

University of Konstanz

Department of Computer and Information Science

1st Supervisor: Sven Kosub

2nd Supervisor: Tobias Sutter

Konstanz, 2023

Abstract

A Random Forest is an ensemble of randomised decision trees. How are differences between individual trees related to the performance of the forest? Despite their simplicity and success, it is not yet fully clear when and why Random Forests work well. We approach this problem from the general perspective of ensemble learning. Guided by the diversity decomposition of the ensemble error, we analyse the role of diversity in regression and classification ensembles and argue that this theory is particularly relevant to Random Forest ensembles. We provide a thorough introduction to the diversity theory and relate it to previous results on ensemble learning. We further link it theoretically to the recently developed notion of ensemble competence. Focusing on 0/1-classification, we explore methods to regulate diversity in Random Forests. We see that it is possible to obtain smaller and better Random Forest ensembles. We further propose a generalisation of a well-known diversity regularisation scheme for neural network ensembles.

About this version

This document might contain (slightly) more than what I handed in for the thesis examination. When (or if) I continue thinking about this topic, I will add notes (more likely to-dos) to this document.

As in any project, time is very limited. There might be minor errors here and there. Let me know if you spot anything.

Contents

Contents iii

1. Introduction 1
1.1. Overview . . . 1
1.2. Contributions . . . 2
1.3. Statistics . . . 3
1.4. Supervised Learning . . . 5
1.5. Bias, Variance and their Effects . . . 6
1.6. Bregman Divergences and Centroids . . . 8

2. Ensemble Learning 11
2.1. Methods . . . 11
2.2. Notation . . . 11
2.3. Motivation . . . 13

3. Random Forests 17
3.1. Decision Trees . . . 17
3.2. The Random Forest scheme . . . 20

4. Diversity 23
4.1. Measures of Diversity . . . 23
4.2. The Diversity-Effect Decomposition . . . 26
4.3. Diversity for Bregman Divergences . . . 28
4.4. Diversity of the 0/1-loss . . . 30
4.5. Dependency of diversity on outcomes . . . 38

5. Growth Strategies 39
5.1. Diversity is a measure of model fit . . . 39
5.2. Diversity in Random Forests and in Neural Networks . . . 40
5.3. Guided sequential training of member models . . . 41
5.4. Guided parallel training of member models . . . 49

6. Conclusion 52
6.1. Summary . . . 52
6.2. Outlook . . . 52

Appendix 55

A. Full experiment results 56

B. Proofs and additional results 59

C. Implementation 64

Bibliography 65

1. Introduction

Random Forests are some of the most successful and widely used methods for machine learning [1, 2]. Despite their intuitive formulation, it is not yet fully understood when and why Random Forests work well. A Random Forest is an ensemble of randomized decision trees. If all trees were very similar, we could not expect to gain much over just using a single tree. It seems reasonable to suspect that differences between individual trees must be essential to the performance gain. How are the differences between individual trees related to the quality of the Random Forest model?

We approach this question from the perspective of ensemble learning. The key challenge is to formally grasp a concept of *diversity* of the ensemble members that can be related to the ensemble generalisation error. This has been an ongoing line of inquiry for over 20 years. Even though already in 1995 and 1996, decompositions of the generalisation error that describe the interactions between ensemble members have been published, grasping the concept in full generality has remained elusive. Results were either limited to specific loss functions or remained disconnected from the ensemble error [5].

In this thesis, we take up two very recent publications [6, 7] which, as we will illustrate, provide an exhaustive and unifying framework for analysing ensembles. The central piece is a decomposition of the ensemble error into average member bias, average variance and a diversity term [6]. We provide an alternate derivation of this decomposition and link it to the theory of ensemble competence [7]. Further, we consider the role of diversity in Random Forests.

In Section 1.1 and Figure 1.1, we provide an overview of the contents of this thesis. In Section 1.2, we summarise our contributions.

1.1. Overview

We begin with a brief introduction to supervised learning and the statistical language used to analyse it (⇝Sections 1.3 and 1.4). We motivate and define the well-known bias-variance decomposition for the squared-error loss and proceed to generalise it to arbitrary loss functions (⇝Section 1.5). We define a particular family of loss functions called Bregman divergences (⇝Section 4.3).

We then turn our attention towards ensemble learning (⇝Chapter 2). We review several results that indicate when and why ensemble learning is effective. A key insight is that the ensemble performance depends on the diversity of predictions of the member models. We note that many of these results have been derived only for specific loss functions, or under assumptions. As a prominent example of ensemble learning techniques we introduce Random Forests (⇝Chapter 3).

In Chapter 4, we work towards formally expressing diversity as a component of the ensemble generalisation error. We first review some *ad-hoc* diversity measures. We then consider the *ambiguity decomposition* which was initially proven independently for the squared-error loss and the KL-divergence. We show that ambiguity, like bias and variance, can be generalised to arbitrary loss functions in terms of its effect on the error. This leads to a decomposition of the ensemble generalisation error into average member bias, average member variance and *diversity*, which is the expected ambiguity. Equipped with the tools to rigorously analyse ensemble diversity, we

consider two main settings: Bregman divergences for regression tasks and the 0/1-loss for classification tasks.

The developed theory is then applied in Random Forest ensembles. We argue that diversity is a highly relevant aspect of Random Forests, much more so than for neural network ensembles (↝Section 5.2). We introduce a method to regulate diversity in 0/1-classification ensembles via example weights (↝Section 5.3).

Finally, we propose a full generalisation of the Negative Correlation Learning scheme for neural networks (↝Section 5.4).

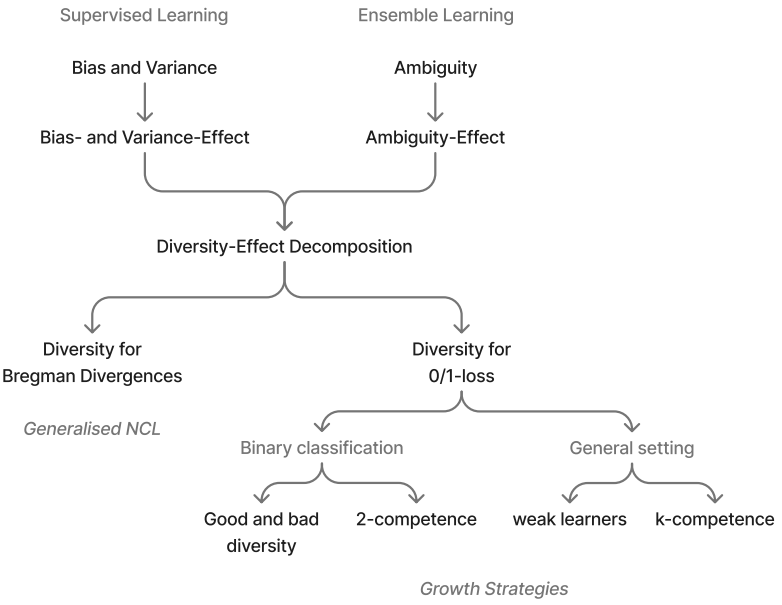

Figure 1.1.: Overview of the main theoretical topics in this thesis.

1.2. Contributions

We give a thorough and coherent introduction to ensemble diversity. We derive the fully general diversity-effect decomposition (↝4.2.2) from first principles and then show how the diversity decomposition (↝4.3.3) is a special case of it. This is a different approach to the original publication [6]. Wood et al. start from assuming that there exists *a* bias-variance decomposition and *an* ambiguity decomposition. Seeing that effect-decompositions trivially hold for any loss, we develop an alternate line of argument. For Bregman divergences, the effect-terms coincide with the corresponding terms in the specialised decomposition (↝Lemmas 4.3.1 and 4.3.2). For loss functions such as the 0/1-loss for which this is not possible, the general decomposition is still useful.

We review several arguments on ensemble diversity and relate them to the diversity theory. We illustrate that many statements which were previously only proven for specific loss functions or under assumptions are covered by the diversity theory (↝Chapter 4). Informed by this theory, we infer practical guidelines for ensemble learning.

For decision trees, we give a rigorous theoretical foundation for the most widely used impurity measures (↝3.1.1). We show that constructing a tree according to an impurity measure implies that tree construction greedily optimises a specific loss function. Further, we show that the choice of impurity measure implies the leaf combiner function.

For Random Forests, we illustrate a simple way to express the ensemble generalisation error in terms of the structure of the model (↝3.2.1). Likewise, we can express the components of the bias-variance-diversity decomposition in terms of the structure of the Random Forest model (↝5.2.1). This is a first step towards bringing characteristics of a specific model into the general theory of ensemble diversity.

For 0/1-loss classification, we review and relate two assumptions that enable bounds on the generalisation error: the weak-learner assumption and competence. We show that the former is a special case of the latter (↝4.4.6). We then consider competence in detail. While competence is a sufficient condition for non-negative diversity-effect, it is not a necessary condition in non-binary classification problems. We define a generalisation of competence that we call k-competence and show that k-competence is equivalent to non-negative diversity-effect (↝Definition 4.4.6 and Theorem 4.4.8). This connects the theory on diversity developed by Wood et al. [6] to the analysis and bounds proven by Theisen et al. [7].

We motivate and propose schemes to regulate diversity in Random Forest ensembles. We show empirically that it is indeed possible to encourage diversity in Random Forests. While the theory informs us that a more diverse ensemble is not necessarily better, we do observe that for specific benchmark datasets our regularisation schemes allow us to obtain better and smaller forests as compared to the standard construction procedure.

Original theoretical results are marked with a ⋆-glyph.

1.3. Statistics

We want to reason in a general manner about algorithms which act on data. We use statistical language for this. Consider a data point X that is provided as input to an algorithm. Our intention is to say that X could be *any* data point from some kind of data source. In other words, we can consider X to be *random*, that is, X is a random variable that can take different values. Which values exactly it can take depends on the data source – the *distribution* of the data. If our data source is a fair 6-sided die, X can take values in $\Omega = \{1, \dots, 6\}$ and the distribution of values is constant where each outcome has probability $\frac{1}{6}$, i.e. $\mathbb{P}[1] = \cdots = \mathbb{P}[6] = \frac{1}{6}$.

A σ-algebra over Ω can be thought of a set of subsets of Ω, containing Σ, such that it is closed under complement and countable union and intersection.

Definition 1.3.1 *A probability space is a triple* $(\Omega, \Sigma, \mathbb{P})$ *where*

- Ω *is an arbitrary set modelling the sample space i.e. the set of all possible outcomes.*
- Σ *is a* σ*-algebra over* Ω*, modelling the set of events.*
- $\mathbb{P}$ *is a function* $\Sigma \to [0, 1]$ *such that* $\mathbb{P}(\Omega) = 1$ *and* $\mathbb{P}\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mathbb{P}(A_i)$ *for a countable collection of (pairwise disjoint) sets in* Σ*, and models the probability measure.*

In the following, we assume an underlying probability space implicitly.

Definition 1.3.2 *A random variable is a quantity that depends on a random event, i.e. a function* $\Omega \to M$ *(commonly, we have* $M = \mathbb{R}$*).*

Further, we not only want to reason about the behaviour of an algorithm with respect to one point X but many such points. A basic notion is the *expected value* of X.

Definition 1.3.3 *The probability density function f_X of a random variable X is a nonnegative function such that*

$$\mathbb{P}(a < X < b) = \int_a^b f_X(x)dx$$

Definition 1.3.4 *The expected value (expectation) of random variable X is defined as*

$$\mathbb{E}[X] := \int x \cdot f_X(x)\, dx$$

where the integral is over the support of X.

A function $g(X)$ of a random variable X is a random variable itself and $\mathbb{E}[g(X)] = \int g(x) f_X(x)\, dx$. To emphasize the random variable the function is dependent on, we sometimes mention it in the subscript and write $\mathbb{E}_X[g(X)]$.

Lemma 1.3.1 *(Linearity of Expectations) A basic property of expected values is that they are linear: For any two random variables X, Y and a constant α it holds that $\mathbb{E}[X + Y] = \mathbb{E}[X] + \mathbb{E}[Y]$ and $\mathbb{E}[\alpha X] = \alpha \mathbb{E}[X]$.*

Lemma 1.3.2 *(Law of total expectation) For a function g:*

$$\mathbb{E}_{(X,Y)}[g(X,Y)] = \mathbb{E}_X[\mathbb{E}_Y[g(X,Y) \mid X]]$$

As a special case of this, it holds that

$$X, Y \text{ independent} \quad \rightarrow \quad \mathbb{E}_X[\mathbb{E}_Y[g(X,Y)]] = \mathbb{E}_Y[\mathbb{E}_X[g(X,Y)]]$$

and we write $\mathbb{E}_{X,Y}[g(X,Y)] =_{def} \mathbb{E}_X[\mathbb{E}_Y[g(X,Y)]]$

A random variable is *discrete* if its set of outcomes is a countable set $\{x_i, \ldots, x_n\}$ with probabilities $\{p_i, \ldots, p_n\}$. The probability density function then is $f_X(x_i) = \mathbb{P}(X = x_i) = p_i$. The expected value of a discrete random variable is given by a sum.

$$\mathbb{E}[X] = \sum_{i=1}^{n} p_i x_i$$

The expected value is a statement depending on the entire distribution. Usually, we do not know the distribution itself, but only have a limited set of samples of it. For instance, we may have a certain number of data points or run an algorithm a certain number of times and observe its output. We can use this set of samples to approximate the value of the expectation. Given that the samples are independent and identically distributed, the arithmetic mean of samples approximates the expected value.

$$\frac{1}{n} \sum_{i=1}^{n} x_i \;\rightarrow\; \mathbb{E}[X] \quad \text{as } n \rightarrow \infty$$

1.4. Supervised Learning

Our goal is to find an algorithm that maps objects from $\mathcal{X}$ to outcomes in $\mathcal{Y}$. Objects are described by their *features*. These are commonly numerical, so $\mathcal{X}$ can be thought of as $\mathbb{R}^d$ where d is the number of features. We will call such a representation of an object an *example*.

In *classification* problems, the outcomes are discrete among k possible outcomes and we refer to them as *labels* or *classes*. For sake of simplicity, we identify these with integers, i.e. $\mathcal{Y}$ can be thought of as the set $\{1, \dots, k\}$. In *binary* classification, there are only two possible outcomes. Depending on what is more convenient, we assume either $\mathcal{Y} = \{0, 1\}$ or $\mathcal{Y} = \{-1, 1\}$. In *regression* problems, the outcomes are continous. We can think of $\mathcal{Y}$ as $\mathbb{R}$.

The desired mapping $q : \mathcal{X} \to \mathcal{Y}$ may be such that it is not feasible to come up with explicit rules of how to map examples to outcomes. However, we may try to algorithmically infer such a mapping from a given set of examples and their known outcomes. More specifically, we want to find a deterministic *learning algorithm*, also called *learner*, that, given a random input D, produces a mapping q_D [1]. We call q_D a *model* and note that it is dependent on D. The task of choosing and configuring a learning algorithm is known as *supervised learning*.

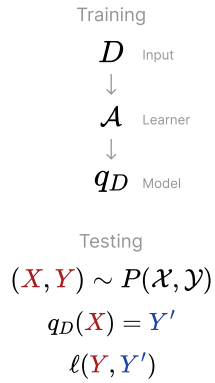

Figure 1.2.: Illustration of the main components of supervised learning. A learning algorithm $\mathcal{A}$ produces a model q_D given some input D. The model is then evaluated on example-outcome pairs of the original data distribution.

To be useful, a model should not only accurately estimate outcomes for the given training examples, but also provide reasonable predictions for examples that were not part of the input to the learning algorithm. Consider a probability distribution $P(\mathcal{X}, \mathcal{Y})$ from which realisations of example-training pairs are drawn. We write this as $(X, Y) \sim P(\mathcal{X}, \mathcal{Y})$ where (X, Y) are random variables from a joint distribution. This distribution is unknown – else the problem is trivially solved already. In order for our solution to be widely applicable, we strive to make as few assumptions about the distribution P as possible.

The training dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ is considered a random vector D drawn from $P(\mathcal{X}, \mathcal{Y})^n$ where n is the number of data points. If there are other sources of randomness in model construction such as, for instance, weight initialisation for neural networks, these can also be considered components of D.

Definition 1.4.1 *A model is a function $q : \mathcal{X} \to \mathcal{Y}$. In supervised learning, the model depends on the training input D. Its output (prediction) when queried with a random variable X taking values in $\mathcal{X}$ is written as*

$$q_D(X)$$

To shorten notation, we sometimes omit explicitly specifying either random variable and write q for $q_D(X)$. A dependency on both D and X is always to be understood.

The quality of a single model prediction is measured by a *loss function* $\ell : \mathcal{Y} \to \mathcal{Y}$ whose value should be low if the predicted outcome is close to the true outcome. To describe the expected loss across the entire distribution, we consider a pair of random variables $(X, Y) \sim P(\mathcal{X}, \mathcal{Y})$.

1: For the statistical analysis it is essential that the learning algoritm is regarded as deterministic. However, any kind of randomness can be introduced by providing it with random input.

Definition 1.4.2 *(Risk and Generalisation Error) The risk of a model q_D is the expected loss over all example-outcome pairs.*

$$Risk(q_D) =_{def} \mathbb{E}_{(X,Y)\sim P}\left[\ell(Y, q_D(X))\right]$$

The quality of a given learning algorithm $\mathcal{A}$ is the expected risk over all possible inputs D. We refer to this as the generalisation error.

$$GE(\mathcal{A}) =_{def} \mathbb{E}_D\left[Risk(q_D)\right] = \mathbb{E}_{(X,Y),D}\left[\ell(Y, q_D(X))\right]$$

Note that the choice of the loss function ℓ is fundamental to the evaluation of a learning algorithm.

1.5. Bias, Variance and their Effects

Since we are ultimately interested in the generalisation error of a learning algorithm, the question arises what forces influence it. To this end, one can strive to mathematically express the generalisation error in terms of meaningful quantities. A classical decomposition is the *bias-variance-decomposition*. It is an essential tool to understand learning algorithms in general and ensembles such as Random Forests in particular. For the purpose of this thesis, it is important to understand the decomposition and its motivation in detail. We will begin by considering the widely known bias-variance-decomposition for regression using the squared-error loss $\ell(y, y') =_{\text{def}} (y - y')^2$. We will then proceed to generalise the decomposition to arbitrary loss functions.

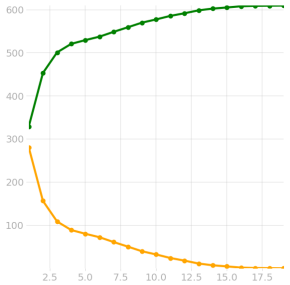

Figure 1.3.: Bias and variance of decision tree models of increasing maximum tree depth. With increasing tree depth, ● bias tends to decrease, as ● variance tends to increase.

The variance of a random variable with respect to the squared-error loss is defined as the expected distance in terms of loss to the the expected value.

$$\text{Var}_X\left(X\right) =_{\text{def}} \mathbb{E}_X\left[(X - \mathbb{E}_X\left[X\right])^2\right]$$

As such, $\mathbb{E}_X\left[X\right]$ is a centroid to the different realisations of X with respect to the loss function $\ell(Y, Y') = (Y - Y')^2$.

$$\mathbb{E}_X\left[X\right] = \arg\min_z \mathbb{E}_X\left[(X - z)^2\right]$$

Recall that a model is a function dependent on the training input D. The variance is a measure of how different the models produced by the learning algorithm will be in terms of outputs if supplied with different realisations of training input D.

$$\text{Var}_D\left(q_D(X)\right) = \mathbb{E}_{X,D}\left[(q_D(X) - \mathbb{E}_D\left[q_D(X)\right])^2\right]$$

$q^\star(X) =_{\text{def}} \mathbb{E}_D\left[q_D(X)\right]$ is the *central model* and does not depend on D.

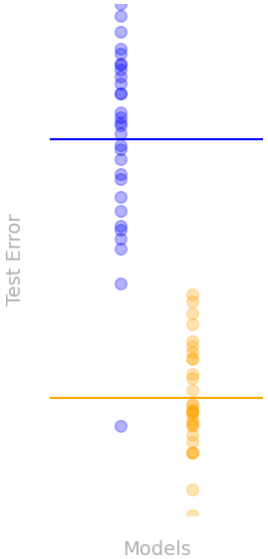

Figure 1.4.: Visualising the variance of ● decision tree and ● Random Forest models. Each glyph corresponds to the test error of one model trained on a random subset of the full available data. The variation of the test error around the mean test error across many dataset samples is exactly the variance. Not only do Random Forests show lower test errors on average, they also have lower variance.

Further, recall that we are considering a joint distribution of example-output pairs (X, Y). We can not assume that the outcomes are not ambiguous. Let $y(X)$ be the outcome associated with X. We measure the variance of actual outcomes $y(X)$ for a given example X around the expected outcome.

$$\text{Var}_{Y|X}\left(Y\right) = \mathbb{E}_Y\left[(Y - \mathbb{E}_{Y|X}\left[Y\right])^2\right]$$

$y^\star(X) =_{\text{def}} \mathbb{E}_{Y|X}\left[Y\right]$ is the *central label* and does not depend on Y.

Using this notation, the bias-variance decomposition for the squared-error loss is given as follows. Note that each variance term is the expected distance in terms of loss to a

certain centroid.

$$\begin{aligned}\underbrace{\mathbb{E}_{(X,Y),D}\left[(Y - q_D(X))^2\right]}_{\text{generalisation error}} &= \underbrace{\mathbb{E}_{(X,Y)}\left[(Y - y^\star(X))^2\right]}_{\text{Var}(Y)\ \text{("noise")}} \\ &+ \underbrace{\mathbb{E}_X\left[(y^\star(X) - q^\star(X))^2\right]}_{\text{Bias}(Y,q)\ \text{("learner bias")}} \\ &+ \underbrace{\mathbb{E}_{X,D}\left[(q^\star(X) - q_D(X))^2\right]}_{\text{Var}(q)\ \text{("learner variance")}}\end{aligned} \tag{1.1}$$

This decomposition is usually derived by expanding the square [8]. The cross-terms then vanish due to that $q^\star = \mathbb{E}_D[q_D]$ and $y^\star = \mathbb{E}_Y[y(X)]$. This is but a special case of a more general structure applying to a certain class of losses. We will provide a more general proof in lemmas 4.3.1 and 4.3.2.

The first term, $\text{Var}(Y)$ is independent of D and q_D. This means we have no means of influencing it with our choice of q_D. It is also referred to as *noise*, *bayes error* or *irreducible error*. The second term, $\text{Var}(q)$ measures the variance of our model around its non-random centroid model with respect to different realisations of the random training dataset D. This can be understood as a measure of spread of the learning algorithm with respect to different realisations of D. The third term, $\text{Bias}(q_D, Y)$ is the distance in terms of loss between the central model and the central label. This can be thought of as a measure of precision of the learning algorithm.

Note that we developed two things: On the one hand, we derived quantities that measure the notions of bias and variance. On the other hand, by virtue of these quantities appearing in the error decomposition 1.1, we have expressed the *effect* of these quantities on the generalisation error.

While the bias-variance decomposition for the squared-error loss is widely accepted, there are many competing decompositions for a range other loss functions [9–13, 29]. In this work, we are particularly interested in the 0/1-loss for classification. A decomposition of it where the model variance is independent of the outcome variable has been proven to not exist [6]. We will now argue that approaching the matter from the perspective of such *loss-effects* allows us to state a general bias-variance decomposition that holds for any loss function. The decomposition for the squared-error loss is a special case of it.

Definition 1.5.1 *(Loss-Effect) For a loss functon ℓ, and random variables Y, Z, Z', we define the change in loss between Z and Z' in relation to Y as:*

$$LE(Z, Z') =_{def} \ell(Y, Z') - \ell(Y, Z)$$

Note that the arguments of loss-effect appear in inverse order in the difference. This is to give the expression a suggestive shape since in section 4.3, we will show that, for a Bregman divergence B_ϕ, it holds that

$$\mathbb{E}[\text{LE}(Z', Z)] = B_\phi(Z', Z)$$

Definition 1.5.2 *(Central model) Given a model q, it's central model is the centroid with respect to D:*

$$q^\star =_{def} \arg\min_z \mathbb{E}_D[\ell(z, q_D)]$$

Definition 1.5.3 *(Central label) The central label is*

$$y^\star =_{def} \mathbb{E}_{Y|X}[Y]$$

The *variance-effect* is the expected change in loss due to using q_D instead of the non-random centroid $q^\star$. Likewise, the *bias-effect* is the expected change in loss due to using the expected model instead of the expected label. In the following, to lighten notation, we will omit explicitly stating the dependence on X. Formally:

In the original publication [14], bias-effect is called the *systematic effect*, i.e. the effect of the systematic components. However, it is clearer to call this *bias-effect*, particularly when we begin to introduce notions of diversity in 4.1.

$$\text{Bias-Effect} =_{\text{def}} \mathbb{E}_{(X,Y)}\left[\ell(Y, q^\star) - \ell(Y, y^\star)\right]$$
$$\text{Variance-Effect} =_{\text{def}} \mathbb{E}_{(X,Y),D}\left[\ell(Y, q_D) - \ell(Y, q^\star)\right]$$

This allows us to state a decomposition of the generalisation error simply in terms of loss-effects.

$$\mathbb{E}\left[\ell(Y, q)\right] = \mathbb{E}\left[\underbrace{\ell(Y, y^\star)}_{\text{"noise"}} + \underbrace{\ell(Y, q^\star) - \ell(Y, y^\star)}_{\text{"bias-effect"}} + \underbrace{\ell(Y, q) - \ell(Y, q^\star)}_{\text{"variance-effect"}}\right]$$

Note that the individual terms on the right-hand side simply cancel out and reduce to $\ell(Y, q)$. As illustrated in figure 1.5, this decomposition divides the the interval from $\ell(Y, Y) = 0$ to $\ell(Y, q)$ into meaningful sections. Note that this decomposition depends solely on the linearity of expectation and is independent of the loss function ℓ or the definitions of $y^\star$ and $q^\star$.

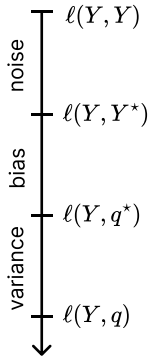

Figure 1.5.: Illustration how the bias-variance-effect decomposition decomposes the loss $\ell(Y, q)$ into meaningful segments.

For the squared-error loss $\ell(Z, Z') = (Z - Z')^2$, bias-effect equals bias and variance-effect equals variance.

$$\mathbb{E}\left[\ell(Y, q^\star) - \ell(Y, y^\star)\right] = \mathbb{E}\left[\ell(y^\star, q^\star)\right]$$
$$\mathbb{E}\left[\ell(Y, q) - \ell(Y, q^\star)\right] = \mathbb{E}\left[\ell(q^\star, q)\right]$$

Thus, the bias-variance decomposition for the squared-error loss is a special case of this.

Theorem 1.5.1 *(Bias-Variance-Effect-Decomposition [14]) For any loss function L, it holds that*

$$\mathbb{E}_{(X,Y),D}\left[\ell(Y, q_D)\right] = \underbrace{\mathbb{E}_{(X,Y)}\left[\ell(Y, y^\star)\right]}_{\textit{noise}} + \underbrace{\mathbb{E}_{(X,Y)}\left[LE(q^\star, y^\star,)\right]}_{\textit{bias-effect}} + \underbrace{\mathbb{E}_{(X,Y),D}\left[LE(q_D, q^\star,)\right]}_{\textit{variance-effect}}$$

Note that while for the squared error the variance compares *model predictions*, the variance-effect is based solely on the *change in loss*. In section 4.3, we will see that the former is in fact only a special property of a specific family of loss functions.

1.6. Bregman Divergences and Centroids

To measure the difference between predicted and ground-truth outcomes, we use a loss function ℓ. The choice of loss function depends on the data domain, the learning task and computational considerations. Examples are the squared-error loss [15] for regression, or the 0/1-loss [7] or the KL-divergence [16] for classification. The well-known bias-variance decomposition for the squared-error loss is usually shown directly in teaching materials [8, 17]. The question then arises which properties are specific to the loss function and which are part of a more general structure.

We will now define a family of loss functions, called *Bregman divergences*, that encompasses many widely used loss functions in supervised learning (see Table 1.1).

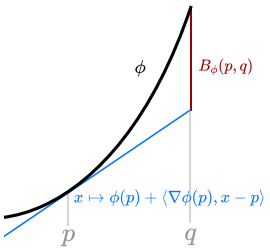

Figure 1.6.: Given a strictly convex generator ϕ, the Bregman divergence for points p, q is the difference between the linear approximation around p and ϕ at the point q.

Definition 1.6.1 *(Bregman Divergence [13, 18]) The Bregman divergence* $B_\phi(p, q) : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ *is defined based on a generator function* ϕ *as follows:*

$$B_\phi(p, q) =_{def} \phi(p) - \phi(q) - \langle \nabla\phi(q), (p - q) \rangle$$

where $\langle \cdot, \cdot \rangle$ *is the inner product,* $\nabla\phi(q)$ *is the gradient vector of* ϕ *at* q *and* $\phi : \mathcal{S} \to \mathbb{R}$ *is a strictly convex function on a convex set* $\mathcal{S} \subseteq \mathbb{R}^k$ *such that it is differentiable on the relative interior of* $\mathcal{S}$*.*

Table 1.1.: Examples of commonly used loss functions that are Bregman divergences [6, 19]

Divergence $B_\phi(p, q)$	Generator $\phi(q)$	Domain S	Loss function
$(p - q)^2$	q^2	$\mathbb{R}$	Squared Error
$x \log\left(\frac{x}{y}\right) + (1 - x) \log\left(\frac{1-x}{1-y}\right)$	$x \log x + (1 - x) \log(1 - x)$	$[0, 1]$	Logistic loss
$\frac{x}{y} - \log\left(\frac{x}{y}\right) - 1$	$-\log x$	$\mathbb{R}_{>0}$	Ikura-Saito distance
$\|\| x - y \|\|^2$	$\|\| x \|\|^2$	$\mathbb{R}^d$	Squared Euclidean distance
$(x - y)^\top A(x - y)$	$x^\top A y$	$\mathbb{R}^d$	Mahalanobis distance
$\sum_{j=1}^d x_j \log_2\left(\frac{x_j}{y_j}\right)$	$\sum_{j=1}^d x_j \log_2 x_j$	d-simplex	KL-divergence
$\sum_{j=1}^d x_j \log\left(\frac{x_j}{y_j}\right) - \sum_{j=1}^d (x_j - y_j)$	$\sum_{j=1}^d x_j \log x_j$	$\mathbb{R}^d_{\geq 0}$	Generalized I-divergence
$\sum_{j=1}^d x_j \log x_j$	$\sum_{j=1}^d x_j \log x_j$	$\mathbb{R}_{\geq 0}$	Poisson loss

In order to talk about variances with respect to Bregman divergences, we need a notion of a centroid with respect to a Bregman divergence. Bregman divergences are in general not symmetric and hence there is a *left* and *right* centroid.

Lemma 1.6.1 *(Left and right Bregman centroids, [13]) Let* B_ϕ *be a Bregman divergence of generator* $\phi : \mathcal{S} \to \mathbb{R}$*. For a random variable* Y *taking values in* $\mathcal{S}$*, it holds that*

- *the right Bregman centroid is*

$$\arg\min_z \mathbb{E}_X \left[B_\phi(X, z) \right] = \mathbb{E}[X]$$

- *the left Bregman centroid is*

$$\arg\min_z \mathbb{E}_X \left[B_\phi(z, X) \right] = (\nabla\phi)^{-1} \mathbb{E} \left[\nabla\phi(X) \right] =_{def} \mathfrak{C}[X]$$

If B_ϕ *is symmetric, i.e.* $B_\phi(Y, Y') = B_\phi(Y', Y)$ *then* $\mathbb{E}[X] = \mathfrak{C}[X]$*.*

Definition 1.6.2 *(Dual expectation [13]) The left Bregman centroid is the expected value in the dual space implied by* $\nabla\phi$*. Due to this, we define the dual expectation as*

$$\mathfrak{C}[X] =_{def} (\nabla\phi)^{-1} \mathbb{E} \left[\nabla\phi(X) \right]$$

In accordance with Definition 1.5.2, for Bregman divergences, $q^\star$ is the left Bregman centroid and $y^\star$ is the right Bregman centroid.

A generalised measure of variance is then the expected divergence around a Bregman centroid.

Definition 1.6.3 *The variance around the right Bregman centroid is known as the Bregman information* $I_\phi(X)$ *[19].*

$$I_\phi(X) =_{def} \mathbb{E}_X \left[B_\phi \left(X, \mathbb{E}_X \left[X \right] \right) \right]$$

In other words, the choice of loss function implies a measure of variance. Various well-known variance measures can now be seen to actually be implied by a Bregman divergence. For example, let $X = \{X_1, \ldots, X_n\} \subset \mathbb{R}^d$. Then the squared Euclidean distance corresponds to the *sample variance* [19].

$$B_\phi \left(p, q \right) = || \ p - q \ ||^2 \ \rightarrow \ \ I_\phi(X) = \frac{1}{n} \sum_{i=1}^{n} (X_i - \mathbb{E} \left[X \right])^2$$

For the KL-divergence, the Bregman information is the *mutual information*. Consider a random variable X over probability distributions with probability measure p [19].

$$B_\phi \left(u, v \right) = \sum_{j=1}^{d} u_j \log \left(\frac{u_j}{v_j} \right) \ \rightarrow \ I_\phi(X) = \sum_{i=1}^{n} \sum_{j=1}^{m} p \left(u_i, v_j \right) \log \frac{p \left(u_i, v_j \right)}{p \left(u_i \right) p \left(v_j \right)}$$

2. Ensemble Learning

Ensemble Learning is the method of training M individual models $q_1, \ldots, q_M$ for a given task and aggregating their outputs via an *ensemble combiner* $\bar{q}$ to form an ensemble prediction [20]. The individual models $q_1, \ldots, q_M$ are referred to as *members*. When all members are constructed using the same learning algorithm, we call it a *homogeneous* ensemble. The learning algorithm is then referred to as the *base learner*. Otherwise the ensemble is *heterogeneous*.

2.1. Methods

There are three main variants of ensemble learning [21]:

- *Parallel*: All members are trained independently. The outputs of all members are then aggregated to form the ensemble prediction.
- *Stacking* or *Meta-Learning*: All members are trained independently. The member outputs serve as input data for another learning algorithm, which then provides the ensemble prediction.
- *Sequential*: Members are trained in sequence. The output of the previous ensemble member informs the construction of the next member.

Random Forests [22] are an example of parallel ensemble construction. M decision trees are constructed independently and the tree's predictions are aggregated by a kind of mean (see section 3.1). A classical example for sequential ensemble construction is *Boosting* [23]. In boosting algorithms, the ensemble combiner $\bar{q}$ is not a mean but a (weighted) sum $\bar{q} = \sum_{i=1}^{M} \alpha_i q_i$. The first member q_1 provides a base prediction. Successive members are then trained to predict not an output value but *increments* (*pseudo-targets*) to the base prediction such that the sum $\alpha_1 q_1 + \alpha_2 q_2 + \ldots$ moves towards a more precise prediction. In this thesis, we will focus on the Random Forest learner and variations of it.

2.2. Notation

In the supervised learning setting with a single model, we have defined the model as a function $q_D(X)$. In ensemble learning, M individual models are constructed and their outputs are aggregated via an ensemble combiner $\bar{q}$ to form an ensemble output. The individual models are referred to as *members*.

Analysing ensembles means analysing differences between the members. There are two dimensions in which ensemble members can differ. First, in terms of provided training data. [1] Second, in terms of other randomness used in model construction. [2] To distinguish these two sources, we sometimes denote the training data as a random vector $D = (D_1, \ldots, D_M)$ and other member parameters as $\Theta = (\Theta_1, \ldots, \Theta_M)$.

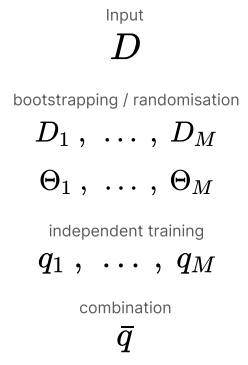

Figure 2.1.: Illustration of parallel ensemble learning.

1: For instance, each learner may be trained on a random subset of the available training data (see Section 3.2)

2: For instance, random numbers used in decision tree construction (see Section 3.1).

Definition 2.2.1 *(Ensemble member model) The i-th ensemble member model q_i is a function depending on training input D_i and additional parameters Θ_i.*

$$q_{D_i,\Theta_i}(X) : \mathcal{X} \to \mathcal{Y}$$

To shorten notation, we also write $q_i =_{def} q_{D_i,\Theta_i}(X)$.

Due to that the i-th member depends only on D_i and Θ_i and is independent of D_j and Θ_j for $j \neq i$, using the law of total expectation (see Lemma 1.3.2), we can write

$$\mathbb{E}_D[q_i] = \mathbb{E}_{D_{j\neq i}}[\mathbb{E}_{D_i}[q_i]] = \mathbb{E}_{D_i}[q_i]$$

and likewise for Θ.

The output of an ensemble is produced by aggregating the outputs of the member models using a combiner function.

Definition 2.2.2 *(Ensemble combiner) The ensemble combiner $\bar{q} : \mathcal{X} \to \mathcal{Y}$ for a given loss function ℓ is the centroid with respect to model parameters Θ:*

$$\bar{q} =_{def} \arg\min_z \mathbb{E}_\Theta[\ell(z, q_\Theta)]$$

Note that this is the central model with respect to Θ (⇝1.5.2). For Bregman divergences, this is the left Bregman centroid, i.e. the the dual expectation (⇝1.6.2). This definition agrees with combiners commonly used in practise (plurality vote, arithmetic mean). It links combiners to loss functions. Further, it suggests combiners for other loss functions. In each case, using a combiner as defined here enables a very general and expressive theory of ensemble diversity and a decomposition of the ensemble generalisation error exactly measuring ensemble improvement due to diversity.

For the 0/1-loss for a k-class classification problem, the implied combiner is the plurality vote:

$$\bar{q} = \arg\min_{z\in[k]} \mathbb{E}_\Theta\left[\ell_{0/1}(z, q_\Theta)\right]$$

For Bregman divergences, i.e. $\ell = B_\phi$, the combiner implied by ℓ is the dual expectation:

$$\begin{aligned}\bar{q} &= \mathscr{E}_\Theta\left[q_{\Theta,D}\right] \\ &= (\nabla\phi)^{-1}\mathbb{E}_\Theta\left[\nabla\phi(q_\Theta)\right]\end{aligned}$$

The squared-error loss is a symmetric Bregman divergence and hence the ensemble combiner is the arithmetic mean:

$$\bar{q} = \mathscr{E}_\Theta[X] = \mathbb{E}_\Theta[X] \approx \frac{1}{M}\sum_{i=1}^{M} q_i$$

An ensemble is homogeneous if and only if $\Theta_1, \ldots, \Theta_M$ are identically and independently distributed. This is the case if the member models are constructed according to the same base learner and do not influence each other.

Lemma 2.2.1 ⋆ *(Generalised from [24]) In homogeneous ensembles, the central models of any two members and the combiner are the same. That is, for any $i, j \in \{1, \ldots, M\}$ it holds that*

$$q_i^\star = q_j^\star = \bar{q}^\star$$

Proof. If $D_1, \ldots, D_M$ and $\Theta_1, \ldots, \Theta_M$ are identically distributed and independent, it holds that

$$q_i^\star = \mathbb{E}_D[q_{D_i,\Theta_i}] = \mathbb{E}_{D_i}[q_{D_i,\Theta_i}] = \mathbb{E}_{D_j}\left[q_{D_j,\Theta_i}\right] = q_j^\star$$

For the ensemble combiner, it holds that

this is showing it only for Bregman divergences

$$\begin{aligned}\bar{q}^\star &= \mathscr{E}_\Theta[q_{D,\Theta}]^\star = \mathscr{E}_D[\mathscr{E}_\Theta[q_{D,\Theta}]] \\ &= (\nabla\phi)^{-1}\mathbb{E}_D\left[(\nabla\phi)(\nabla\phi)^{-1}\mathbb{E}_\Theta[q_{D,\Theta}]\right] \\ &= \mathscr{E}_\Theta[\mathscr{E}_D[q_{D,\Theta}]] = \mathscr{E}_\Theta\left[q_\Theta^\star\right]\end{aligned}$$

Due to the result above, $q_\Theta^\star$ is constant over Θ and thus $\mathscr{E}_\Theta\left[q_\Theta^\star\right] = q_\Theta^\star$. □

A basic measure for classification ensembles under the 0/1-loss is the winning margin of the majority vote.

Definition 2.2.3 *(Ensemble margins for majority voting [22]) The margin for class Y of an example X is the difference between the number of member votes for class Y and the number of votes for the next-best class.*

$$mr(X, Y) =_{def} \frac{1}{M} \sum_{i=1}^{M} \mathbb{1}\left[q_i = Y\right] - \max_{j \neq y} \frac{1}{M} \sum_{i=1}^{M} \mathbb{1}\left[q_i = j\right] \quad \in [-1, 1]$$

2.3. Motivation

We will now review some classical arguments that motivate ensemble learning. We do this to provide context to the results in Chapter 4, which also show when and how ensemble learning is beneficial.

Ensemble improvement is non-negative The arithmetic mean combiner can be seen as approximating an expectation over member models, i.e. $\bar{q} = \mathbb{E}_\Theta\left[q_\Theta\right] \approx \frac{1}{M} \sum_{i=1}^{M} q_i$. This motivated [25] to invoke Jensen's inequality . For a loss function ℓ that is convex in its second argument, it holds that

Jensen's inequality, in a probabilistic setting, states that, for a function $\phi : \mathbb{R} \to \mathbb{R}$ and a random variable X

ϕ convex $\to \phi\left(\mathbb{E}\left[X\right]\right) \leq \mathbb{E}\left[\phi(X)\right]$

$$\underbrace{\ell(Y, \mathbb{E}_\Theta\left[q_\Theta(X)\right])}_{\text{"ensemble loss"}} \quad \leq \quad \mathbb{E}_\Theta\left[\underbrace{\ell(Y, q_\Theta(X))}_{\text{"member loss"}}\right]$$

and thus

$$\mathbb{E}_\Theta\left[\ell(Y, q_\Theta(X))\right] - \ell(Y, \mathbb{E}_\Theta\left[q_\Theta(X)\right]) \geq 0$$

Corollary 2.3.1 *For convex loss functions and using the arithmetic mean combiner, ensembling can never hurt performance: The ensemble loss is always smaller-equal than the average member error.*

Abe et al. [25] interpret the difference between these quantities, as a measure of ensemble improvement, i.e. the gain from using an ensemble instead of a single member.

Ensemble bias equals average member bias It can be shown that the bias of a homogeneous ensemble is equal to the average bias of the ensemble members. We will give an illustrative argument for the arithmetic mean combiner here. We will show this in detail in a more general and inuitive way in chapter 4.

Lemma 2.3.2 *(Ensemble bias equals average member bias under arithmetic mean combiner in homogeneous ensembles [24])*

$$\mathbb{E}_X\left[\ell(y^\star(X), \bar{q}^\star(X))\right] = \mathbb{E}_X\left[\ell(y^\star(X), q^\star(X))\right]$$

Proof. Consider individual learner inputs $D = (D_1, \ldots, D_M)$. Each member depends on some D_i and the combiner $\bar{q}$ depends on D. Assuming that $D_1, \ldots, D_M$ are

independent and identically distributed, we can write

$$\mathbb{E}_{D,\Theta}\left[\bar{q}\right] = \mathbb{E}_D\left[\frac{1}{M}\sum_{i=1}^{M} q_{D_i}\right] = \frac{1}{M}\sum_{i=1}^{M}\mathbb{E}_{D_i}\left[q_{D_i}\right] = \mathbb{E}_{D'}\left[q_{D'}\right]$$

where D' is distributed as any D_i. We can conclude $\mathbb{E}_D\left[\bar{q}\right] = \mathbb{E}_D\left[q_D\right]$ (see section 2.2). This implies

$$\bar{q}^\star = \mathbb{E}_D\left[\bar{q}\right] = \mathbb{E}_D\left[q_D\right] = q^\star$$

and thus the bias of the ensemble is the same as the bias of a member model q. □

This argument depends on the linearity of expectations, the arithmetic mean combiner and the fact that the ensemble is homogeneous. We will later show this more directly for any loss function and any combiner (see 4.2).

Corollary 2.3.3 *For homogeneous ensembles under the arithmetic mean combiner, ensemble improvement is solely due to variance reduction.*

Variance reduction for squared-error regression Consider the regression setting under the squared-error loss and the arithmetic mean combiner. Assume $q_1, \ldots, q_M$ are identically and independently distributed with equal variance σ^2.

$$\mathrm{Var}\left(\bar{q}\right) = \mathrm{Var}\left(\frac{1}{M}\sum_{i=1}^{M} q_i\right) = \frac{1}{M^2}\sum_{i=1}^{M}\mathrm{Var}\left(q_i\right) = \frac{1}{M}\sigma^2$$

As the number of members M increases, the ensemble variance is reduced. Further, one can also see that the interactions between members determine the variance reduction. Assume ensemble members have equal pairwise covariance. Then

$$\rho =_{\text{def}} \frac{\mathrm{Cov}(q_i, q_j)}{\sigma^2} \quad \leftrightarrow \mathrm{Cov}(q_i, q_j) = \rho\sigma^2$$

Further,

$$\mathrm{Var}\left(\bar{q}\right) = \mathrm{Var}\left(\frac{1}{M}\sum_{i=1}^{M} q_i\right) = \frac{1}{M^2}\left(\underbrace{\sum_{i=1}^{M}\mathrm{Var}\left(q_i\right)}_{M\sigma^2} + \underbrace{2\sum_{i<j}^{M}\mathrm{Cov}(q_i, q_j)}_{M(M-1)\rho\sigma^2}\right) = \frac{\sigma^2}{M} + \frac{M-1}{M}\rho\sigma^2$$

One can show that $\rho \geq 0$ [24].

Corollary 2.3.4 *Under the squared-error loss, ensemble variance is minimised if member outputs are uncorrelated.*

Variance reduction for classification margins For classification, a classical analysis is the bound given by Breiman [22], in which Random Forests are first introduced. The basic idea is to consider variances with respect to the ensemble margin (see definition 2.2.3). The analysis is enabled by imposing an assumption on the performance of member models. Let $s =_{\text{def}} \mathbb{E}_{X,Y}\left[\mathrm{mr}(X, Y; D)\right]$ be the *strength* of the ensemble. We assume s to be be non-negative. Intuitively, this means that the ensemble is more

likely to be correct than incorrect. For binary classification, this is equivalent to the assumption that a member predicts the correct class with probability $1/2$. This is exactly the the weak-learner assumption which we treat in detail in Definition 4.4.4.

The assumption that $s \geq 0$ enables us to open with Chebychev's inequality to bound the generalisation error in terms of the variance of the ensemble margin.

$$\mathbb{E}\left[\ell(Y, \bar{q}(X))\right] = \mathbb{P}\left[\text{mr}(X, Y; D) < 0\right] \;\leq\; \frac{\text{Var}\left(\text{mr}(X, Y; D)\right)}{\mathbb{E}_{X,Y}\left[\text{mr}(X, Y; D)\right]^2}$$

We can already see that we have an interaction between the performance of the individual members, as reflected in the ensemble margin $\mathbb{E}_{X,Y}\left[\text{mr}(X, Y; D)\right]$, and the variance of the margin. The generalisation error is in part determined by the ratio of these two quantities.

Note that

$$\text{mr}(X, Y; D) = \mathbb{E}_\Theta\left[\underbrace{\mathbb{1}\left[q_i = Y\right] - \mathbb{1}\left[q_i = K\right]}_{=_{\text{def}}\ \text{rmg}(X,Y,\Theta)} \mid (X, Y), D\right]$$

where K is the next-best class and we define the *raw margin function* $\text{rmg}(X, Y, \Theta)$ to be the inner part of that expectation. So, $\text{mr}(X, Y; D) = \mathbb{E}_\Theta\left[\text{rmg}(X, Y, \Theta) \mid (X, Y), D\right]$.

Theorem 2.3.5 *([22]) The variance of the ensemble margin can be expressed in terms of the covariance between the raw member margins of two members parameterised by i.i.d* Θ, Θ'*.*

$$Var_{X,Y}\left(mr(X, Y; D)\right) = \mathbb{E}_{\Theta,\Theta'}\left[Cov_{(X,Y)}(rmg(\Theta), rmg(\Theta'))\right]$$

Proof. For brevity, we write $Z =_{\text{def}} (X, Y)$, $\text{mr}(Z) =_{\text{def}} \text{mr}(X, Y; D)$ and $\text{rmg}(\Theta) =_{\text{def}} \text{rmg}(X, Y; \Theta)$. By the definition of variance, have

$$\begin{aligned}\text{Var}_Z\left(\text{mr}(Z)\right) &= \mathbb{E}_Z\left[\left(\text{mr}(Z) - \mathbb{E}_Z\left[\text{mr}(Z)\right]\right)^2\right] \\ &= \mathbb{E}_Z\left[\text{mr}(Z)^2\right] - \mathbb{E}_Z\left[\text{mr}(Z)\right]^2\end{aligned}$$

For the left-hand-side term, by the rule of iterated expectation and the fact that Z and Θ are independent (see lemma 1.3.2), it holds that

$$\mathbb{E}_Z\left[\text{mr}(Z)^2\right] = \mathbb{E}_Z\left[\mathbb{E}_\Theta\left[\text{rmg}(\Theta) \mid Z\right]^2\right] = \mathbb{E}_{Z,\Theta}\left[\text{rmg}(\Theta)^2\right] = \mathbb{E}_\Theta\left[\mathbb{E}_Z\left[\text{rmg}(\Theta)^2\right]\right]$$

For the right-hand-side term, we can make use of the fact that, for some function f, it holds that $\mathbb{E}_\Theta\left[f(\Theta)^2\right] = \mathbb{E}_{\Theta,\Theta'}\left[f(\Theta)f(\Theta')\right]$ where Θ and Θ' are independent and identically distributed. We apply the rule of iterated expectation and exploit that Z and Θ are independent.

$$\begin{aligned}\mathbb{E}_Z\left[\text{mr}(Z)\right]^2 &= \mathbb{E}_Z\left[\mathbb{E}_\Theta\left[\text{rmg}(\Theta) \mid Z\right] \cdot \mathbb{E}_{\Theta'}\left[\text{rmg}(\Theta') \mid Z\right]\right] \\ &= \mathbb{E}_Z\left[\mathbb{E}_\Theta\left[\text{rmg}(\Theta)\right]\mathbb{E}_{\Theta'}\left[\text{rmg}(\Theta')\right]\right] \\ &= \mathbb{E}_{\Theta,\Theta'}\left[\mathbb{E}_Z\left[\text{rmg}(\Theta)\right]\mathbb{E}_Z\left[\text{rmg}(\Theta')\right]\right] \\ &= \mathbb{E}_\Theta\left[\mathbb{E}_Z\left[\text{rmg}(\Theta)\right]^2\right]\end{aligned}$$

In summary, we can conclude that the variance of the ensemble margin is equal to the expected variance of the raw classifier margin. This variance can then also be expressed as the covariance between two independent, identically distributed random variables Θ and Θ'.

$$\begin{aligned}
\mathrm{Var}_Z\left(\mathrm{mr}(Z)\right) &= \mathbb{E}_\Theta\left[\mathbb{E}_Z\left[\mathrm{rmg}(\Theta)^2\right]\right] - \mathbb{E}_\Theta\left[\mathbb{E}_Z\left[\mathrm{rmg}(\Theta)\right]^2\right] \\
&= \mathbb{E}_\Theta\left[\mathbb{E}_Z\left[\mathrm{rmg}(\Theta)^2\right] - \mathbb{E}_Z\left[\mathrm{rmg}(\Theta)\right]^2\right] \\
&= \mathbb{E}_\Theta\left[\mathrm{Var}_Z\left(\mathrm{rmg}(\Theta)\right)\right] \\
&= \mathbb{E}_{\Theta,\Theta'}\left[\mathrm{Cov}_Z(\mathrm{rmg}(\Theta), \mathrm{rmg}(\Theta'))\right]
\end{aligned}$$

□

Corollary 2.3.6 *Under the* 0/1*-loss in classification ensembles, the ensemble error can be bounded from above by a pointwise covariance between ensemble margins. The ensemble error is lower if individual members are uncorrelated.*

Unfortunately, due to the initial application of Chebychev's inequality, this is only an upper bound.

3. Random Forests

In this chapter, we describe the Random Forest learning algorithm. We first motivate and describe decision trees, which are the basic components of a Random Forest. We then proceed to describe a particular property of decision trees: they are likely to exhibit high variance. While this is undesirable for a learning algorithm, we will see that combining several randomized decision trees into a Random Forest ensemble turns exactly that property into a crucial advantage (see also Section 4.2).

A Random Forest is a collection of randomized decision trees. A decision tree is a data-driven recursive partitioning scheme, combined with a means to produce a prediction based on the training points in a partition cell. The Random Forest prediction then is an aggregate of the predictions of all individual trees.

3.1. Decision Trees

We are interested in learning algorithms that, given some training data, produce a model that is able to predict a reasonable outcome when queried with a previously unseen example (see Section 1.4) One intuitive approach is to consider the examples in the training data that are "close" to the query point. Then, one might claim that the outcome for the query point must surely be similar to the outcomes of the close points – which we already know. Indeed, finding a proper notion of "closeness" is at the heart of many machine learning algorithms such as k-Nearest-Neighbours, k-Means, Support Vector Machines, and others.

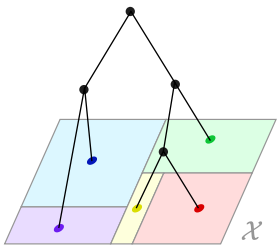

Figure 3.1.: Rendering of a decision tree structure. Each inner node corresponds to a partitioning of the parent edge. In standard decision trees, this is a binary partition. In other words, the examples are *split* at a certain value threshold in a certain feature dimension.

Constructing a decision tree means recursively partitioning the input space $\mathcal{X}$, guided by the training data D. Then, given a query X, we check the partition cell that X belongs to and all the training examples that are in it. These are the examples we consider "close" to X. The tree's prediction will be an aggregation of the outcomes of all training points in that cell. The partition is constructed greedily and recursively: At each iteration, the parent cell is split into two child cells based on a local impurity criterion.

Because we are recursively partitioning the input space, we have at hand a tree structure of decision rules. The cells of the resulting partition are the leaves of the tree. The non-leaf nodes are also referred to as *decision nodes*, but there is no inherent difference between leaf and non-leaf nodes. We will use either of the terms *leaf* and *cell*, depending which aspect we want to emphasize.

There are the following main components to the implementation of a decision tree:

- The *splitting criterion* to apply recursively to subsets of the training data.
- The *stopping criterion* that determines whether a node should be split further. This will determine the depth of the decision tree.
- The *leaf aggregation function* that produces a prediction for a specific cell. When using the constructed tree for prediction, this will be the leaf node that the query point is assigned to.

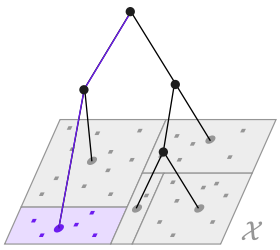

Figure 3.2.: A decision tree partitions the data space. For a query example, the corresponding leaf node is determined by traversing the tree downwards from the root node and applying the learned decision criteria.

Like any learner, the quality of a decision tree q is measured with respect to a loss function ℓ evaluated over some test set $\{(x_i, y_i)\}_1^n$. We are ultimately interested in

finding a tree that minimises this loss, as an approximate to the generalisation error (see Section 1.4).

$$\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, q(x_i))$$

Proposition 3.1.1 ⋆ *The choice of loss function, impurity measure and leaf combiner are tightly related and chosing one of these will determine the other choices.*

The basic idea is that an impurity measure is in fact nothing else than the variance around a centroid. If $q(f)$ is chosen to be a centroid with respect to a loss ℓ, then an impurity measure is exactly the loss of the points in a leaf.

In decision trees, the tree prediction q is constant over a leaf f: If $q(f)$ is the output for leaf f then $\forall x \in f : q(x) = q(f)$. Thus, we can write the loss as a sum over the losses of the individual leaves f.

$$\frac{1}{n}\sum_{i\in D}\ell(y_i, q(x_i)) = \sum_{f\in q}\frac{n_f}{n}L(f) \quad \text{for} \quad L(f) =_{\text{def}} \frac{1}{n_f}\sum_{i\in f}\ell(y_i, q(f))$$

Given a loss function ℓ and assuming the tree structure (and thus the leaves f) are fixed, how should we define the leaf combiner $q(f)$? From the equation above, we can already see that $L(f)$ is minimised if $q(f) = \arg\min_z \sum_{i\in f}\ell(y_i, z)$. That is, $q(f)$ is a centroid with respect to ℓ.

Squared Error and Variance reduction A commonly used impurity measure for regression is the squared-error variance.

$$H_{\text{var}}(f) = \frac{1}{n_f}\sum_{i\in f}(y_i - \bar{y})^2 \quad \text{for } \bar{y} =_{\text{def}} \frac{1}{n_f}\sum_{i\in f} y_i$$

$\bar{y} = \arg\min_z \sum_{i\in f}(y_i - z)^2$ is a centroid with respect to the squared-error loss. $q(f) = \bar{y}$ implies $H_{\text{var}}(f) = L(f)$. In other words, splitting according to H_{var} minimises the squared-error loss of the tree if the leaf combiner is the arithmetic mean.

0/1-loss and majority vote The majority vote is a centroid with respect to the 0/1-loss. The implied impurity measure is the error rate.

$$H_{0/1}(f) = \frac{1}{n_f}\sum_{i\in f}\ell_{0/1}(y_i, q(f))$$

Entropy and Information Gain We can further express the loss of a leaf $L(f)$ for each of the k classes separately. Further, let us assume the leaf combiner provides a probability distribution over the classes k (this is without loss of generality since predicting a single class can be seen as a one-hot distribution). Let $q(f)_k$ be the k-th

entry of the distribution $q(f)$. Then

$$\begin{aligned} L(f) &= \frac{1}{n_f} \sum_{i \in f} \ell(y_i, q(f)) \\ &= \frac{1}{n_f} \sum_{i \in f} \sum_k \mathbb{1}[y_i = k] \; \ell(k, q(f)_k) \\ &= \sum_k \sum_{i \in f} \frac{1}{n_f} \mathbb{1}[y_i = k] \; \ell(k, q(f)_k) \\ &= \sum_k p_{fk} \, \ell(k, q(f)_k) \end{aligned} \tag{3.1}$$

where p_f is the class distribution in leaf f. Consider the negative entropy impurity measure

$$H_{\text{entr}}(f) = \sum_k p_{fk} \log(p_{fk})$$

If the leaf combiner is the distribution of classes in f, i.e. $q(f) = p_f = [p_{f_1}, \ldots, p_{fK}]^\top$ then H_{entr} maximises $\ell(k, q(x_i)) = -\log q(x_i)_k = -\log p_{fk}$. Rewriting this over all examples, this is the *log-loss*, also known as *cross-entropy* loss.

Gini impurity To measure the purity of class labels in a cell, one may consider the probability of drawing two different outcomes from the examples in the current cell. Let p_f be the probability distribution of classes in leaf f. Let p_{fk} be the probability of drawing an example of class k from leaf f. The probability of drawing one example of class k and one of a different class is $p_{fk}(1 - p_{fk})$. The probability of drawing two examples of *any* two different classes then is the *Gini impurity*

$$H_{\text{Gini}}(f) =_{\text{def}} \sum_k p_{fk}(1 - p_{fk}) = \sum_k p_{fk} - \sum_k p_{fk}^2 = 1 - \sum_k p_{fk}^2$$

Analogous to the analysis for H_{entr}, let the leaf combiner be the class distribution in f, that is $q(f) =_{\text{def}} p_f$. From Equation 3.1, we can see that H_{Gini} minimises $\ell(y_i, q(x_i)) = 1 - q(x_i)_{y_i}$, i.e. it maximises the probabilty of the target class. It remains to be seen that $q(f) = p_f$ is in fact a centroid with respect to ℓ. The centroid is

$$\arg\min_z \frac{1}{n_f} \sum_{i \in f} (1 - z_{y_i}) \equiv \arg\max_z \frac{1}{n} \sum_{i \in f} z_{y_i}$$

Let e_k be the vector that contains 1 at position k and 0 otherwise. Then

$$z_{y_i} = \langle z, e_{y_i} \rangle$$

Further, $\frac{1}{n_f} \sum_i e_{y_i}$ is exactly the vector of class frequencies p_f. Continuing, we have

$$\begin{aligned} \arg\max_z \frac{1}{n} \sum_{i \in f} z_{y_i} &= \arg\max_z \langle z, \frac{1}{n} \sum_i e_{y_i} \rangle \\ &= \arg\max_z \langle z, p_f \rangle \end{aligned}$$

which is maximised if and only if $z = p_f$.

3.2. The Random Forest scheme

The deeper a decision tree is grown, the closer the resultig partition cells will fit the training data. This approximation is in fact guided *only* by the training data. In the extreme case, if the tree is fully grown, each partition cell will correspond to a single example and the outcome of that cell will be the outcome of that example. The tree essentially turns into a 1-nearest-neighbour scheme with respect to the training dataset. This means that trees constructed with different samples D of training datasets from the original distribution $P(X, Y)$ potentially predict quite different outcomes for testing examples. This is captured in the concept of learner variance as defined in 1.5.

Mitigating this strong dependence on the training data is one of the main motivations of Random Forests. The basic idea is as follows: If we produce several uncorrelated decision trees and average their predictions, then the predictions should exhibit lower variance In standard forests, this is achieved by introducing randomness into the decision tree construction algorithm by two mechanisms [15, 22]:

- *Bootstrapping*: Each tree is constructed not on the entire training dataset but a random subset of it. Usually, this *bootstrap sample* is produced by drawing the same amount of points with replacement.
- *Random feature selection*: When determining where to split a node, not all features are considered but only a random subset of a given size.

The predictions of individual trees are aggregated to form the forest prediction. Unlike single decision trees, trees in Random Forests are usually grown until each leaf is perfectly pure. Besides their theoretical appeal, Random Forests have several practical advantages.

- The construction of individual trees is easily parallelisable.
- The trained forest model is relatively small: Only decision rules and leaf outputs need to be stored.
- Inference is comparatively fast.
- A Random Forest model naturally provides a measure of feature importance [26].
- Random Forests models naturally support classification tasks with more than two target classes.
- They are well-suited for problems with extremely high feature dimensionality [2].
- There are few hyperparameters to pick and common choices have been shown to work robustly across a wide range of tasks.
- They are not prone to overfitting (⇝Section 5.2).

3.2.1. Bagging

A vital ingredient to Random Forests is the *Bagging* procedure, which stands for *bootstrapping and aggregating*. Bagging is an ensemble learning technique not specific to Random Forests. In Bagging, each member is constructed not on the full training dataset but a *bootstrap sample* of it. The bootstrap sample is usually determined by drawing n out of n examples uniformly with replacement [22, 28]. We will refer to this as *uniform bootstrapping*. One can also determine the bootstrap sample by drawing n out of n points with replacement according to a probability distribution $\{p_1, ..., p_n\}$, which we call *weighted bootstrapping*. Bootstrapping means that each member is trained on a different dataset.

In uniform bootstrapping, if we draw n samples from n available points, the probability of an example being selected in a single draw is $\frac{1}{n}$. Conversely, the probability of an example not being selected in a single draw is $1 - \frac{1}{n}$. We draw n times. Hence, the probability of an observation not being selected in any of the draws is $(1 - \frac{1}{n})^n$. The probability of an example indeed being selected in at least one of the draws then is $1 - (1 - \frac{1}{n})^n$. For large n, one can approximate $\lim_{n\to\infty} 1 - (1 - \frac{1}{n})^n = 1 - e^{-1} \approx 0.632$ [27].

3.2.2. Tree and Forest Partitions

The generalisation error, and consequently individual terms of any decomposition of it have been defined point-wise. That is, they are measured by an expectation over possible realisations of example-outcome pairs (X, Y). They further depend on a random variable D representing the training input to the learner. In order to estimate such an expectation for a given model (a realisation of D), one has to sample realisations of (X, Y). In practise, these example-outcome pairs typically come from a validation dataset that was withheld from training. We have seen in Chapter 4 that for some losses, diversity can be expressed independently of the outcome variable Y. To approximate an actual value, we would still need to sample realisations of X.

Decision Trees, particularly if grown deeply, can be considered to *approximate* the training data, i.e. they are a lossy representation of the training data. A grown decision tree model contains two kinds of parameters, both derived from the training data D.

- The tree structure, i.e. the decision boundaries. These are used for determining the leaf node for a query example.
- The output value of a leaf node. This is the predicted value for a query example falling into that leaf. The leaf predictions depend on the decision boundaries but are not solely determined by them.

This leads to the question whether characteristics of a Random Forest model could be expressed solely in terms of its tree parameters, and not in terms of predictions on query points.

Each level of a decision tree induces a partition of the space of examples $\mathcal{X}$. Because each example is associated with an outcome, we can also think of it as a partition of $(\mathcal{X}, \mathcal{Y})$. We call such a partition a *tree partition* and a cell a *tree cell*. Decision trees produce predictions via an aggregate of the queried leaf node's outcomes. Thus, the predictions of a decision tree over a single cell are constant. An ensemble of trees also induces a partition: the partition obtained by intersecting all tree partitions. We call these *forest cells*. Formally, if $T_1, \ldots, T_M$ are tree partitions, the forest partition is given as

$$\{c_1 \cap \cdots \cap c_M \mid c_1 \in T_1, \ldots, c_M \in T_M\}$$

Each forest cell is associated with M tree cells whose intersection constitutes it. For any query point that falls within a certain tree cell, the forest prediction is given by an aggregate over the associated tree cells. Thus, the predictions of a random forest are constant over a single forest cell. This means that also a loss, as well as any decomposition constituents of the loss are constant over forest cells.

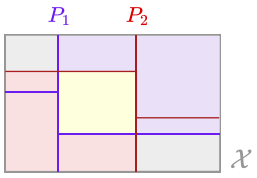

Figure 3.3.: The ● data space $\mathcal{X}$ (gray) and partitions ● P_1 and ● P_2 of it induced by two decision trees. The ● intersections of any cell of P_1 and any cell of P_2 form the *forest partition*. One such intersection is highlighted in yellow.

Exploiting that partitions are cell-wise constant, we can apply a special case of the law of total expectation to express it in terms of cells. [1] Consider a forest partition $Z = Z_1 \dot\cup \ldots \dot\cup Z_P$ of $Z = (X, Y)$.

Proposition 3.2.1 ⋆ *The generalisation error of a random forest model can be expressed in terms of intersections of tree cells.*

$$\mathbb{E}_{Z,D}\left[\ell(y, \bar{q})\right] = \mathbb{E}_{Y,D}\left[\sum_{p=1}^{P} \mathbb{P}\left[Z_p\right] \cdot \mathbb{E}_Z\left[\ell(Y, \bar{q}) \mid Z_p\right]\right]$$

A similar statement can be made for the simpler case of a single decision tree and is used in Section 3.1 to derive impurity measures.

1: Let $X_1 \dot\cup \ldots \dot\cup X_M$ be a disjoint, countable partition of the sample space of X. Then

$$\mathbb{E}_X[X] = \sum_{i=1}^{M} \mathbb{E}_X[X \mid X_i] \cdot \mathbb{P}[X_i]$$

While we have eliminated the dependence on a query point X, the quantity $\mathbb{E}_Z \left[\ell(y, \bar{q}) \mid Z_p \right]$ still depends on a realisation of an outcome Y. In Section 5.2 we see that we can decompose this term further into components that are dependent and independent of Y, respectively.

4. Diversity

It is evident that the interactions between members are a driving force behind member performance (see, for example, Section 2.3). Understanding ensembles means understanding how individual member models can be related to each other and how these relationships affect the ensemble performance. In this section, we will give an overview over some of the measures proposed to quantify ensemble diversity. We focus on the classification task.

4.1. Measures of Diversity

For a suitable diversity measure, the following properties are desirable.

- The measure captures the differences between member models.
- There is a relationship between the diversity measure and the ensemble generalisation error.
- The diversity measure is independent of the outcome variable.

Naturally, many diversity measures are based on some notion of spread of the member models. This can, for instance, be a measure of disagreement, variance, impurity, entropy, or covariance.

4.1.1. Disagreement

We will begin with two simple measures based on the contingency table [20]. To shorten notation, we will refer to its entries as given in table 4.1. The *Disagreement Measure* is the proportion of examples on which two members make different predictions.

	$q_i = +1$	$q_i = -1$
$q_j = +1$	$n_{(+,+)}$	$n_{(+,-)}$
$q_j = -1$	$n_{(-,+)}$	$n_{(-,-)}$

Table 4.1.: Notation for entries of the contingency table.

$$\frac{1}{n}(n_{(+,-)} + n_{(-,+)})$$

The Q-*Statistic* is given as

$$Q_{ij} =_{\text{def}} \frac{n_{(+,+)}n_{(-,-)} - n_{(-,+)}n_{(+,-)}}{n_{(+,+)}n_{(-,-)} + n_{(-,+)}n_{(+,-)}}$$

$Q_{ij} = 0$ if q_i and q_j are independent, positive if the members make similar predictions and negative if the members make different predictions.

4.1.2. Ambiguity

Krogh and Vedelsby [3] propose a decomposition of the ensemble generalisation error into two terms: One describing the average error of ensemble members and a so-called *ambiguity* term.

$$\mathbb{E}\left[\ell(y, \bar{q})\right] = \frac{1}{M}\sum_{i=1}^{M}\ell(y, q_i) - \frac{1}{M}\sum_{i=1}^{M}\ell(\bar{q}, q_i)$$

This perfectly divides the ensemble error into error due to characteristics of the individual member models and error due to interactions between member predictions. Note that the contribution of the ambiguity term here is negative, that is, ambiguity is

a beneficial influence. This decomposition was originally given for the squared-error loss [3] and later proven for the KL-divergence [16]. We will treat this quantity in detail in the rest of this chapter. Right now, note that the ambiguity term can be interpreted as a measure of *variance* where individual distances are measured by the loss ℓ. For the squared-error loss and the arithmetic mean combiner, this is exactly the "statistical" variance over the members $\frac{1}{M}\sum_{i=1}^{M}\left(q_i - \frac{1}{M}\sum_{i=1}^{M} q_i\right)^2$. For other loss functions, this yields other well-known quantities (see definition 1.6.3).

4.1.3. Impurity

Kohavi and Wolpert [29] give a bias-variance decomposition of the 0/1-loss as follows.

$$\mathbb{E}\left[\ell_{0/1}\left(Y, q(X)\right)\right] = \mathbb{E}\left[1/2\left(\sigma_x^2 + \text{bias}(q) + \text{var}(q)\right)\right]$$

$$\begin{aligned}
\text{for bias}(q) &= \sum_y \left(\mathbb{P}\left[y^\star = y \mid x\right] - \mathbb{P}\left[q = y \mid x\right]\right)^2 \\
\text{var}(q) &= 1 - \sum_y \mathbb{P}\left[q = y \mid x\right]^2 \\
\sigma^2 &= 1 - \sum_y \mathbb{P}\left[y^\star = y \mid x\right]
\end{aligned}$$

Note that the variance and noise terms are of the form of the Gini impurity (Proposition 3.1.1) and can be interpreted as measures of impurity. Kuncheva [5] takes inspiration from this variance term and proceed as follows. Instead of considering the impurity over positive or negative labels, they instead consider the impurity over labels for which the ensemble is correct ($\tilde{y} = +1$) or incorrect ($\tilde{y} = -1$), respectively. As such, the probability is not with respect to the distribution of labels but the distribution Θ of members in the ensemble. This yields

$$\text{var}'_x =_{\text{def}} 1 - \sum_{\tilde{y}} \mathbb{P}_\Theta\left[q = \tilde{y} \mid x\right]^2$$

Averaging this over the entire training dataset, this can be understood as a measure of diversity. More precisely, it is the impurity as measured by the Gini impurity of the predictions of the ensemble members.

4.1.4. Entropy

Cunningham and Carney [30] propose the entropy between member predictions as a diversity measure. For a single point, it is given as

$$\sum_{y \in \{-1,+1\}} -\mathbb{P}_\Theta\left[y \mid x\right] \log \mathbb{P}_\Theta\left[y \mid x\right]$$

where, for an ensemble, $\mathbb{P}_\Theta\left[y \mid x\right] \approx \frac{1}{M}\sum_{i=1}^{M} \mathbb{1}\left[q_i(x) = y\right]$. Shipp and Kuncheva [31] propose a target-dependent measure that is reminiscent of the entropy. With $m_+(x) = \sum_{j=1}^{M} \mathbb{1}\left[q_i(x) = y(x)\right]$:

$$\frac{1}{M - \frac{M}{2}} \min\{m_+(x), M - m_+(x)\}$$

4.1.5. Covariance

In Section 2.3, we have already seen evidence that the covariance between members is an essential factor to ensemble performance. Indeed, the notion of covariance and uncorrelatedness has been a guiding thought in the literature [11, 28, 32]. We will now introduce an exact decomposition of the ensemble error for the squared-error loss that includes the average covariance between member predictions.

Theorem 4.1.1 *(Bias-Variance-Covariance decomposition [4, 32]) It holds that*

$$\mathbb{E}_{(X,Y),D}\left[(y-\bar{q})^2\right] = \overline{bias}^2 + \frac{1}{M}\overline{var} + \left(1-\frac{1}{M}\right)\overline{covar}$$

$$\begin{aligned}
\text{for } \overline{bias} &=_{def} \frac{1}{M}\sum_{i=1}^{M}(\mathbb{E}_D\left[q_i\right]-y)\\
\overline{var} &=_{def} \frac{1}{M}\sum_{i=1}^{M}\mathbb{E}_D\left[(q_i-\mathbb{E}_D\left[q_i\right])^2\right]\\
\overline{covar} &=_{def} \frac{1}{M(M-1)}\sum_{i\neq j}\mathbb{E}_D\left[(q_i-\mathbb{E}_D\left[q_i\right])(q_j-\mathbb{E}_D\left[q_j\right])\right]
\end{aligned}$$

Proof. We begin by applying the bias-variance decomposition (theorem 1.5.1) to the ensemble model $\bar{q}$.

$$\mathbb{E}\left[\ell(y,\bar{q})\right] = \mathbb{E}\left[\ell(Y,y^\star)\right] + \mathbb{E}\left[\ell(\bar{q}^\star,y^\star)\right] + \mathbb{E}\left[\ell(\bar{q}^\star,\bar{q})\right]$$

For the bias term, it holds that [4]

$$\text{bias}(\bar{q}) = \ell(\bar{q}^\star,y^\star) = \overline{\text{bias}}^2$$

The variance of the ensemble can be decomposed into terms describing variances of individual members and covariances between members.

Using that

$$\left(\sum_{i=1}^{n}a_i\right)^2 = \sum_{i=1}^{n}a_i + \sum_{j\neq i}a_ia_j$$

$$\begin{aligned}
\text{Var}\,(\bar{q}) &= \mathbb{E}_D\left[\ell(\bar{q},\bar{q}^\star)\right] = \mathbb{E}_D\left[(\bar{q}-\mathbb{E}_D\left[\bar{q}\right])^2\right]\\
&= \mathbb{E}_D\left[\left(\frac{1}{M}\sum_{i=1}^{M}q_i - \mathbb{E}_D\left[\frac{1}{M}\sum_{i=1}^{M}q_i\right]\right)^2\right]\\
&= \frac{1}{M^2}\sum_{i=1}^{M}\mathbb{E}_D\left[(q_i-\mathbb{E}_D\left[q_i\right])^2\right] + \frac{1}{M^2}\sum_{j\neq i}\mathbb{E}_D\left[(q_i-\mathbb{E}_D\left[q_i\right])(q_j-\mathbb{E}_D\left[q_j\right])\right]
\end{aligned}$$

Rearraging the coefficients yields the form of the theorem. □

$$\frac{1}{M^2} = \left(1-\frac{1}{M}\right)\frac{1}{M(M-1)}$$

The Bias-Variance-Covariance decomposition can be interpreted as a decomposition into characteristics of individual learners (mean bias and variance), plus a quantity describing the interactions between learners. Again, one can see that ensemble performance profits if members are uncorrelated. However, this decomposition is only given for the squared error and the arithmetic mean combiner.

4.1.6. Relationship between Ambiguity and Covariance

We have now seen two approaches to expressing the ensemble generalisation error:

- In terms of *covariance* as in the bias-variance-covariance decomposition of theorem 4.1.1. A very similar notion also appears in when considering ensemble margins as in 2.3.5
- In terms of variation around a centroid with respect to some loss function as in the diversity decomposition of Theorem 4.2.2. [1]

1: For any loss function:

$$\mathbb{E}_D\left[\mathrm{LE}(q^\star, q)\right]$$

For the squared-error loss:

$$\mathbb{E}_D\left[(q - \mathbb{E}_D[q])^2\right] = \mathbb{E}_D\left[\ell(q^\star, q)\right]$$

For Bregman divergences, this is analogous to the *Bregman information*, see definition 1.6.3:

$$\mathbb{E}_D\left[B_\phi\left(q^\star, q\right)\right]$$

Inspecting the proofs of theorems 4.1.1 and 2.3.5, one can observe that in both cases, one begins with a measure of variance. Based on this, a covariance expression is then extracted, enabled by the definition of variance involving a square, producing cross-terms. In other words, *the covariance term is an artifact of the squared-error loss*. The notion of variance is more general.

For the case of squared-error loss, Brown et al. [32] relate the ambiguity decomposition (⇝4.2.1) and the bias-variance-covariance decomposition (⇝4.1.1) directly.

$$\mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M}(y - q_i)^2 - \frac{1}{M}\sum_{i=1}^{M}(\bar{q} - q_i)^2\right] = \overline{\mathrm{bias}} + \frac{1}{M}\overline{\mathrm{var}} + \left(1 - \frac{1}{M}\right)\overline{\mathrm{covar}}$$

which yields

$$\begin{aligned}\mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M}(y - q_i)^2\right] &= \overline{\mathrm{bias}} + \Omega \\ &= \Omega - \left(\frac{1}{M}\overline{var} + \left(1 - \frac{1}{M}\right)\overline{\mathrm{covar}}\right) \\ \text{for } \Omega =_{\mathrm{def}} \overline{\mathrm{var}} + \frac{1}{M}\sum_{i=1}^{M}&\left(\mathbb{E}[q_i] - \mathbb{E}[\bar{q}]\right)^2\end{aligned}$$

The first quantity describes the average member error, the second the diversity of the ensemble. The appearance of Ω in both terms illustrates that there is a trade-off between individual member error and diversity. In other words, one cannot maximise diversity without also affecting other parts of the ensemble error. Vice versa, optimising for other components of the error will also affect diversity.

4.2. The Diversity-Effect Decomposition

Generalising Ambiguity In Section 1.5, we have derived a generalised bias-variance decomposition by considering the effect on the loss of using a random variable instead of its non-random centroid. For the variance, this would be the expected difference in loss between using a model dependent on the training input D and the expected model, where the expectation is over D. Consider now an ensemble of learners, in which each learner is constructed according to a random parameter Θ. Similar to bias and variance, we may consider the distribution of models over Θ around a central model $\bar{q}$ that is non-random with respect to Θ. The expected *effect* of using the central model instead of some single member is expressed as follows.

$$\mathbb{E}_\Theta\left[\mathrm{LE}(\bar{q}, q_\Theta)\right] = \mathbb{E}_\Theta\left[\ell(Y, q_\Theta) - \ell(Y, \bar{q}))\right] \approx \frac{1}{M}\sum_{i=1}^{M}\ell(Y, q_i) - \ell(Y, \bar{q})$$

Using the same strategy as for the bias-variance-effect decomposition of Theorem 1.5.1, we can use this to formulate a decomposition of the generalisation error with respect to the central model.

Theorem 4.2.1 *(Ambiguity-Effect decomposition [6]) For any loss function ℓ, target label Y, ensemble members $q_1, \dots, q_M$ with combiner $\bar{q}$*

$$\ell(Y, \bar{q}) = \frac{1}{M}\sum_{i=1}^{M} \ell(Y, q_i) - \underbrace{\left(\frac{1}{M}\sum_{i=1}^{M} \ell(Y, q_i) - \ell(Y, \bar{q})\right)}_{\textit{Ambiguity-Effect / Ensemble Improvement}}$$
$$= \frac{1}{M}\sum_{i=1}^{M} \ell(Y, q_i) - \frac{1}{M}\sum_{i=1}^{M} LE(\bar{q}, q_i)$$

If the ensemble combiner $\bar{q}$ is chosen to be the central model (↝2.2.2), this is a decomposition of the ensemble generalisation error and a generalisation of the ambiguity decomposition described in Section 4.2.

We can then also find an intuitive interpretation of ambiguity-effect: It is the effect of ensembling on the error. If we consider the members to be constructed according to a parameter Θ, a reasonable measure of the member performance is its loss in expectation over the parameter distribution: $\mathbb{E}_\Theta[\ell(Y, q(X;\Theta))] \approx \frac{1}{M}\sum_{i=1}^{M} \ell(Y, q(X;\Theta_i)$. What we gain or lose from using an ensemble $\bar{q}$ over just a single member model is exactly measured by ambiguity-effect. Due to this, this quantity is also known as *ensemble improvement* [7, 33].

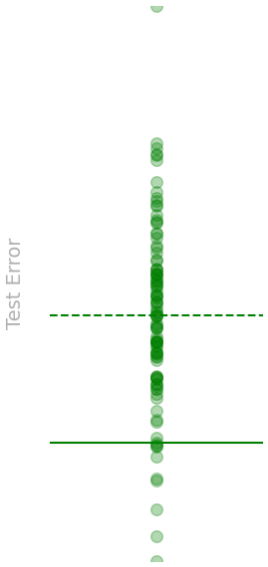

Figure 4.1.: The spread of individual tree predictions in a random forest ensemble. Glyphs correspond to test errors of individual trees. The dashed line is the average test error of individual trees $\frac{1}{M}\sum_1^M \ell(y, q_i)$. The solid line is the test error of the ensemble $\ell(y, \bar{q})$. The difference between these values is the *ensemble improvement* or *ambiguity-effect*.

Diversity-Effect The ambiguity decomposition divides the ensemble error into the average member error and the variance among members. How does this relate to the well-known bias-variance decomposition? Indeed, nothing prevents us from applying the bias-variance decomposition of Theorem 1.5.1 to the average member error, resulting in a decomposition into *average* bias, *average* variance and expected ambiguity, which is also referred to as *diversity* [6]. In summary, the decomposition is given in the following theorem. Note that it holds for *any* loss function.

Theorem 4.2.2 *(Bias-Variance-Diversity-Effect decomposition [6])*

$$\mathbb{E}[\ell(y, \bar{q})] = \overbrace{\mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M} \ell(y, q_i)\right]}^{\textit{avg. member error}} - \overbrace{\mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M} LE(\bar{q}, q_i)\right]}^{\textit{ambiguity-effect}}$$
$$= \underbrace{\mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M} LE(q_i^\star, y^\star)\right]}_{\textit{avg. bias-effect}} + \underbrace{\mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M} LE(q_i^\star, q_i)\right]}_{\textit{avg. variance-effect}} - \underbrace{\mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M} LE(\bar{q}, q_i)\right]}_{\textit{diversity-effect}}$$

Similar to variance, ambiguity and ambiguity-effect are measures of spread. Variance measures the spread of training error across models trained with different draws of the training dataset D around a model that is a centroid with respect to the distribution of D. Similarly, ambiguity measures the spread of individual member model errors around a model that is centroid with respect to the distribution of Θ, namely the combiner $\bar{q}$. In case of squared loss, this is indeed the statistical variance around the arithmetic mean. For other losses, this is a different quantity.

should not be $y^\star$ there

Unchanged bias and reduction in variance In Section 2.3 we gave arguments for how in specific cases, the ensemble bias equals the bias of any member. We now have the tools to show this in a more general manner [6]. For the ensemble bias, application

of the ambiguity-effect decomposition (↝4.2.1) to a set of centroid models $q_i^\star$ yields:

$$\underbrace{\text{LE}(y, \bar{q}^\star)}_{\text{ens. bias}} = \underbrace{\frac{1}{M}\sum_{i=1}^{M} \text{LE}(y, q_i^\star)}_{\text{avg. bias}} - \underbrace{\frac{1}{M}\sum_{i=1}^{M} \text{LE}(\bar{q}^\star, q_i^\star)}_{\Delta}$$

For the ensemble variance, application of the diversity-effect decomposition (4.2.2) while substituting $y \leftarrow \bar{q}^\star$:

$$\underbrace{\mathbb{E}_D\left[\text{LE}(\bar{q}^\star, \bar{q})\right]}_{\text{ens. var.}} = \underbrace{\frac{1}{M}\sum_{i=1}^{M} \text{LE}(\bar{q}^\star, q_i^\star)}_{\Delta} + \underbrace{\frac{1}{M}\sum_{i=1}^{M} \mathbb{E}_D\left[\text{LE}(q_i^\star, q_i)\right]}_{\text{avg. var.}} - \underbrace{\mathbb{E}_D\left[\frac{1}{M}\sum_{i=1}^{M} \text{LE}(\bar{q}, q_i)\right]}_{\text{diversity}}$$

Due to Lemma 2.2.1 which states that in homogeneous ensembles $q_i^\star = q_j^\star = \bar{q}^\star$, we can conclude that $\Delta = 0$.

Corollary 4.2.3 *For homogeneous ensembles, we can conclude the following:*

- *The ensemble bias is equal to the average member bias:*

$$\Delta = 0 \quad \rightarrow \quad LE(y, \bar{q}^\star) = \frac{1}{M}\sum_{i=1}^{M} LE(y, q_i^\star)$$

- *Diversity is a component of ensemble variance. The other component is the average member variance. In other words, ensemble variance reduction is measured exactly by diversity.*
- *Diversity is bounded from above by the average member variance.*

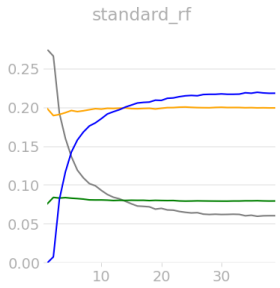

Figure 4.2.: Components of the diversity-effect decomposition by number of trees in a standard Random Forest ensemble trained on *mnist-subset*. ● Average bias and ● average variance stay (almost) constant with increasing number of trees, while ● diversity increases. The ensemble error is the difference between the average member error (which is the sum of average bias and average variance) and diversity.

this shows it only for Bregman divergences. Not convinced this also holds for 0/1-loss, although wood23 claim so (maybe just imprecise writing)

4.3. Diversity for Bregman Divergences

Note that bias-effect, variance-effect and ambiguity-effect are all of the form $\mathbb{E}\left[\text{LE}(\circ, \square)\right]$ and depend directly on the target label Y. With variance-effect, we capture the effect of variations between different training datasets on the prediction error. With ambiguity-effect, we have captured the effect of variations between the different member models on the prediction error. We have already seen that for the squared error, the variance-effect coincides with familiar notion of "statistical" variance between the predictions. We will now argue that for Bregman divergences (↝1.6.1), the loss-effect terms reduce to the loss between the two objects. The class of Bregman divergences covers many widely used loss functions and thus allows us to formulate a unified bias-variance-decomposition.

Lemma 4.3.1 *([13], Theorem 0.1b) Let q be a function of a random variable Z and independent of Y. For $y^\star = \mathbb{E}_Y[Y]$, i.e. the right Bregman centroid w.r.t. Y, it holds that*

$$\mathbb{E}_Z\left[B_\phi\left(y^\star, q\right)\right] = \mathbb{E}_{Z,Y}\left[B_\phi\left(Y, q\right) - B_\phi\left(Y, y^\star\right)\right]$$

Using $q \leftarrow q^\star$ shows that bias-effect collapses to bias for Bregman divergences:

$$B_\phi\left(y^\star, q^\star\right) = \mathbb{E}_Y\left[B_\phi\left(Y, q^\star\right) - B_\phi\left(Y, y^\star\right)\right]$$

Divergence	Combiner	Name
Squared loss	$\frac{1}{M}\sum_{i=1}^{M} q_i$	Arithmetic mean
Poisson regression loss	$\prod_{i=1}^{M} q_i^{\frac{1}{M}}$	Geometric mean
KL-divergence	$Z^{-1}\prod_{i=1}^{M}\left(\mathbf{q}_i^{(c)}\right)^{\frac{1}{M}}$	Normalised geometric mean
Itakura-Saito loss	$1/\left(\frac{1}{M}\sum_{i=1}^{M}\frac{1}{q_i}\right)$	Harmonic mean

Figure 4.3.: Examples for the combiner $\mathscr{C}_\Theta[q_\Theta]$ implied by a Bregman divergence [6].

Lemma 4.3.2 ⋆ *(Generalised from [6]) Let q be a function of random variable Z and independent of Y. For $q^\star = \mathscr{C}_Z[q]$, i.e. the left Bregman centroid w.r.t. Z, it holds that*

$$\mathbb{E}_Z\left[B_\phi\left(q^\star, q\right)\right] = \mathbb{E}_{Z,Y}\left[B_\phi\left(Y, q\right) - B_\phi\left(Y, q^\star\right)\right]$$

Proof. The proof is given in Appendix B.0.1. □

Lemma 4.3.2 shows that variance- and ambiguity-effect collapse to variance and ambiguity. The variance (in the bias-variance sense) is the variance of the estimates q_D dependent on D with respect to different realisations of D around its D-centroid.

$$\text{For } q^\star = \mathscr{C}_D[q_D]\text{:}\quad B_\phi\left(q^\star, q\right) = \mathbb{E}\left[B_\phi\left(Y, q\right) - B_\phi\left(Y, q^\star\right)\right]$$

Ambiguity/Diversity is the variance of the estimates q_Θ dependent on Θ with respect to different realisations of Θ around its centroid.

$$\text{For } \bar{q} = \mathscr{C}_\Theta[q_\Theta]\text{:}\quad B_\phi\left(\bar{q}, q\right) = \mathbb{E}_Y\left[B_\phi\left(Y, q\right) - B_\phi\left(Y, \bar{q}\right)\right]$$

This yields a generalised bias-variance-diversity decomposition for Bregman divergences as a special case of the corresponding effect decomposition.

Theorem 4.3.3 *(Bias-Variance-Diversity decomposition for Bregman divergences)*

$$\begin{aligned}\mathbb{E}_{(X,Y),D}\left[B_\phi\left(Y, q\right)\right] = &\underbrace{\mathbb{E}_{Y|X}\left[B_\phi\left(Y, y^\star\right)\right]}_{\textit{noise}} \\ &+ \underbrace{\frac{1}{M}\sum_{i=1}^{M} B_\phi\left(y^\star, q_i^\star\right)}_{\overline{\textit{bias}}} \\ &+ \underbrace{\frac{1}{M}\sum_{i=1}^{M}\mathbb{E}_D\left[B_\phi\left(q_i^\star, q_i\right)\right]}_{\overline{\textit{variance}}} \\ &- \underbrace{\mathbb{E}_D\left[\frac{1}{M}\sum_{i=1}^{M} B_\phi\left(\bar{q}, q_i\right)\right]}_{\textit{diversity}}\end{aligned}$$

Ensemble improvement for Bregman Divergences In section 2.3, we have used Jensen's inequality to show that the ensemble improvement is non-negative for some

cases.

$$\mathbb{E}_\Theta\left[\ell(Y, q_\Theta(X))\right] - \ell(Y, \mathbb{E}_\Theta\left[q_\Theta(X)\right]) \geq 0$$

It is evident that the Jensen gap is but a special case of ambiguity-effect (↝4.2.1) for $\bar{q} =_{\text{def}} \frac{1}{M}\sum_{i=1}^{M} q_i$ and convex loss functions. This shows that the ensemble loss is always smaller-equal than the expected member loss, but *only* if the ensemble output is actually produced by an arithmetic mean.

However, it can not be assumed from the outset that the arithmetic mean is the best ensemble combiner. Indeed, for the cross-entropy loss, Abe et al. [25] proceed to note that the Jensen gap corresponds to a form that is "not immediately recognizable". Although they do find an interpretation of it, it is still necessarily dependent on the outcome Y . It seems reasonable to define the ensemble combiner in accordance to the Bregman divergence, i.e. to be the *dual* expectation $\mathscr{C}_\Theta\left[q_\Theta\right]$. Non-negativity is then easily shown since in that case ambiguity-effect reduces to ambiguity.

In fact, for the case of cross-entropy, [6] show that the ambiguity term is still nonnegative, i.e. that the arithmetic mean combiner does not hurt performance.

$$B_\phi\left(\bar{q}, q\right) = \mathbb{E}\left[B_\phi\left(Y, q\right) - B_\phi\left(Y, \bar{q}\right)\right] \quad \text{for } \bar{q} = \mathscr{C}_\Theta\left[q_\Theta\right]$$

and the value of any Bregman divergence is always non-negative. Further, ambiguity is now independent of the outcome Y.

Corollary 4.3.4

- *For any Bregman divergence, ensembling using the combiner implied by the divergence can not hurt performance.*
- *The diversity is an intuitive measure of ensemble improvement.*
- *This ensemble improvement (diversity) appears in an exact decomposition of the ensemble generalisation error.*

4.4. Diversity of the 0/1-loss

While for Bregman divergences ensembling cannot hurt performance, this is not given for the general case. For the 0/1-loss, a way forward is to impose assumptions on the performance of member models. In the remainder of this section, we will review some assumptions that imply ensemble effectivity and show that they are in fact tightly related. We begin by focussing on binary classification problems and then consider non-binary problems.

Definition 4.4.1 *The 0/1-loss between two outcomes Y, Y' is*

$$\ell_{0/1}\left(Y, Y'\right) =_{def} \mathbb{1}\left[Y \neq Y'\right]$$

The ensemble combiner implied by the 0/1-loss is the plurality vote.

The proper term here would actually be *plurality* vote since, for a class to win the vote, it is required to have more than $1/k$ votes. Strictly speakin, a *majority* vote win requires the majority of all votes, i.e. more than $1/2$. For $k = 2$, majority and plurality voting is equivalent.

Definition 4.4.2 *(Majority/Plurality vote combiner) For a k-class classification problem, the majority vote combiner is defined as*

$$\bar{q}(X) = \arg\min_{z \in [k]} \mathbb{E}_\Theta\left[\ell_{0/1}\left(z, q_\Theta\right)\right]$$

This assumes that each member model predicts a single class, although this does not imply that member models must necessarily be trained based on the 0/1-loss, too. If a

member predicts a class, we also say that the member *votes* for that class. The plurality vote combiner is the centroid with respect to the 0/1-loss.

In the remainder of this section, we will analyse classification ensembles as measured by the 0/1-loss. A basic quantity for this will be the ratio of members that are incorrect for a given example-outcome pair.

Definition 4.4.3 *For a distribution of members constructed according to input data $D = (D_1, \ldots, D_M)$ and parameters $\Theta = (\Theta_1, \ldots, \Theta_M)$, the expected ratio of incorrect ("wrong") members is*

$$W(X, Y) =_{def} \mathbb{E}_{D,\Theta}\left[\ell_{0/1}\left(Y, q_{D,\Theta}(X)\right)\right]$$

As with other variables, we sometimes omit explicitly stating the dependence on (X, Y). Further, we write $W_\Theta =_{def} \mathbb{E}_\Theta\left[\ell_{0/1}\left(Y, q_{D,\Theta}\right)\right]$. For the complement, we write $\overline{W} =_{def} 1 - W$

Lemma 4.4.1 *([7]) The average ratio of incorrect members is equal to the error rate of an average member.*

$$\mathbb{E}_{(X,Y)}[W] = \mathbb{E}_{(X,Y)}\left[\mathbb{E}_{D,\Theta}\left[\ell_{0/1}\left(Y, q_{D,\Theta}(X)\right)\right]\right] = \mathbb{E}_{D,\Theta}\left[\mathbb{E}_{(X,Y)}\left[\ell_{0/1}\left(Y, q_{D,\Theta}(X)\right)\right]\right]$$

A simple bound Using Markov's inequality, we can readily upper-bound the error of the ensemble in terms of expected errors of the members [7] [2] .

$$0 \leq \mathbb{E}\left[\ell_{0/1}\left(Y, \bar{q}\right)\right] = \mathbb{P}\left[W \geq \kappa\right] \leq \mathbb{P}\left[W \geq 1/2\right] \leq 2\mathbb{E}\left[W\right]$$

Where $\kappa =_{\text{def}} 1 - t$ if t is the voting threshold. While there exist examples for which this upper bound is tight [7], it is reasonable to suspect that the ensemble being worse by a factor of two is only a pathological case and not relevant for practise.

4.4.1. Diversity in binary classification problems

We begin by considering binary classification problems which have $k = 2$ two possible outcomes. The special property of binary classification problems is that any vote which is not correct is automatically a vote for the single other class. In other words, diversity can be measured in terms of disagreement between members. This is not given for problems with $k > 2$ where an incorrect vote might correspond to any other class. We will see that, in this case, diversity will be measured in terms of differences in *correctness* of members.

The dichotomy of binary classification problems allows us to succinctly express the diversity-effect.

Lemma 4.4.2 *([34]) For a classification problem with $k = 2$ classes, let $y, \bar{q} \in \{-1, 1\}$. It then holds that*

$$\frac{1}{M}\sum_{i=1}^{M}\left[\ell_{0/1}\left(y, q_i\right) - \ell_{0/1}\left(y, \bar{q}\right)\right] = (y \cdot \bar{q})\frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}\left(\bar{q}, q_i\right) \in \{-1, 0, 1\}$$

Proof. Let $y, \bar{q} \in \{-1, 1\}$.

2: Markov's inequality states that for a nonnegative random variable X and $a > 0$

$$\mathbb{P}[X \geq a] \leq \frac{\mathbb{E}[X]}{a}.$$

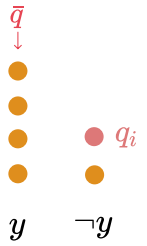

Figure 4.4.: Example of the effect of a member's vote q_i on the diversity on a point for which the ensemble majority vote is correct. Example where q_i has positive contribution to the diversity effect term, i.e. $\ell_{0/1}\left(y, q_i\right) - \ell_{0/1}\left(y, \bar{q}\right) = 1$. The member q_i is incorrect but due to the discreteness of the majority vote combiner, the ensemble performance does not suffer – unless the majority vote is tipped. Any correct vote while the ensemble already is correct is effectively "wasted" and incorrect votes correspond to diversity.

Figure 4.5.: Example where q_i has negative contribution to the diversity effect term, i.e. $\ell_{0/1}\left(y, q_i\right) - \ell_{0/1}\left(y, \bar{q}\right) = -1$. Any further incorrect vote while the ensemble is already incorrect would be wasted. The negative effect here eventually results in the 0/1-loss of 1.

- Assume the ensemble is correct, i.e. $y = \bar{q}$. Then $\ell_{0/1}(y, \bar{q}) = 0$ and the left-hand-side equals $\frac{1}{M}\sum_{i=1}^{M} \ell_{0/1}(y, q_i) = \frac{1}{M}\sum_{i=1}^{M} \ell_{0/1}(\bar{q}, q_i)$. Further, $y \cdot \bar{q} = 1$.
- Assume the ensemble is incorrect, i.e. $y \neq \bar{q}$. Then $y \cdot \bar{q} = -1$ and, for the left-hand-side, we can write

$$\frac{1}{M}\sum_{i=1}^{M}\left[\ell_{0/1}(y, q_i)\right] - 1 = -\left(1 - \frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(y, q_i)\right) = -\left(\frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(\bar{q}, q_i)\right)$$

using that, since $y \neq \bar{q}$, $\left(1 - \frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(y, q_i)\right) = \frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(\bar{q}, q_i)$.

□

An intuition of this is also that of "wasted votes": Under the majority vote combiner, for the ensemble to be correct, we require only at least half of the members to be correct. Any higher ratio of correct ensemble members does not improve the ensemble performance on this point and these can be seen as "wasted". Likewise, the ensemble is incorrect if not more than half of the members are correct. Any positive votes do not influence the ensemble improvement and can be considered "wasted".

We can divide the range of X into two disjoint subsets. Let X_+ be the examples on which the ensemble is correct. Ambiguity on these points has a decreasing effect on the overall ensemble error. Let X_- be the examples on which the ensemble is incorrect. Ambiguity on these points has an increasing effect on the overall ensemble error. This yields a decomposition into *good* and *bad* diversity.

Corollary 4.4.3 *([34]) For a classification problem with $k = 2$ classes, let $y, \bar{q} \in \{-1, 1\}$. It then holds that*

$$\begin{aligned}\mathbb{E}_X\left[\ell_{0/1}(Y, \bar{q})\right] = \mathbb{E}_X&\left[\frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(Y, q_i)\right] \\ &\underbrace{-\mathbb{E}_{X_+}\left[\frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(\bar{q}, q_i)\right]}_{\text{"good" diversity}} \\ &\underbrace{+\mathbb{E}_{X_-}\left[\frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(\bar{q}, q_i)\right]}_{\text{"bad" diversity}}\end{aligned}$$

Note that this is a special case of the ambiguity-effect decomposition of Theorem 4.2.1. There is a tradeoff between average member error and ambuigity/diversity. Here, however, diversity is not always beneficial. On points where the ensemble is incorrect, disagreements have a negative effect on the overall ensemble error. In other words, for majority vote ensembles, diversity is only beneficial *on points at which the ensemble can actually afford to be diverse*.

Further, from Corollary 4.4.3, one can already see that the ensemble improvement (i.e. diversity-effect) in binary classification problems is only non-negative if good diversity outweights bad diversity.

Good and bad diversity can be expressed solely in terms of the ratio of incorrect members.

Lemma 4.4.4 ⋆ *Let y be the true outcome for a given example. Let $\neg y$ be an outcome that is not y. Write $\ell_{0/1}(q_i, \neg y) =_{def} \sum_{k \neq y} \ell_{0/1}(q_i, k)$ for the indication whether q_i is incorrect. Then the following identities hold.*

$$\mathbb{E}_{X_+}\left[\frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(q_i, \bar{q})\right] = \mathbb{E}_{X_+}\left[W_1^M\right]$$

$$\mathbb{E}_{X_-}\left[\frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(q_i, \bar{q})\right] = \mathbb{E}_{X_-}\left[\frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(q_i, \neg y)\right] = \mathbb{E}_{X_-}\left[1 - W_1^M\right]$$

Analogous equalities hold in expectation over member parameter Θ.

this usage of $\neg y$ doesnt quite check out, should be limited to $k = 2$. Arguments about k-competence should still work though.

4.4.2. Ensembles of Weak Learners

The following result holds for an arbitrary number of classes. Note that this is also the assumption enabling Theorem 2.3.5.

Definition 4.4.4 *(Weak learner [7, 10]) A model q_Θ is a weak learner if and only if performs better than randomly guessing.*

$$\mathbb{E}_{(X,Y)}\left[\ell_{0/1}(Y, q_\Theta)\right] \geq 1/2$$

Theorem 4.4.5 *([6]) In an ensemble of weak learners, diversity-effect is non-negative:*

$$\mathbb{E}_{(X,Y),D}\left[\frac{1}{M}\sum_{i=1}^{M}\ell_{0/1}(Y, q_i) - \ell_{0/1}(Y, \bar{q})\right] \geq 0$$

A proof is given in [6]. For binary classification problems, one can see that the weak learner condition implies that good diversity outweighs bad diversity.

4.4.3. Competence in binary classification problems

Theisen et al. consider the question of ensemble improvement under the 0/1-loss. Although they do not acknowledge the connection, their notion of ensemble improvement is exactly diversity-effect. They define an assumption on the ratio of inccorect members.

Definition 4.4.5 *(2-competence, [7]) An ensemble is 2-competent iff*

$$\forall t \in [0, 1/2] : \mathbb{P}_{(X,Y)}\left[W(X,Y) \in [t, 1/2]\right] \geq \mathbb{P}_{(X,Y)}\left[W(X,Y) \in [1/2, 1-t]\right]$$

The condition is illustrated in Figure 4.6.

Proposition 4.4.6 ⋆ *The weak learner condition of Definition 4.4.4 is a special case of 2-competence.*

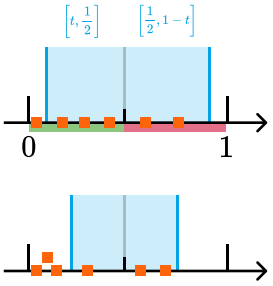

Figure 4.6.: Illustration for the competence condition 4.4.5 for binary classification. ● Squares correspond to pairs (X, Y) from the joint distribution of examples and outcomes. For each of these pairs, the average/expected member error $W_\Theta(X, Y)$ is the ratio of incorrect members. The center $1/2$ is the majority vote threshold. Informally, an ensemble is competent, if, for any two ● intervals defined by t left and right of the threshold, more examples are in the left part. For the upper example, this holds. For the lower example, even though many examples are classified correctly by many members, the ensemble is *not* competent.

Proof. The weak learner condition implies that, also in expectation

$$\mathbb{E}_{\Theta,D}\left[\mathbb{E}_{(X,Y)}\left[\ell_{0/1}\left(Y,\bar{q}\right)\right]\right] \geq 1/2$$

Consequently, using Lemma 4.4.1, we have

$$\begin{aligned} 1/2 \leq \mathbb{E}_{\Theta,D,(X,Y)}\left[\ell_{0/1}\left(Y,\bar{q}\right)\right] &= \mathbb{E}_{(X,Y)}\left[W\right] \\ \leftrightarrow\ \mathbb{E}_{(X,Y)}\left[\,\mathbb{1}\left[W \in [0,1/2]\right]\right] &= \mathbb{P}_{(X,Y)}\left[W \in [0,1/2]\right] = 1 \end{aligned}$$

□

The 2-competence condition was used to show two kinds of results [7]:

- In 2-competent ensembles, diversity-effect is non-negative.
- For 2-competent ensembles, the ensemble generalisation error is bounded from above and below by linear functions of the expected disagreement between two members.

One can see that competence is essentially determined by the distribution of examples (X, Y) over the range of $W(X, Y)$ which is divided by the majority vote threshold $1/2$. We have already seen that, similarly, diversity-effect in its apparent form of good and bad diversity is determined by just the same characteristics. How are these two related? We will argue that non-negative diversity-effect is in fact equivalent to a notion of competence generalised to $k > 2$ classes. Unless otherwise noted, all expectations and probabilites are over the distribution of (X, Y).

While proving that 2-competence implies non-negative diversity-effect, Theisen et al. establish the following fact.

$$\text{ens. 2-competent} \ \leftrightarrow\ \mathbb{E}\left[W\,\mathbb{1}\left[W < 1/2\right]\right] \geq \mathbb{E}\left[\overline{W}\,\mathbb{1}\left[\overline{W} \leq 1/2\right]\right] \tag{4.1}$$

We can rearrange this into a more suggestive form. Recall that $W = \mathbb{E}_{D,\Theta}\left[\ell_{0/1}\left(Y, q_{D,\Theta}(X)\right)\right]$. We now split off the expectation over D and instead consider $W_\Theta = \mathbb{E}_\Theta\left[\ell_{0/1}\left(Y, q_{D,\Theta}(X)\right)\right]$. Rearranging the above and exploiting the linearity of expectation, we obtain

$$d =_{\text{def}} \mathbb{E}_{(X,Y),D}\left[W_\Theta\,\mathbb{1}\left[W_\Theta < 1/2\right]\right] - \mathbb{E}_{(X,Y),D}\left[\overline{W_\Theta}\,\mathbb{1}\left[\overline{W_\Theta} \leq 1/2\right]\right] \ \geq\ 0$$

The indicator functions are mutually exclusive and can thus be understood as a case distinction.

$$d = \begin{cases} \mathbb{E}_{X_+}\left[W_\Theta\right] & \leftrightarrow W_\Theta < 1/2 \\ \mathbb{E}_{X_-}\left[\overline{W_\Theta}\right] = \mathbb{E}\left[1 - W_\Theta\right] & \leftrightarrow \overline{W_\Theta} \leq 1/2 \end{cases}$$

For $k = 2$, the voting threshold is $1/2$ and thus the conditions correspond exactly to the ensemble being either correct or incorrect.

$$k = 2 \ \rightarrow\ \begin{cases} W_\Theta < 1/2 \ \leftrightarrow\ \bar{q}(X) = Y \\ \overline{W_\Theta} \leq 1/2 \ \leftrightarrow\ \bar{q}(X) \neq Y \end{cases} \tag{4.2}$$

Recalling the characterisation of good and bad diversity of Lemma 4.4.4, we can see that d is nothing else but the diversity-effect.

Corollary 4.4.7 *Non-negative effect is exactly equivalent to* 2*-competence for a classification problem with* $k = 2$ *classes.*

Moreover, it is important to note that the gap of Equation 4.1 and consequently the gap in the definition of 2-competence (⇝4.4.5) is exactly the diversity-effect and exactly measures the ensemble improvement. In other words, we can now speak about the degree of competence of an ensemble.

4.4.4. Competence and Diversity in non-binary classification problems

For $k > 2$, the equivalence between the correctness of the ensemble and less than half of the members being incorrect (⇝4.2) is not given. While it is sufficient (a class with more than $1/2M$ votes will win any plurality vote), it is not necessary: a plurality vote can be won with less than $1/2M$ votes. Thus, there are ensembles which have non-negative diversity-effect (ensemble improvement) that are not 2-competent.

The key difference is that for $k > 2$, the voting threshold for a pair (X, Y) is no longer the same value for all examples. Since a class wins if and only if it has more votes than any other class, the voting threshold depends on the distribution of class votes, which is potentially different for any pair (X, Y). Nevertheless, there is still *a* classification threshold, namely the ratio of votes for the next-best class. Because we will be considering the ratio of incorrect votes as a basic quantity, we will now define it from the reciprocal perspective:

$$\kappa(X, Y) = 1 - \max_{Z \neq Y} \mathbb{E}_\Theta \left[\mathbb{1} \left[q_\Theta = Z \right] \right]$$

and it holds that

$$W_\Theta < \kappa \quad \leftrightarrow \quad \bar{q}(X) = Y \tag{4.3}$$

$$\overline{W_\Theta} \leq 1 - \kappa \quad \leftrightarrow \quad \bar{q}(X) \neq Y \tag{4.4}$$

Let

$$t =_{\text{def}} \max_{Z \neq Y} \mathbb{E}_\Theta \left[\mathbb{1} \left[q_\Theta = Z \right] \right]$$

Then

$$\begin{aligned} \bar{q} = y &\leftrightarrow \overline{W} \geq t \\ &\leftrightarrow W = 1 - \overline{W} < 1 - t = \kappa \end{aligned}$$

and

$$\begin{aligned} \bar{q} \neq y &\leftrightarrow \overline{W} < t \\ &\leftrightarrow 1 - \overline{W} \geq 1 - t = \kappa \\ &\leftrightarrow 1 - (1 - \overline{W}) < 1 - \kappa \\ &\leftrightarrow \overline{W} < 1 - \kappa \end{aligned}$$

Definition 4.4.6 ⋆ *(k-competence) An ensemble is k-competent iff*

$$\forall t \in [0, 1] : \mathbb{P}_{(X,Y)} \left[W \in [t, \kappa] \right] \geq \mathbb{P}_{(X,Y)} \left[W \in [1 - \kappa, 1 - t] \right]$$

for $\kappa =_{def} 1 - \max_{Z \neq Y} \mathbb{E}_\Theta \left[\mathbb{1} \left[q_\Theta = Z \right] \right]$.

Theisen et al. showed that 2-competence implies non-negative diversity-effect. We now show that a very similar line of argument using k-competence actually holds in *both* directions.

Theorem 4.4.8 ⋆ *Consider an ensemble for a k-class classification problem. Then*

$$k\text{-competence} \quad \leftrightarrow \quad \text{diversity-effect} \geq 0$$

Theorem 4.4.9 ⋆ 2*-competence is a special case of k-competence in* 2*-class problems.*

Proof. From Corollary 4.4.7, it holds that

$$k = 2: \quad \text{2-competence} \;\leftrightarrow\; \text{diversity-effect} \geq 0$$

and Theorem 4.4.8 establishes that

$$\text{diversity-effect} \geq 0 \;\leftrightarrow\; k\text{-competence}$$

□

The main work for proving Theorem 4.4.8 lies in establishing the following lemma, which is a generalised form of Equation 4.1.

Lemma 4.4.10 ⋆ *(Generalised from [7]) For an increasing function f with $f(0) = 0$, it holds that*

$$k\text{-competence} \;\leftrightarrow\; \mathbb{E}\left[f(W)\,\mathbb{1}\left[W < \kappa\right]\right] \geq \mathbb{E}\left[f(\overline{W})\,\mathbb{1}\left[\overline{W} \leq \kappa\right]\right]$$

where $\kappa =_{def} 1 - \max_{Z \neq Y} \mathbb{E}_\Theta\left[\,\mathbb{1}\left[q_\Theta = Z\right]\right]$.

Proof. We begin by observing that, for all $x \in [0, 1]$

$$\mathbb{P}\left[W \in [x, \kappa]\right] \cdot\; \mathbb{1}\left[x \leq \kappa\right] = \mathbb{P}\left[W\,\mathbb{1}\left[W < \kappa\right] \;\geq\; x\right]$$
$$\mathbb{P}\left[W \in [1-\kappa, 1-x]\right] \cdot\; \mathbb{1}\left[x \leq \kappa\right] = \mathbb{P}\left[\overline{W}\,\mathbb{1}\left[\overline{W} \leq \kappa\right] \;\geq\; x\right]$$

$$\begin{aligned}
&\mathbb{P}\left[W \in [1-\kappa, 1-x]\right]\;\mathbb{1}\left[x \leq \kappa\right] \\
=&\mathbb{P}\left[W \in [1-\kappa, 1-x]\right]\;\mathbb{1}\left[1-x > 1-\kappa\right] \\
=&\mathbb{P}\left[W\,\mathbb{1}\left[W > 1-\kappa\right] \;<\; 1-x\right] \\
=&\mathbb{P}\left[W\,\mathbb{1}\left[\overline{W} \leq \kappa\right] \;<\; 1-x\right] \\
=&\mathbb{P}\left[\overline{W}\,\mathbb{1}\left[\overline{W} \leq \kappa\right] \;\geq\; x\right]
\end{aligned}$$

where the first factors on the left-hand-side appear in the definition of k-competence. Since W is nonnegative, using that $\mathbb{E}[X] = \int \mathbb{P}[X \geq x]\,dx$, we can conclude that, for any $x \in [0, 1]$

$$\begin{aligned}
(k\text{-comp.}) \;&\leftrightarrow\; \mathbb{P}\left[W\,\mathbb{1}\left[W < \kappa\right] \geq x\right] \;\geq\; \mathbb{P}\left[\overline{W}\,\mathbb{1}\left[\overline{W} \leq \kappa\right] \geq x\right] \\
&\leftrightarrow\; \mathbb{E}\left[W\,\mathbb{1}\left[W < \kappa\right]\right] \;\geq\; \mathbb{E}\left[\overline{W}\,\mathbb{1}\left[\overline{W} \leq \kappa\right]\right]
\end{aligned}$$

□

Using this, we can now directly prove theorem 4.4.8.

Proof. (For Theorem 4.4.8, generalised from [7])

$$\begin{aligned}
0 &= \mathbb{E}\left[(W-1)\,\mathbb{1}\left[W \geq \kappa\right]\right] - \mathbb{E}\left[(W-1)\,\mathbb{1}\left[W \geq \kappa\right]\right] \\
&= \mathbb{E}\left[(W-1)\,\mathbb{1}\left[W \geq \kappa\right]\right] + \mathbb{E}\left[(1-W)\,\mathbb{1}\left[W \geq \kappa\right]\right] \\
&= \mathbb{E}\left[(W-1)\,\mathbb{1}\left[W \geq \kappa\right]\right] + \mathbb{E}\left[\overline{W}\,\mathbb{1}\left[\overline{W} < 1-\kappa\right]\right] \\
&\leq \mathbb{E}\left[(W-1)\,\mathbb{1}\left[W \geq \kappa\right]\right] + \mathbb{E}\left[\overline{W}\,\mathbb{1}\left[\overline{W} < \kappa\right]\right]
\end{aligned}$$

Where the final inequality is enable due to that, for $k \geq 2$, we have $\max_{Z \neq Y} \mathbb{E}_\Theta\left[\,\mathbb{1}\left[q_\Theta = Z\right]\right] < 1/2$ and consequently $\kappa \geq 1/2$ and $\kappa > 1 - \kappa$. Applying Lemma 4.4.10 for $f = \mathrm{id}$ to the

second term yields

$$\begin{aligned}&\mathbb{E}\left[(W-1)\,\mathbb{1}\left[W \geq \kappa\right]\right] + \mathbb{E}\left[\overline{W}\,\mathbb{1}\left[\overline{W} < \kappa\right]\right]\\ \leq\ &\mathbb{E}\left[(W-1)\,\mathbb{1}\left[W \geq \kappa\right]\right] + \mathbb{E}\left[W\,\mathbb{1}\left[W < \kappa\right]\right]\end{aligned}$$

The above already is nothing but the diversity-effect:

$$\begin{aligned}0 \leq&\mathbb{E}\left[(W-1)\,\mathbb{1}\left[W \geq \kappa\right]\right] + \mathbb{E}\left[W\,\mathbb{1}\left[W < \kappa\right]\right]\\ =&\mathbb{E}\left[W\right] - \mathbb{E}\left[\,\mathbb{1}\left[W \geq \kappa\right]\right]\\ =&\mathbb{E}\left[W\right] - \mathbb{P}\left[W \geq \kappa\right]\end{aligned}$$

The first term is the ratio of incorrect members in expectation over all examples and is equal to the error rate of an average member (see Lemma 4.4.1). The second term is the ensemble error. □

$$\begin{aligned}W \geq \kappa &\leftrightarrow 1 - t\\ &\leftrightarrow \overline{W} = 1 - W < t\\ &\leftrightarrow \bar{q} = Y\end{aligned}$$

4.4.5. Bounds for competent ensembles

2-competence was used to show upper and lower bounds for the diversity-effect [7]. Now we show that, with minor adjustments, the same bounds can be derived from k-competence. Besides giving performance guarantees, these bounds are interesting due to that they are expressed in terms of disagreements between members – which until now we have only seen for Bregman divergences.

Theorem 4.4.11 *(Upper bound) In k-competent ensembles,*

$$\mathbb{E}\left[W\right] - \mathbb{P}\left[W \geq \kappa\right] \leq \mathbb{E}_{\rho,\rho'}\left[D(q_\rho, q_{\rho'})\right]$$

for $D(q_\rho, q_{\rho'}) = \mathbb{E}_X\left[\,\mathbb{1}\left[q_\rho \neq q_{\rho'}\right]\right]$ *and* $\rho = (\Theta, D)$.

Proof. The proof can be found in [7]. It does not make use of competence and therefore still holds for k-competent ensembles. □

Theorem 4.4.12 *(Lower bound) In k-competent ensembles,*

$$\frac{2(k-1)}{k}\mathbb{E}\left[D(q_\rho, q_{\rho'})\right] - \frac{3k-4}{k}\mathbb{E}\left[W\right] \ \leq\ \mathbb{E}\left[W\right] - \mathbb{P}\left[W \geq \kappa\right]$$

for $D(q_\rho, q_{\rho'}) = \mathbb{E}_X\left[\,\mathbb{1}\left[q_\rho \neq q_{\rho'}\right]\right]$ *and* $\rho = (\Theta, D)$.

Proof.

$$\begin{aligned}&\mathbb{P}\left[W \geq \kappa\right]\\ &\leq 2\mathbb{E}\left[W^2\right] && \text{(Lemma 4.4.13)}\\ &= 2\mathbb{E}_{\rho,\rho'}\left[L(q_\rho, q_\rho)\right] && \text{(Lemma 3 in [7])}\\ &= \frac{4(k-1)}{k}\left(\mathbb{E}\left[W\right] - {}^1\!/_2\mathbb{E}_{\rho,\rho'}\left[D(q_\rho, q_{\rho'})\right]\right) && \text{(Lemma 4 in [7])}\end{aligned}$$

Rearranging the terms yields the statement. Lemmas 3 and 4 hold without adjustments for k-competence and are shown in [7]. □

We conclude by checking the inequality in the proof of Theorem 4.4.12.

Lemma 4.4.13 ⋆ *(Generalised from [7]) In k-competent ensembles it holds that*

$$\mathbb{P}\left[W \geq \kappa\right] \leq 2\mathbb{E}\left[W^2\right]$$

Proof. The proof is given in B.0.2. □

4.5. Dependency of diversity on outcomes

In the general case, diversity cannot be expressed independently of the outcome variable Y. If ℓ satisfies the triangle inequality, then diversity-effect can be bounded from above by a target-independent term that is reminiscent of the diversity term for Bregman divergences introduced in theorem 4.3.3.

Lemma 4.5.1 ⋆ *Under a symmetric loss satisfying the triangle inequality, diversity-effect is bounded from above by diversity.*

$$\ell \text{ metric } \rightarrow \mathbb{E}_{(X,Y),D}\left[\frac{1}{M}\sum_{i=1}^{M}\ell(Y,q_i) - \ell(Y,\bar{q})\right] \leq \mathbb{E}_{(X,Y),D}\left[\frac{1}{M}\sum_{i=1}^{M}\ell(\bar{q},q_i)\right]$$

Proof. For a $a, b, c \in X$, due to the triangle inequality and symmetry of d, it holds that

$$\begin{aligned} & d(a,b) - d(b,c) \\ & \leq d(a,c) + d(c,b) - d(b,c) \\ & = d(a,c) \end{aligned}$$

□

In particular, the 0/1-loss is symmetric and satisfies the triangle inequality.

5. Growth Strategies

In the previous chapters, we have developed an understanding of how diversity affects the ensemble generalisation error. We have seen that the ensemble error is determined by a tradeoff between the average member error and the diversity, a measure of variance between member predictions. We will now consider how to influence this tradeoff during ensemble construction. The basic idea is to guide the construction of an ensemble member with respect to its interactions with the ensemble constructed so far.

5.1. Diversity is a measure of model fit

There has been a line of ongoing research about capturing the notion of diversity in general ensembles [5, 35, 36] in Random Forests [37–39] or in neural networks [25, 33, 40, 41] (see also the review in section 4.1). One of the early insights was that many *ad-hoc* diversity measures are not clearly correlated to the ensemble performance [5]. We claim that the search for a diversity measure that is always directly correlated with ensemble performance is misguided. As can be seen from the diversity decomposition, there is a trade-off between average member error (in turn decomposed into average member bias and variance) and diversity. As such, diversity is a measure of model fit, just like bias and variance. This was already observed empirically for the case of Negative Correlation Learning with neural network ensembles [27] (as introduced in section 5.4). By varying the regularisation strength parameter λ, one can obtain ensembles with varying diversity but equal ensemble performance. We can also observe this in Random Forests for 0/1-classification problems. Plotting the average member error versus diversity for each ensemble shows a linear trend, see for example Figure 5.1.

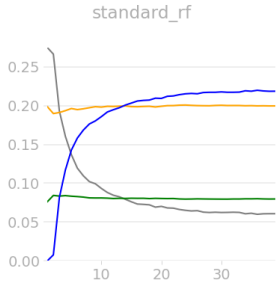

Figure 5.1.: Development of the ● ensemble error of a standard Random Forest ensemble as an increasing number of trees are added (evaluated on *mnist* under 0/1-loss). One can see that ● average member bias and ● average member variance stay roughly equal while ● diversity increases, causing the ensemble error to decrease.

However, diversity is an essential component of ensemble *growth*. For instance in Figure 5.1, we can see that, as the number of trees in the ensemble increases, average member bias and average member variance stay almost constant. This is to be expected since all members are constructed using the same base learner (decision trees). In other words the ensemble is homogeneous. Since the two components stay equal, the improvement in ensemble performance is *exactly* given by the improvement in diversity.

The diversity-effect decomposition holds for both the 0/1-loss and Bregman divergences. The key difference is that Bregman divergences are non-negative and thus there are no points which contribute negatively to the ambiguity-effect term. In other words, any kind of ambiguity is "beneficial" under Bregman divergences – as was previously observed empirically for the squared-error loss [25]. However, the tradeoff between average member error and ambiguity still holds. Since $\frac{1}{M}\sum_{i=1}^{M} B_\phi(y, q_i) - B_\phi(\bar{q}, q_i) = B_\phi(y, \bar{q}) \geq 0$, diversity is upper-bounded by the average member error. This means that any improvement due to diversity can only happen by reducing the amount of error introduced by the individual member error. If there is little average member error to begin with, encouraging diversity will not necessarily improve ensemble performance.

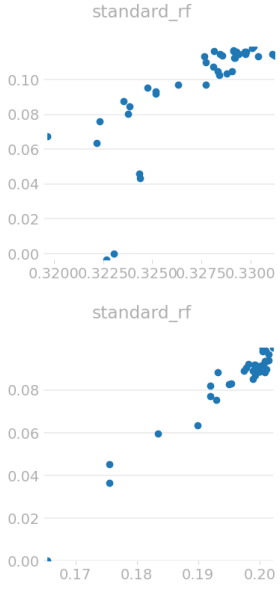

Figure 5.2.: Random Forests with varying number of trees plotted across average member error (vertical axis) and diversity (horizontal axis).

Observation 5.1.1

- *A more diverse ensemble is not necessarily better. The ensemble performance depends on the trade-off between member performance and diversity.*
- *Diversity behaves fundamentally differently when measured in terms of a Bregman divergence or the* $0/1$*-loss. For Bregman divergences, diversity in predictions on a single example is always beneficial (↝4.3). For the* $0/1$*-loss, additional assumption on the member performance are required (↝4.4).*

5.2. Diversity in Random Forests and in Neural Networks

Interpolating and non-interpolating models In the majority of this section, we have considered general ensembles without assumptions on the actual member learners. It appears much of the work on ensemble diversity has been done in the context of neural networks [25, 33, 40, 41], potentially due to the early publishing of Negative Correlation Learning [40] in 1999 and the easy of encouraging diversity simply via a regularisation term. However, it has been observed that encouraging diversity is not always beneficial for neural networks [39–41]. We will now provide a brief explanation on when and why diversity in neural network ensembles is beneficial. Further, we will argue that diversity is an essential component of Random Forests, even more so than for neural network ensembles.

Recall the lower and upper bounds for diversity-effect in Theorems 4.4.11 and 4.4.12. Dividing by the member error rate and in simplified form for $k = 2$ classes, we have

$$\frac{D(q_\rho, q_{\rho'})}{\mathbb{E}[W]} \geq \overbrace{\frac{\mathbb{E}[W] - \mathbb{P}[W \geq \kappa]}{\mathbb{E}[W]}}^{\text{diversity-effect / ensemble improviement}} \geq \frac{D(q_\rho, q_{\rho'})}{\mathbb{E}[W]} - 1$$

where $D(q_\rho, q_{\rho'}) = \mathbb{E}_{X,\Theta}\left[\mathbb{1}\left[q_\rho \neq q_{\rho'}\right]\right]$ is the *disagreement rate*. Considering the quantities in relation to $\mathbb{E}[W]$ normalises the error rates in relation to the problem difficulty and gives a more informative measure of ensemble improvement, the ensemble improvement *rate*.

$$\text{EIR} =_{\text{def}} \frac{\mathbb{E}[W] - \mathbb{P}[W \geq \kappa]}{\mathbb{E}[W]}$$

The normalised variant of the disagreement is the *disagreement-error-ratio*:

$$\text{DER} =_{\text{def}} \frac{D(q_\rho, q_{\rho'})}{\mathbb{E}[W]}$$

If DER < 1, then the lower bound is lower than zero and the ensemble improvement can potentially be negative, i.e. ensembling may hurt performance. If DER ≥ 1, the power bound is greater than zero and ensembling can not hurt performance. Theisen et al. [7] analyse the behaviour of DER with varying model complexity. Of particular interest is the behaviour around the *interpolation threshold*, which is the minimum model complexity that allows a zero training error. They show that, in neural networks and bagged logistic regression ensembles, disagreement rate and ensemble improvement are maximised at the interpolation threshold and decrease beyond. In other words, higher-capacity models benefit less from diversity, or diversity may even hurt performance. The same conclusion was reached after empirical investigation in

neural network ensembles by Abe et al. [33]. In Random Forests, on the other hand, the model complexity is inherently bounded by the data [7, 27]. A decision tree can only be grown until each leaf is perfectly pure, which corresponds to zero training error. In other words, Random Forests naturally can not go *into* the interpolation regime.

Further, decision trees could be understood to be of much lower capacity than deep neural networks. This likely reflects in the average member error, although a quantitative comparison has to be left for future work. Ensemble improvement can only be high if the average member error is high to begin with. In other words, ensembling can be expected to be much more beneficial for low-capacity models. This was also observed by Abe et al. [33] for the case of neural networks.

Expressing diversity and average member error in terms of tree structure As we have seen in Subsection 3.2.2, the generalisation error can be expressed as a weighted sum over forest cells. We can also express the ambiguity decomposition (⇝4.2.1) in terms of forest cells. For sake of clarity we omit the expectation over D and write $Z =_{\text{def}} (X, Y)$.

Proposition 5.2.1 ⋆ *Let $Z = Z_1 \dot\cup \ldots \dot\cup Z_P$ be a forest partition of $Z = (X, Y)$.*

$$\mathbb{E}_Z\left[\ell(y, \bar{q})\right] = \sum_{p=1}^{P} \mathbb{P}\left[Z_p\right] \underbrace{\frac{1}{M}\sum_{i=1}^{M} \mathbb{E}_Z\left[\ell(y, q_i)\right]}_{err(Z_p)} - \underbrace{\frac{1}{M}\sum_{i=1}^{M} \mathbb{E}_Z\left[LE(\bar{q}, q_i)\right]}_{div(Z_p)}$$

$$= \sum_{p=1}^{P} \mathbb{P}\left[Z_p\right] \frac{1}{M}\sum_{i=1}^{M} \mathbb{E}_Z\left[\ell(y, q_i) - LE(\bar{q}, q_i) \mid Z_p\right]$$

$\mathbb{P}\left[Z_p\right]$ corresponds to the area of the forest cell p and can be determined based solely on the decision boundaries of the corresponding tree cells. $\text{div}(Z_p)$ depends solely on the outputs of the tree cells that constitute Z_p. It is simply the generalised variance (see 1.6.3) of these leaf outputs. If leaf outputs are saved with the tree model after construction, these can be directly read off the model. However, these values still depend directly on the training data. Instead, we would like to infer the value based on the decision boundaries alone. In Random Forests, trees are grown *deeply*, that is, until each leaf contains only a single point. Consequently, if the number of data points n is large, leaf cells will become small. Provided that the regression function $m(x) = \mathbb{E}\left[Y \mid X = x\right]$ is uniformly continous, small leaf cells imply that the variation of m throughout a leaf is bounded. In fact Scornet, Biau, and Vert [15] show that, as n grows, the variation in a leaf becomes arbitrarily small (under assumptions). This motivates that, instead of the data-dependent leaf output $q_i(x)$, one could instead use the center of the cell $\tilde{q}_i(x)$, depending only on the decision boundaries. If the variation inside a cell vanishes, then also the error of using $\tilde{q}_i(x)$ over $q_i(x)$ vanishes.

5.3. Guided sequential training of member models

The statistical analysis in Chapter 4 has always considered a given ensemble of fixed size. We consider the question on how to make use of the diversity theory to guide ensemble construction.

Can we encourage diversity in Random Forests? In previous chapters, we have seen that diversity is *the* essential force that makes Random Forests work. In the standard Random Forest scheme as introduced by Breiman, individual trees are diverse due to random choices made during tree construction. Can we instead guide tree construction such that the Random Forest ensemble is more diverse? Encouraging diversity informed by the ambiguity decomposition has already been investigated for neural networks under the Negative Correlation Learning framework (⇝5.4). For Random Forests, there are many proposed variations that are thought to influence an intuitive notion of diversity [37, 38, 42–45] but to the best of our knowledge none so far have considered the theory based on the ambiguity decomposition we introduced here.

Can we produce Random Forests with better generalisation error? Early experiments have shown that diversity-informed training strategies can indeed produce forests with better generalisation error as compared to training trees independently [28, 46, 47]. However, the approaches were based on intuition or disconnected theory. Informed by the diversity theory presented herein, can we guide tree construction such that the ensemble performance is better?

Can we produce smaller ensembles? In standard Random Forests, the ensemble becomes increasingly diverse as more random trees are grown. Can we make more informed choices during tree construction such that fewer trees are needed to achieve the same diversity, without affecting other components of the error?

We focus on classification problems evaluated by the 0/1-loss. Due to the inherently different nature of binary and non-binary classification problems (⇝4.4.4 on competence), we first focus on binary classification problems and then later consider non-binary problems later.

Arguably one of the simplest ways to influence a learning algorithm is by assigning *weights* $w : X \to [0, 1]$ to examples. During any computations that involve a term corresponding to an example, for instance the impurity for a tree node (⇝3.1), the term is multiplied by the example weight. For random forests, example weights can possibly come into effect via two mechanisms.

- *Weighted bootstrapping*: Instead of drawing the bootstrap samples uniformly, draw a sample with probability according to its weight. If the bootstrap sample is large, examples with higher weight are more likely to be oversampled and thus appear multiple times in the bootstrap sample.
- *Weighted tree construction*: We have seen in Section 3.1 that tree construction according to some impurity measure greedily optimises a loss function. Likewise, weighting examples during computation of the impurity measure optimises a weighted loss.

We perform experiments for both approaches.

Note that if a next member is constructed based on the performance of a previous members, the member models are no longer statistically independent and the ensemble is no longer homogeneous. This does not prevent us from considering their bias-variance-diversity decomposition (it holds just as well for heterogeneous ensembles).

In the following sections we visualise the development of the components of the diversity decomposition as trees are added to a Random Forest ensemble. All quantities are averaged over 3 trials to resemble the expectation over D.

5.3.1. A simple weighting function for binary classification

We begin by considering a simple weighting function that is already given in the literature without theoretical motivation. We explain this weighting function from the perspective of diversity and argue that, while its application can be beneficial, it is in contradiction to theory. This motivates further weighting functions which show equal or better performance. We further close some gaps in the original publication.

Definition 5.3.1 *(DRF weighting scheme [46]) Let $\bar{q}$ be the ensemble constructed so far. For a pair $(X, Y) \in D$, define the Dynamic Random Forest weighting scheme as*

$$w_{DRF}(X) =_{def} W(X) \qquad \text{for } W(X) =_{def} \frac{1}{M} \sum_{i=1}^{M} \ell_{0/1}(Y, q_i(X))$$

This will have the effect that correctly classified examples are assigned lower weight and incorrectly classified examples are assigned higher weight.

Xu and Chen [48] re-iterate on the DRF weighting scheme and propose an alternative scheme.

$$w_{\text{XuChen}}(X) =_{\text{def}} \begin{cases} W(X)^2 & \text{if } W(X) \leq 1/2 \\ \sqrt{W(X)} & \text{if } W(X) > 1/2 \end{cases}$$

Again, the authors provide only a heuristic motivation, which is that, compared to w_{DRF}, their method has a more drastic effect of up- and downweighting.

Assuming the weights are used effectively, w_{DRF} and w_{XuChen}, will have the effect of moving example-outcome pairs towards $1/2$ on the distribution of $W(X, Y)$.

We perform experiments using w_{DRF} (both weighted bootstrapping and weighted tree construction) and w_{XuChen} (weighted bootstrapping only). The full results for all benchmark datasets are given in Appendix A. An illustrative comparison for a single dataset is given in Figures 5.4 and 5.5.

Observation 5.3.1

- *w_{DRF} and w_{XuChen} with weighted bootstrapping indeed lead to more diverse ensembles. For weighted tree construction, diversity is very similar to standard Random Forests.*
- *For weighted bootstrapping, there is an initial sharp increase in diversity and average member error. Later, diversity decreases. We will come back to this effect in later experiments.*

Comparing weighted bootstrapping and weighted tree construction In practise, any bootstrap sample is finite. The bootstrap sample is drawn with replacement, thus a bootstrap sample does not necessarily include all examples from the training dataset. Under uniform bootstrapping, each example has equal chance to be included in the bootstrap sample (↪3.2.1). It is then considered during tree construction according to its weight. On the other hand, under weighted bootstrapping, examples with high weight are more likely and examples with low weight are less likely to appear in the bootstrap sample. In particular, this means that examples with low weight are more likely to not be included at all in the bootstrap sample and consequently not be considered at all during tree construction. This might be an intuitive explanation why

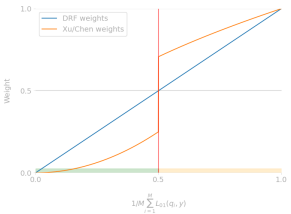

Figure 5.3.: Illustration of ● w_{DRF} and ● w_{XuChen}.

Using diversity-effect, we can give a more informed interpretation of w_{XuChen}. Inspecting w_{DRF}, which is continuous around the majority vote threshold, one can see that very similar weights are assigned to examples which are classified just barely correctly (resulting in a 0/1-loss of 0) and examples which are classified just barely incorrectly (resulting in a 0/1-loss of 1). This may mean suboptimal guidance in ensemble construction since both cases have very similar weights, but their effect on the ensemble loss is actually dramatically different. One disadvantage is that we take a heuristic step away from theory.

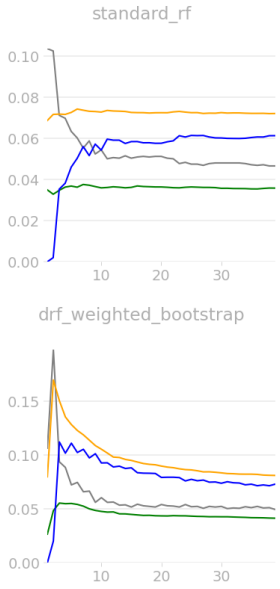

Figure 5.4.: Standard Random Forest and an ensemble constructed with the DRF weighting scheme (↪5.3.1, weighted bootstrapping). For the DRF scheme, neither ● average bias nor ● average variance are lower, hence the average member error is not lower. However, ● diversity is much larger. Evaluated on the *spambase* dataset.

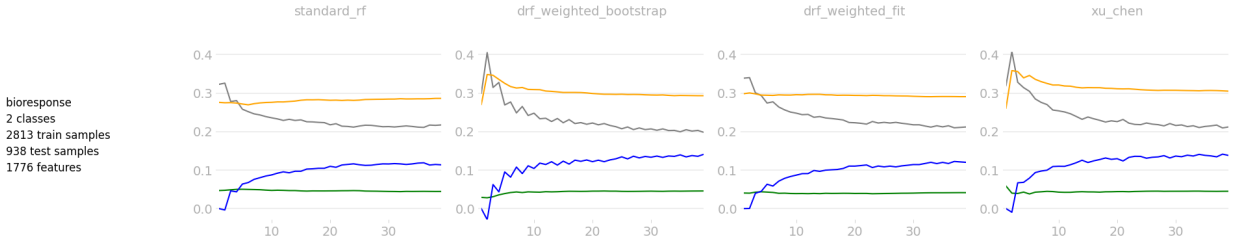

Figure 5.5.: Comparison of w_{DRF} and w_{XuChen} on a high-dimensional binary classification dataset. The full results for all benchmark datasets can be found in Figure A.1 (Legend: ● average member bias, ● average member variance, ● diversity, ● ensemble generalisation error.)

weighted bootstrapping shows a stronger effect than weighted tree construction in our experiments. A thorough discussion has to be left for future work.

5.3.2. A weighting function informed by diversity for binary classification

Informed by the theory on diversity, we make the following two claims.

1. Incorrectly classified examples should not receive a particularly higher weight as w_{DRF} does. If anything, examples for which the majority of members is incorrect should receive *lower* weight since any subsequent correct vote will only increase bad diversity (unless the majority vote is tipped).
2. Weights should not be related linearly to the ratio of incorrect members W. For diversity in 0/1-classification tasks, it does not matter how many members are correct, as long as the ensemble prediction is correct. Thus, any correctly classified point should receive the same low weight. In the extreme case, the weighting function could be a step function. Interpolating between a step function and a linear function could be expected to provide some sort of smoothing.

We introduce a weighting function in the form of a parameterised sigmoid function. This is mostly a pragmatic choice as it easily allows to configure the shape of the function by varying its parameters. We make no claims of the superiority of this function as compared to, for example, a piecewise linear function with varying slope.

$$\text{sigmoid}(x; a, b, k) =_{\text{def}} 1/k(1 + \exp(a - bx))$$

For $a = 0$ and $k = 1$, this function ranges in $[0, 1]$ with its inflection point at $(0, 1/2)$ and its lower half in the range $[0, 1/2]$. The parameter b controls the "steepness". For low b, sigmoid becomes similar to $x \mapsto 1/2$. For high b, sigmoid becomes similar to the step function with threshold 0.

Definition 5.3.2 *(Clipped sigmoid weighting function)*

$$\begin{aligned} w_{sigm}(X) &=_{def} \min\{s, 1/2\} \\ \text{for } s &=_{def} sigmoid(W(X) - t; 0, b, 1) \\ t &=_{def} 1/2 \\ b &=_{def} b_{max} \end{aligned}$$

where the minimum clips the function values to a maximum of $1/2$, $t = 1/2$ is the voting threshold.

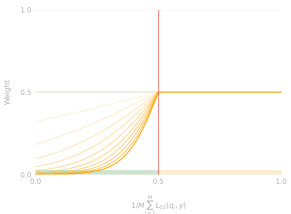

Figure 5.6.: Illustration of ● w_{sigm} for ● $t = 1/2$ and varying b.

The value $b_{\max} =_{\text{def}} 15$ was chosen based on intuition to provide a smoothed version of a step function (compare Figure 5.6). The clipping using t has the effect of clipping the function exactly at the classification threshold, meaning that incorrectly classified points will be assigned uniform weights.

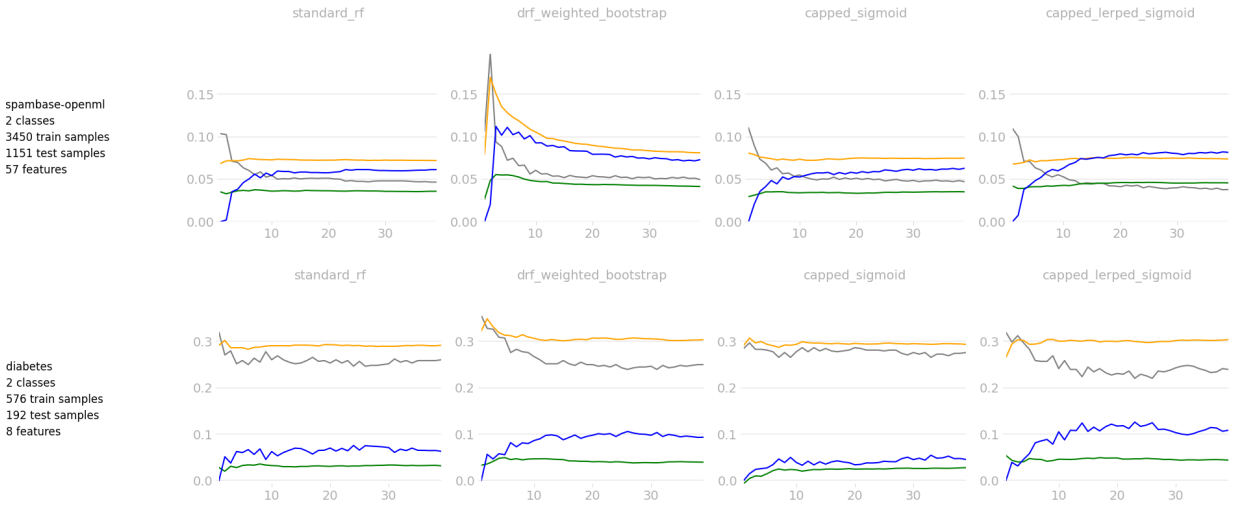

Figure 5.7.: Comparison of w_{DRF}, w_{sigm} and w_{lerp}. *diabetes* is a very small dataset with a high best achievable error rate. The full results for all benchmark datasets can be found in Figure A.2. (Legend: average member bias, average member variance, diversity, ensemble generalisation error.)

Additionally, $W \approx \frac{1}{M}\sum_{i=1}^{M} \ell_{0/1}(Y, q_i)$ can not be expected to be a good approximation for low M. For the first couple of trees, the distribution will in fact be very discrete. We have already seen that in weighted bootstrapping, there is an initial sharp increase in member error and diversity (↝5.3.1). We aim to downregulate the influence of the weighting function for the first couple of trees in the ensemble. After this initial period, the function should have the same shape for all $M > M_{\text{max}}$.

Definition 5.3.3 *(Lerped clipped sigmoid weighting function) $w_{lerp}(X)$ is defined analogously to w_{sigm} (↝5.3.2) but b is linearly interpolated ("lerp"-ed) between 0 and b_{max} according to M.*

$$b \leftarrow \frac{\min\{M, M_{max}\}}{M_{max}} b_{max}$$

Observation 5.3.2 *(↝Figures 5.7 and A.2)*

- *Clipping (not assigning higher weights to incorrectly classified examples) already mitigates the spike in member error and diversity. Interpolating the steepness of the weighting function b is not required for this.*
- *Surprisingly, interpolating leads to higher diversity (sometimes even higher than w_{DRF}) and lower or similar ensemble generalisation error.*

5.3.3. Weighting functions for non-binary classification

The weighting schemes of the previous section are all based on influencing the ratio of incorrect members for a given example. Due to the diversity decomposition, for examples that are still classified correctly by the ensemble, the improvement due to diversity is greater if the ratio of incorrect members is closer to the classification threshold. For binary classification problems, one can always assume a voting threshold

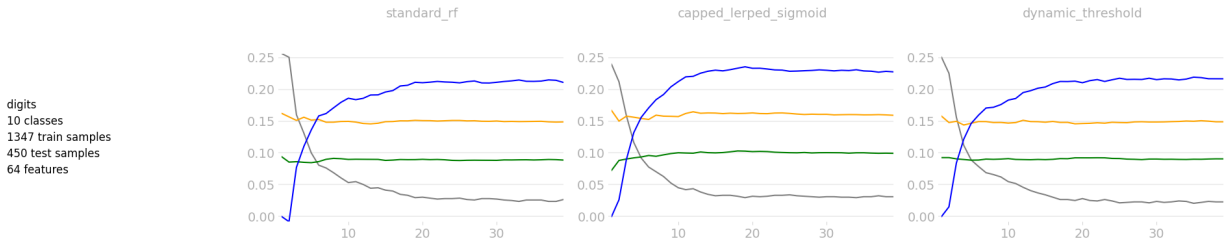

Figure 5.8.: Comparison of w_{lerp} and w_{dyn} for a non-binary classification problem. Full results for all benchmark datasets can be found in Figure A.3

of $1/2$.

$$k = 2 \;\rightarrow\; \begin{cases} W_\Theta < 1/2 \;\leftrightarrow\; \bar{q}(X) = Y \\ \overline{W_\Theta} \leq 1/2 \;\leftrightarrow\; \bar{q}(X) \neq Y \end{cases} \tag{5.1}$$

In Subsection 4.4.4, we have observed that for non-binary classification problems, the voting threshold is exactly the ratio of votes for the next-best class. This is a quantity that depends on the ensemble and the given example and the equivalences above no longer holds. Because of this, we hypothesize that it is inadequate to use a weighting function centered around a threshold of $1/2$ and it should rather be centered around the actual, dynamic threshold t.

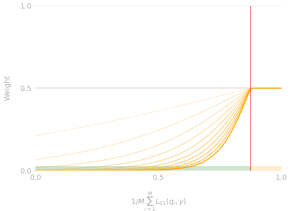

Figure 5.9.: w_{dyn} for $t = 1 - \frac{1}{8}$

Definition 5.3.4 *(Sigmoid weighting function with dynamic threshold) w_{dyn} is defined analogously to w_{lerp}, but t is given as*

$$t =_{def} \kappa(X, Y) = 1 - \max_{Z \neq Y} \mathbb{E}_\Theta [\, \mathbb{1}\,[q_\Theta = Z]]$$

We evaluate this approach on multiple non-binary classification problems.

Observation 5.3.3 *(↝Figure 5.8)*

- *Using w_{dyn}, one can achieve slighly improvided ensemble generalisation error as compared to any other learner (we consider the generalisation error in Subsection 5.3.5)*
- *The development of diversity is slower than with w_{lerp} and more similar to standard Random Forests.*

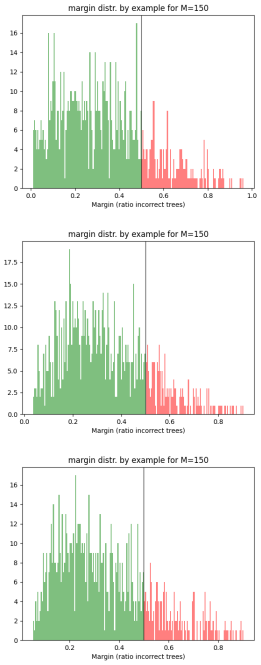

Figure 5.10.: Frequency histogram of the distribution of $W(X, Y)$ over all test pairs (X, Y) for ensembles of $M = 150$ trees. In binary classification problems, $W(X, Y) < 1/2$, the ensemble is ● correct, else it is ● incorrect.

5.3.4. Ensemble margins

Recall that the ensemble margin for an example-outcome pair (X, Y) is the difference between the ratio of members voting for the correct class and the ratio of members voting for the next-best class. In binary classification problems, this corresponds to the average member error. The main motivation behind the voting schemes introduced in this section is that they supposedly encourage diversity. We have argued that diversity should be encouraged on examples while the ensemble prediction is still correct. In other words, on correctly classified examples, the average member error should *increase*. This corresponds to the notion of "good" diversity in binary classification. Weighting schemes like w_{DRF} and w_{XuChen} further encourage diversity on incorrectly classified examples. The expected member error should *decrease*. We can measure and visualise these two components separately as they develop as the size of the ensemble grows.

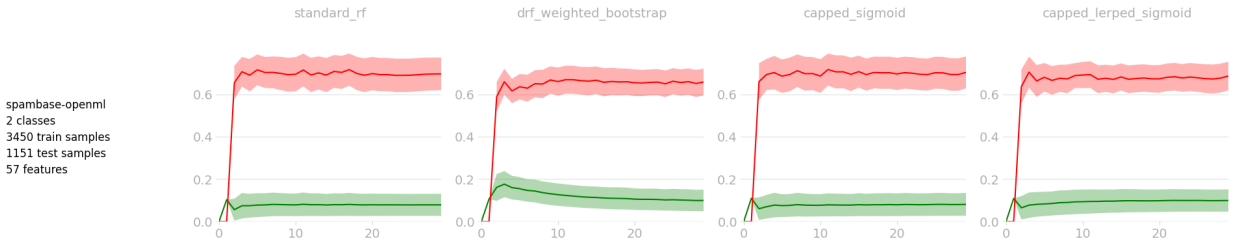

Figure 5.11.: The ensemble margin (↪2.2.3) plotted separately for points for which the ensemble is ● correct (X_+) or ● incorrect (X_-).

Likewise, for a given ensemble, we look at the distribution of ensemble margin per example. We plot a histogram of examples with respect to the number of trees incorrectly classifying that example. In binary classification, more than $1/2M$ trees being incorrect leads to an incorrect ensemble prediction.

Observation 5.3.4 *(↪Figures 5.10 and 5.11)*

- *For w_{DRF}, we can clearly observe that, as expected, the ensemble margin is moved towards the threshold of $1/2$.*
- *For w_{lerp}, the ratio of incorrect members on correctly classified points is higher, while the behaviour for incorrectly classified points is similar.*

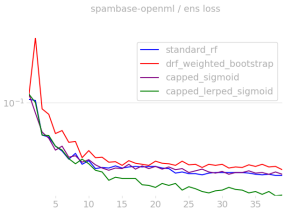

Figure 5.12.: Development of the ensemble generalisation error of different learners as additional trees are added to the resp. ensembles. For the first 10 trees, the error develops similarly in any learner. Beyond, ● w_{lerp} continues to improve while other learners do not. The y-axis is log-scaled.

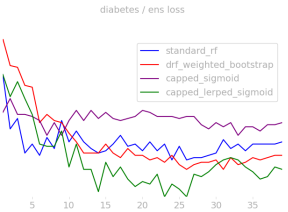

Figure 5.13.: Development of the generalisation error of different learners for a noisy binary classification problem. ● w_{lerp} achieves lowest generalisation error across all learners with just $M = 23$ trees, but behaviour is very irregular. The y-axis is log-scaled.

5.3.5. Ensemble size and generalisation error

Binary classification

We evaluate the overall ensemble generalisation error of ensembles produced by different weighting strategies. We emphasize that we make no claims about the applicability of a specific learner to a specific dataset. We specifically picked the benchmark datasets to provide diverse classification problems. The datasets have different properties: some are higher-dimensional and some have a high best achievable error rate, implying that the data may be noisy. Likewise, we make no claims about the adequacy of hyperparameter choices for a given dataset. We merely compare different strategies on the same benchmark datasets and for the same hyperparameter choices.

Let us first consider the weighting function for binary classification tasks. In Table 5.1, we compare the minimum achieved generalisation error over all values of M. As already observed in the literature [27, 47], the generalisation error can be expected to be a decreasing function of the number of trees, given that the data is sufficiently large and regular. In Figures 5.12 and 5.13, we can observe that the behaviour can be irregular for some datasets, although a decreasing trend is still clear.

Table 5.1.: minimum ensemble test error achieved over all M

	RF	w_{DRF}	w_{sigm}	w_{lerp}
qsar-biodeg	**0.101**	0.144	0.130	0.133
diabetes	0.247	0.240	0.266	**0.220**
bioresponse	0.211	**0.198**	0.217	0.209
spambase-openml	0.047	0.050	0.047	**0.038**
digits	**0.024**	0.033	0.029	0.030
mnist-subset	**0.060**	0.066	0.064	0.062
cover	0.049	**0.042**	0.054	0.042

Non-binary classification

In Table 5.2, we analyse the dynamic threshold weighting function w_{dyn} for non-binary classification tasks. We additionally include w_{lerp} in the comparison but note that it's theoretical motivation here is lacking: It is based on a classification threshold of $1/2$, which does not apply to non-binary classification problems. w_{DRF} was derived with a threshold of $1/2$ but can be applied to any threshold since it is a linear function.

Table 5.2.: minimum ensemble test error achieved over all M

	RF	w_{DRF}	w_{lerp}	w_{dyn}
digits	0.024	0.033	0.030	**0.021**
mnist-subset	0.060	0.066	0.062	**0.056**
cover	0.049	**0.042**	0.042	0.050

Observation 5.3.5

- *It is indeed possible to obtain better ensembles using diversity-encouraging weighting strategies, as compared to the standard Random Forest procedure. However, none of the evaluated strategies yielded consistently better ensembles across all benchmark datasets.*
- *It is possible to obtain smaller ensembles with similar generalisation error. However, here too no strategy yields a consistent improvement across all datasets.*
- *There is no evident relationship between diversity and ensemble generalisation error. This is not surprising, as we have already argued that ensemble performance is determined by a trade-off of member performance and diversity.*

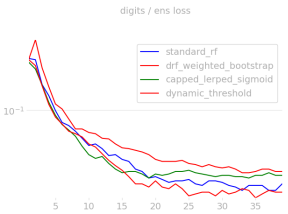

Figure 5.14.: Development of the ensemble generalisation error for a non-binary classification problem. The y-axis is log-scaled.

It is interesting that w_{sigm} consistently performs worse than w_{lerp}. This suggests that the interpolation of the steepness of the weighting function indeed has a beneficial effect.

We further note that even though we have argued that w_{DRF} is counterproductive in theory in that it also increases bad diversity, we observe that nevertheless, we often obtain competetive models with this strategy. This strategy has a very characteristic growth behaviour, as described in Section 5.3.

5.4. Guided parallel training of member models

Consider the bias-variance-covariance decomposition for the squared error loss given in Theorem 4.1.1. The covariance term contributes positively to the ensemble error if the outputs of the members are positively correlated. This suggests the idea that it might be beneficial to train ensemble members in coordination such that their outputs are uncorrelated. Liu and Yao [40] provide an implementation of this idea for neural network ensembles called *Negative Correlation Learning* (NCL). Neural network training involves a *forward* and a *backward* pass. In the forward pass, the neural network in its current state is queried with training data. Its prediction is evaluated according to a loss function. Then, the model parameters are updated according to the gradient of the loss function with respect to the parameters. A common practise is to add a *regularisation term* to this loss function in order to bias the model towards e.g. sparse predictions [8]. In Negative Correlation Learning, forward and backward pass are performed synchronously for all members. This enables using loss functions that depend not only on an individual member but the entire ensemble.

Note that the name is misleading here. The goal is not that members are negatively correlated, but *un*correlated.

The i-th member network is trained using a loss function e_i that contains a regularisation term p_i, which is intended to influence the training of the i-th member such that its predictions are uncorrelated with the other members. The hyperparameter $\lambda \in \mathbb{R}$ determines how much weight is put on either of the two components. The training objective for the i-th network is defined as

$$e_i =_{\text{def}} (y - q_i)^2 \;+\; \lambda\, p_i$$

where the first term is the individual error of the i-th member and p_i is the regularisation term. [40] defined p_i as

$$p_i =_{\text{def}} \left((\bar{q} - q_i) \sum_{i \neq j}^{M} (\bar{q} - q_i) \right) = -(\bar{q} - q_i)^2$$

$\bar{q}$ is the arithmetic mean:

$$\bar{q} = \frac{1}{M} \sum_{i=1}^{M} q_i$$

The sum of deviations around the mean is zero:

$$\sum_{i=1}^{M} (\bar{q} - q_i) = 0$$

Omitting one member from the sum implies

$$\sum_{i \neq j} (\bar{q} - q_i) = -(\bar{q} - q_j)$$

This is reminiscent in shape of the contribution of the i-th member to the covariance term in the bias-variance-covariance decomposition — however, the expectation over D is replaced with the ensemble combiner $\bar{q}$! Seeing that here the ensemble combiner is the arithmetic mean of member outputs, this means that instead of a variance under variations in D, this considers the variance between member model outputs.

Note that the penalty term corresponds to the contribution of the i-th member to the ambiguity (⇝4.2.1). Indeed, as Brown et al. [32] noted, the NCL approach can be motivated more soundly via the ambiguity decomposition.

Definition 5.4.1 *(NCL neural network objective for squared error [32]) The loss function of the i-th neural network ensemble member is defined as*

$$e_i(y, x) =_{def} (y - q_i)^2 \;- \lambda(\bar{q} - q_i)^2$$

The penalty coeffient λ smoothly interpolates between training q_i to either maximise its individual performance or the ensemble performance. For sake of simplicitly, let us assume a factor of $1/2$ on the objective. This yields an equivalent optimisation problem, but allows us to to express the gradients in a simple form.

$$\frac{\partial e_i}{\partial q_i} = \frac{1}{M} \left((q_i - y) - \lambda(q_i - \bar{q}) \right)$$

and

$$\lambda = 0 \;\rightarrow\; \frac{\partial e_i}{\partial q_i} = \frac{1}{M}(q_i - y) = \frac{1}{M}\frac{\partial e_i}{\partial q_i}$$
$$\lambda = 1 \;\rightarrow\; \frac{\partial e_i}{\partial q_i} = \frac{1}{M}(\bar{q} - y) = \frac{\partial e_{\text{ens}}}{\partial q_i}$$

Several authors have attempted to generalise Negative Correlation Learning or transfer it to other loss functions. For instance, Webb et al. [16] directly prove an ambiguity decomposition for the KL-divergence K and propose an analogous objective where $\bar{q}$ is the geometric mean combiner. The proof is not trivial and does not give insight into whether such a decomposition and learning strategy might also exist for other loss functions.

Buschjäger, Pfahler, and Morik [28] approached the problem by attempting to derive a generalised decomposition of the ensemble error that incorporates diversity. The basic idea is to consider a Taylor approximation around the ensemble combiner, which is assumed to be the arithmetic mean combiner. $\bar{q} =_{\text{def}} \mathbb{E}_\Theta[q_\Theta] \approx \frac{1}{M}\sum_{i=1}^M q_i$.

$$\begin{aligned}\mathbb{E}[\ell(y,q)] = \mathbb{E}[\ell(\bar{q})] &+ \mathbb{E}\left[(q-\bar{q})^\top \nabla_{q^\star}(\ell(\bar{q}))\right] \\ &+ \mathbb{E}_\Theta\left[1/2(q-\bar{q})^\top \nabla^2_{\bar{q}}(\ell(q^\star))(q-\bar{q})\right] \\ &+ \mathbb{E}[R_3]\end{aligned}$$

R_3 is the remainder of the Taylor approximation, which vanishes if the third derivative of ℓ is zero. By definition of $\bar{q}$, $\mathbb{E}_\Theta\left[\nabla_{\bar{q}}(\ell(\bar{q}))\right] = \nabla_{\bar{q}}(\ell(\bar{q}))$ and $\mathbb{E}_\Theta[q-\bar{q}] = 0$ and thus the second term vanishes.

The expectations are approximated as follows:

$$\bar{q} = \mathbb{E}_\Theta[q_\Theta] \approx \frac{1}{M}\sum_{i=1}^M q_i$$

$$\mathbb{E}_\Theta\left[1/2(q-\bar{q})^\top \nabla^2_{\bar{q}}(\ell(\bar{q}))(q-\bar{q})\right] \approx 1/2\frac{1}{M}\sum_{i=1}^M d_i^\top D d_i$$
$$\text{for } D =_{\text{def}} \nabla^2_{\bar{q}}(\ell(\bar{q}), y) \text{ and } d_i =_{\text{def}} (\bar{q} - q_i)$$

$$\mathbb{E}_\Theta[R_3(x)] \approx \tilde{R}$$

Note that here they *assume* the expected model $\bar{q}$ to also be the ensemble combiner, i.e. $\bar{q} \approx \frac{1}{M}\sum_{i=1}^M q_i$. Based on this, they propose the following generalised training objective.

Definition 5.4.2 *(Generalised NCL objective as proposed by [28])*

$$e_i =_{def} \frac{1}{M}\sum_{i=1}^M \ell(y, q_i) - 1/2\frac{1}{M}\sum_{i=1}^M d_i^\top D d_i$$

where $D =_{def} \nabla^2_{\bar{q}}(\ell(y,\bar{q})$ *and* $d_i =_{def} (q_i - \bar{q})$

This is based on the assumption that the remainder to the Taylor approximation R_3 is negligibly small. Further, it does not apply to all loss functions. Let us consider some examples of commonly used loss functions.

Squared error loss:

$$\ell(y,x) =_{\text{def}} (y-x)^2$$

Negative log-likelihood:

$$\ell(z,y) =_{\text{def}} -\sum_i^k y_i \log(z_i)$$

Cross-entropy loss:

$$\ell(z,y) =_{\text{def}} -\sum_i^k y_i \log\frac{e^{z_i}}{\sum_j e^{z_j}}$$

- For the squared error, the third derivative vanishes and thus the decomposition is exact.
- For the negative log-likelihood, the third derivate is not bounded and thus this decomposition can not be used.
- For the cross-entropy loss, the third derivative is bounded and while the decomposition is not exact, it can be used for approximation.
- For any other loss function, this would have to be checked.

It is evident however, that we already have a fully general ambiguity decomposition at hand, namely the ambiguity-effect decomposition (⇝4.2.1). This decomposition is exact and holds for *any* loss function (including the 0/1-loss). For Bregman divergences, this reduces to the ambiguity decomposition. We claim that the adequate generalisation of the NCL objective follows this structure.

Proposition 5.4.1 ⋆ *We propose the following generalisation of the NCL neural network objective. For general loss functions ℓ:*

$$e_i =_{def} \ell(y, q_i) \quad - \lambda \left(\ell y, q_i) - \ell(y, \bar{q})\right)$$

For Bregman divergences:

$$e_i =_{def} B_\phi\left(y, q_i\right) \quad - \lambda B_\phi\left(\bar{q}, q_i\right)$$

NCL for the squared error loss as originally proposed [32, 40], as well as for the KL-divergence [16] are special cases of this.

This provides a general framework for Negative Correlation Learning with arbitrary loss functions. Because it is founded on the exact bias-variance-diversity decomposition, this also yields a natural and intuitive means for understanding and analysing Negative Correlation Learning and its effects.

Note that the weighting strategies of Section 5.4 implicitly optimise a very similar objective. The construction procedure of individual decision trees greedily optimises its own performance (⇝3.1), which corresponds to the first term in the NCL neural network objective. The introduction of example weights based on the ensemble diversity influences the decision tree construction to improve the ambiguity, which corresponds to the regularisation term.

Although the term *Negative Correlation Learning* in the literature refers specifically to neural networks, we can now see that it is rather a style of training ensemble members with respect to the ambiguity decomposition. To the best of our knowledge, only one other algorithm has been published that realises this: Buschjäger and Morik [27] proposes to refine the leaf predictions in a given Random Forest using using gradient descent according to the objective defined in definition 5.4.2. While an experimental evaluation has to be left for future work, we note that the case of KL-divergence was already investigated empirically in detail [16].

6. Conclusion

6.1. Summary

We have derived a fully general bias-variance-diversity-effect decomposition of the ensemble generalisation error from first principles. We argued that, among many of the proposed measures, diversity is *the* unifying theory. We were able to reframe various classical results on ensemble learning in the context of this decomposition. We have seen that many commonly used loss functions share common structure via the concept of Bregman divergences and have argued that, for Bregman divergences, ensembling cannot hurt performance if the combiner is defined in accordance with the divergence, i.e. as the left Bregman centroid. In 0/1-classification, we have considered three different conditions: the Weak Learner condition, 2-competence and k-competence – and have shown that each is a special case of the next. We were able to see that k-competence is exactly equivalent to non-negative diversity-effect (also known as ensemble improvement). In other words k-competent ensembles never perform worse than an average member. Further, we have seen that bounds originally derived under 2-competence also hold for k-competent ensembles.

The main contributions of this work are summarised in Section 1.2. In Section 6.2, we provide some ideas for future work. In Appendix A, we provide all experimental results. In Appendix B, we provide proofs for some statements that were not included in the main text to maintain brevity. Further, we provide some additional thoughts on impurity in decision tree construction.

6.2. Outlook

The role of diversity in learning problems We have seen that diversity is a central aspect of ensembles and determines the ensemble error in great part. However, it has become clear that a more diverse ensemble is not necessarily better on all tasks. In Section 5.2, we have considered this question under the light of the capacity of the base models. However, the characteristics of the underlying data are another variable that have not been considered at all. Two approaches seem interesting:

1. **Noise**: We have already seen that combining decision trees to a Random Forest can be understood to mitigate the tight fit of a tree to the training dataset. For instance, it has been observed that limiting the maximum depth has a regularisation effect and helps dealing with noisy data [49]. What is the effect of diversity in relationship to the signal-to-noise ratio?
2. **Adversarial Robustness**: It has already been observed that diverse neural network ensembles are more robust against adversarial attacks [50, 51]. However, these approaches do not consider a diversity decomposition and their measures of diversity are *ad-hoc* and disconnected from the ensemble error.

A diversity-aware splitting criterion Imagine growing M trees simultaneously as follows: In each iteration, one split is determined for each tree. Which node is being split is left to the choice of the splitting criterion. One could then attempt to derive a splitting criterion that optimises not only for a split that is pure (improves the prediction performance of the tree), but also one that implies predictions different

to those of other trees (improves diversity). Each split of a tree node yields a new forest partition $Z_1, \ldots, Z_P$. We have already seen that splitting criteria in standard Random Forests optimise a loss function. Hence, the objective function for a general splitting criterion for the i-th member could be written as $\ell(y, q_i)$, which is simply the member error. However, informed by the above discussion, one could define the splitting criterion in coordination to other ensemble members, i.e.

$$\ell(y, q_i) - \lambda \ell(\bar{q}, q_i)$$

which is directly analog to the NCL objective. In the NCL objective, however, this criterion is evaluated per point. In our case, it is evaluated per split, and (under some assumptions), only depends on previous splits in the forest and no extra sampling to determine the values.

Connection to theory of boosting methods The guided sequential training of member models as introduced in Section 5.3 can be viewed as gradient descent in function space. That is, by training the next member, we are searching for a function f_{i+1} such that the ensemble output $\bar{q} = \sum \alpha_i q_i + \alpha_{i+1} q_{i+1}$ is improved. This is the case exactly when q_{i+1} corresponds to a step in the direction of the (negative) gradient of the loss ensemble loss, i.e.

$$\max_q \langle -\nabla \bar{q}, q \rangle$$

In boosting methods for classification, q is obtained by setting adequate example weights. Likewise, the weights in the growth schemes presented herein correspond to a gradient step with respect to a loss function that resembles the NCL objective.

Here, α_i are weights. These can be chosen such that $\bar{q}$ is a mean; then this corresponds to the combination methods of Random Forests. Otherwise, particularly with decreasing weights α_i this is exactly the formulation of boosting ([23, 52]).

Functional derivatives The basic subject of analysis has been the function $q : X \to Y$ that corresponds to a member model. However, we have seen with the effect decomposition that it only ever appears in it's effect on the loss, i.e. as an argument to the loss function ℓ. In supervised learning, we are always concerned with finding a function q that minimises a loss or cost function ℓ. Framing boosting (or any ensemble construction procedure) as gradient descent in function space means that we are considering the gradient of $\ell(q(X))$ with respect to argument X. This means that a gradient descent step amounts to modifying the training data (weights in classification or pseudo-targets in boosting). However, what we really are looking for is a better *model*. One can understand ℓ as a functional, i.e. a function that maps a given function (the model q) to a scalar value (the model's expected loss). One could then use *functional derivatives* [53] to analyse questions such as "What function q will minimise $\ell(q)$?". Based on this idea, one can define *functional Bregman Divergences* [54] which considers Bregman divergences between *functionals*. It would be interesting to see whether the learning objectives can be expressed in these terms and whether this can provide any insight.

Diverse Bregman ensembles In this thesis, we have only considered growth strategies for classification ensembles under the 0/1-loss. The theory laid out in Section 4.3 on Bregman divergences and Section 5.4 on Negative Correlation Learning suggests that similar methods can be developed for regression tasks (this includes tasks where class probabilites are to be estimated). We have already seen hints that diversity behaves fundamentally different in this regime since the ensemble improvement can never be non-negative. We hypothesize that obtaining higher diversity and possibly also achieving lower ensemble error is easier in the regression regime due to the continuous nature of the outcome space.

Diverse Forests In general, diversity is defined in terms of model predictions. In Section 5.2 we suggest that, instead, diversity for Random Forests could be expressed in terms of *model parameters*, that is, the tree structures. Exploring this further seems worthwhile and it seems likely that diversity could be expressed purely in terms of model parameters.

Clustering trees We have established that ensemble diversity is measured in terms of diversity-effect for general loss functions or diversity for Bregman divergences. We have seen that this is simply a measure of variance around the ensemble combiner $\bar{q}$ as implied by the divergence $\mathrm{LE}(\bar{q}, q_i)$ or $B_\phi(\bar{q}, q_i)$. While this does not directly imply pairwise distances $\mathrm{LE}(q_i, q_j)$, this still gives us a way to think about distances between ensemble members that is deeply connected to the ensemble error.

For the squared-error, one can express the variance in terms of pairwise distances, see e.g. [55–57]. This is enabled by the square producing cross-terms. For other loss functions, it may be interesting to look for upper bounds (⇝Theorem 4.4.11).

Given an ensemble, can we be sure that really all members are essential to the ensemble performance? Could we find a subset of members with matching performance? This is *ensemble pruning* [20, 38, 58].

One interesting insight is that clustering strategies such as k-means require only distances to centroids. The k-means scheme can be generalised to Bregman divergences. Let I_ϕ be the Bregman information as defined in definition 1.6.3. Note that applied to ensemble members this is exactly the diversity. Let X be a random variable representing data points. Let C be a random variable taking values in $\mathcal{C}$ representing the set of cluster centroids. Then the objective for generalised k-means hard clustering [19] is to minimise the loss in Bregman information due to the quantisation induced by M:

$$\ell_\phi(C) =_{\mathrm{def}} I_\phi(X) - I_\phi(C)$$

One can show [19] that

$$\ell_\phi(C) = \mathbb{E}\left[I_\phi(X_k)\right] \approx \sum_{h=1}^{K} \sum_{x_i \in \mathcal{X}_k} v_i B_\phi(x_i, \mu_h)$$

where K is the number of clusters, $\mathcal{X}_k$ are the cells of the clustering, v_i is the distribution of the x_i and μ_k is the right Bregman centroid of $\mathcal{X}_k$. Classical k-means is a special case of this for the squared-error divergence. The KL-divergence implies the mutual information as a variance and yields *information-theoretic clustering*. The Ikura-Saito divergence yields the *LBG algorithm* [19]. Using this approach, could we cluster ensemble members and replace each cluster of members with its central representative member? The quantisation error $\ell_\phi(C)$ then directly measures the loss in diversity. This means that a good clustering corresponds to a diverse sub-ensemble.

This is in line with "overproduce-and-select" strategies commonly applied in ensemble pruning (see e.g. [38, 47]), which so far are based on heuristic measures and lack a clear theoretical relationship to the generalisation error.

Appendix

A. Full experiment results

Comparison to [46] Bernard, Adam, and Heutte evaluated only the following combined approach: They would perform both weighted bootstrapping and weighted tree construction. Further, weights would be determined only on out-of-bag-trees. Additionally, unrelated to the question at hand, they also employed a different way to determine candidate split features. Unlike in standard random forests, where a fixed number of candidate split features is sampled from all available features, the number of sampled candidate features was left fully random here. It was left unanswered which of these components actually affect the ensemble to what extent. Further, they did not provide any explanation or empirical analysis in terms of diversity. Looking at our results, the following points seem likely:

- An improvement in generalisation error is still obtained with standard candidate split feature sampling.
- The improvement can be explained using the notions of diversity.
- Weighted tree construction alone appears to have only very little effect as compared to a standard Random Forest.
- Weighted bootstrapping seems to provide the main effect.
- Determination of weights using out-of-bag-trees only does not improve performance for weighted tree construction.

Weighted tree contruction with out-of-bag weights does not appear to bring any advantage.

It may seem reasonable to expect that this [DRF] improves the average member error since the member to be trained puts more emphasis on "hard" examples and disregards "easy" examples, an idea very similar to boosting. This is also the motivation given in [46, 48]. However, across all experiments, we can observe that this is *not* the case, as can be seen e.g. in figures 5.4 and 5.5. This tradeoff is exactly reflected in the ambiguity decomposition.

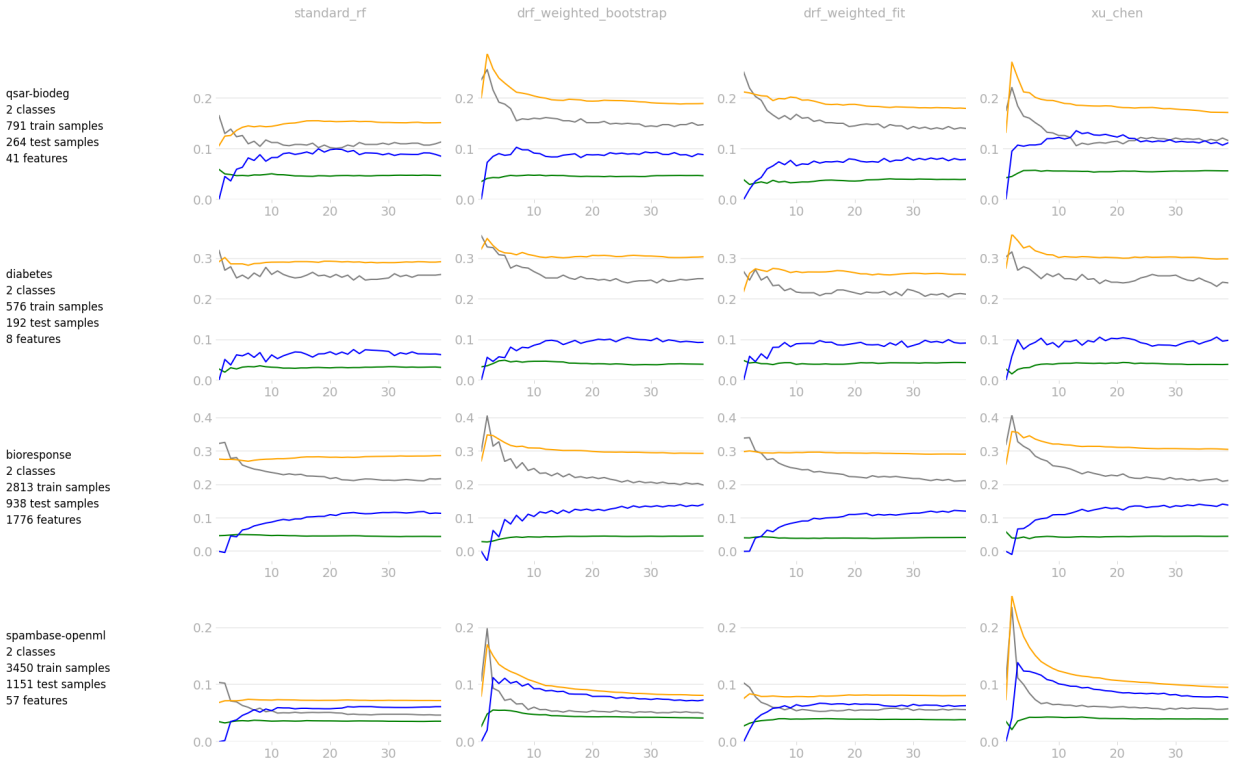

Figure A.1.: Full results for Subsection 5.3.1. (Legend: average member bias, average member variance, diversity, ensemble generalisation error.)

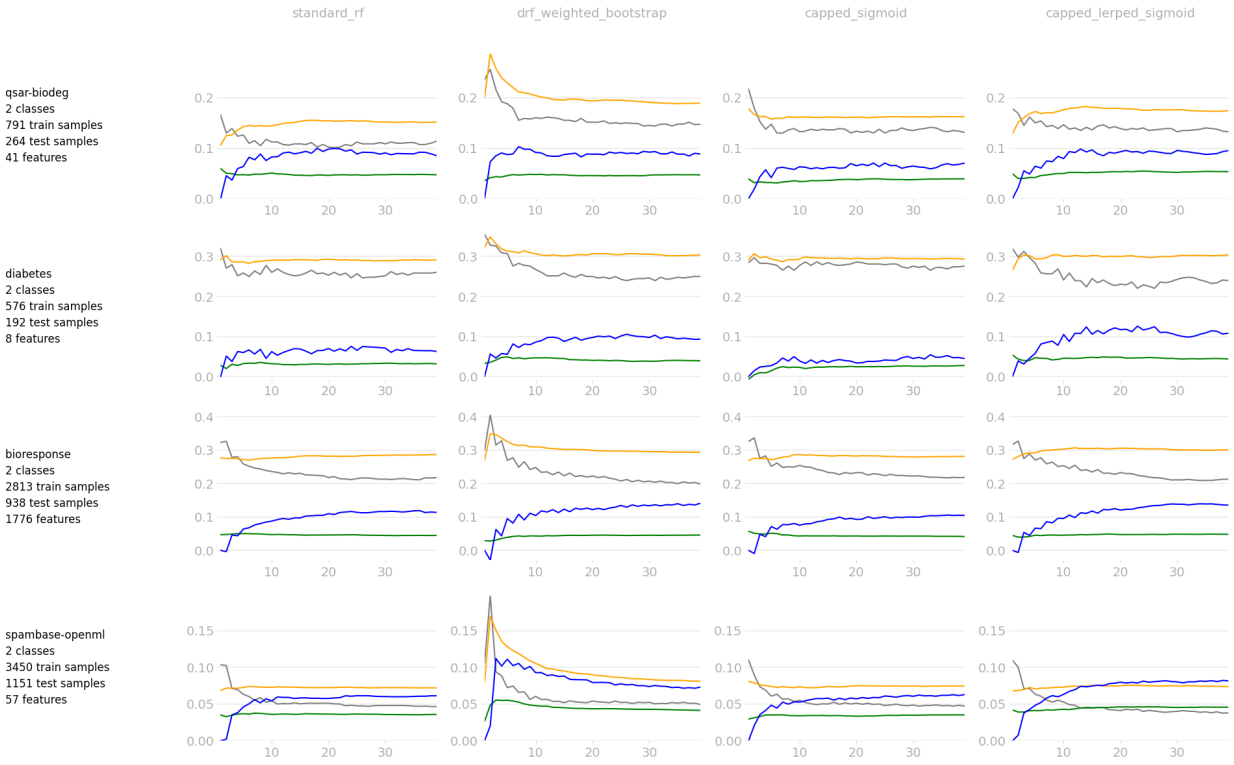

Figure A.2.: Full results for Subsection 5.3.2. (Legend: average member bias, average member variance, diversity, ensemble generalisation error.)

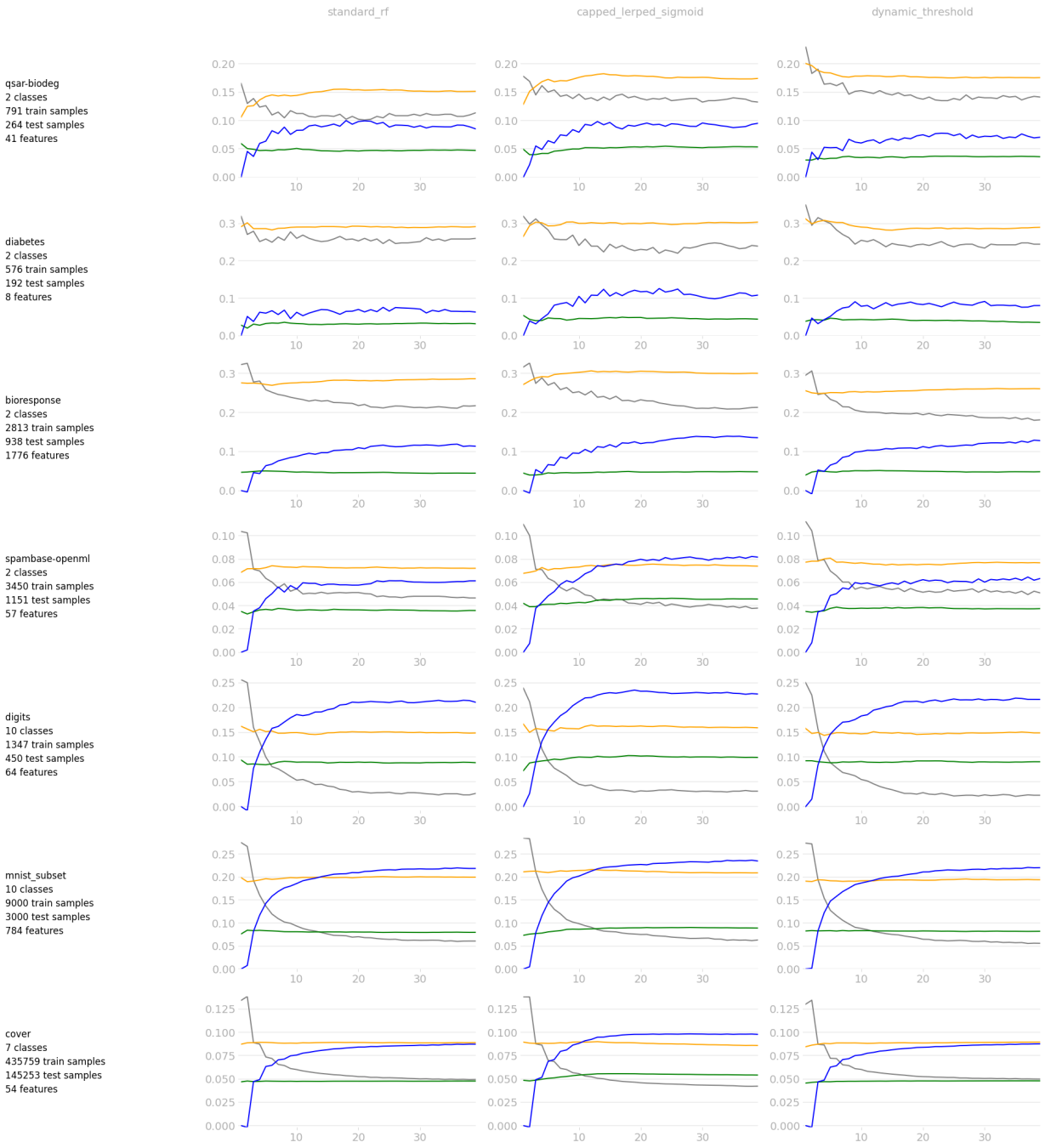

Figure A.3.: Full results for Subsection 5.3.3. (Legend: average member bias, average member variance, diversity, ensemble generalisation error.)

B. Proofs and additional results

B.0.1. Proof of Lemma 4.3.2

Lemma B.0.1 *(★, Generalised from [6]) Let q be a function of random variable Z and independent of Y. For $q^\star = \mathfrak{C}_Z[q]$, i.e. the left Bregman centroid w.r.t. Z, it holds that*

$$\mathbb{E}_Z\left[B_\phi\left(q^\star, q\right)\right] = \mathbb{E}_{Z,Y}\left[B_\phi\left(Y, q\right) - B_\phi\left(Y, q^\star\right)\right]$$

Proof.

$$\begin{aligned}
\mathbb{E}_{Z,Y}\left[B_\phi\left(y,q\right) - B_\phi\left(y,q^\star\right)\right] &= \mathbb{E}_{Z,Y}\left[(\phi(y) - \phi(q) - \langle\nabla\phi(q), y-q\rangle) - (\phi(y) - \phi(q^\star) - \langle y, y-q^\star\rangle)\right] \\
&= \mathbb{E}_{Z,Y}\left[\phi(q^\star) - \phi(q) - \langle\nabla\phi(q), y-q\rangle + \langle\nabla\phi(q^\star), y-q^\star\rangle\right] && q^\star, q \text{ indep. of Y} \\
&= \mathbb{E}_Z\left[\phi(q^\star) - \phi(q)\right] \\
&\quad - \mathbb{E}_{Z,Y}\left[\langle\nabla\phi(q), y\rangle\right] + \mathbb{E}_Y\left[\langle\nabla\phi(q^\star), y\rangle\right] \\
&\quad + \mathbb{E}_Z\left[\langle\nabla\phi(q), q\rangle - \langle\nabla\phi(q^\star), q^\star\rangle\right]
\end{aligned}$$

By definition of $q^\star$, $\mathbb{E}_Z\left[\nabla\phi(q^\star)\right] = \mathbb{E}_Z\left[\nabla\phi(\nabla\phi)^{-1}\mathbb{E}_Z\left[\nabla\phi(q)\right]\right] = \mathbb{E}_Z\left[\nabla\phi(q)\right]$. This allows us to apply the linearity of $\langle\cdot,\cdot\rangle$ twice to obtain

$$\begin{aligned}
\ldots &= \mathbb{E}_Z\left[\phi(q^\star) - \phi(q)\right] - 0 + \mathbb{E}_Z\left[\langle\nabla\phi(q), q - q^\star\rangle\right] \\
&= \mathbb{E}_Z\left[\phi(q^\star) - \phi(q) - \langle\nabla\phi(q), q^\star - q\rangle\right] \\
&= \mathbb{E}_Z\left[B_\phi\left(q^\star, q\right)\right]
\end{aligned}$$

□

B.0.2. Proof of Lemma 4.4.13

Lemma B.0.2 ★ *(Generalised from [7]) In k-competent ensembles it holds that*

$$\mathbb{P}\left[W \geq \kappa\right] \leq 2\mathbb{E}\left[W^2\right]$$

Proof. Note that

$$\mathbb{P}\left[W \geq \kappa\right] \leq 2\mathbb{E}\left[W^2\right] \;\leftrightarrow\; \mathbb{P}\left[W \geq \kappa\right] - 2\mathbb{E}\left[W^2\right] \geq 0$$

$$2\mathbb{E}\left[W^2\right] - \mathbb{P}\left[W \geq \kappa\right] = \mathbb{E}\left[(2W^2 - 1)\,\mathbb{1}\left[W \geq \kappa\right]\right]$$

We will aim to show that this above expression is nonnegative. The final inequality is due to applying lemma 4.4.10 to the second term.

$$\begin{aligned}
&\mathbb{E}\left[2W^2\right] - \mathbb{P}\left[W \geq \kappa\right] \\
&= \mathbb{E}\left[2W^2\right] - \mathbb{E}\left[\,\mathbb{1}\left[W \geq \kappa\right]\right] \\
&= \mathbb{E}\left[(2W^2 - 1)\,\mathbb{1}\left[W \geq \kappa\right]\right] + \mathbb{E}\left[2W^2\,\mathbb{1}\left[W < \kappa\right]\right] \\
&\geq \mathbb{E}\left[(2W^2 - 1)\,\mathbb{1}\left[W \geq \kappa\right]\right] + \mathbb{E}\left[2\overline{W}^2\,\mathbb{1}\left[\overline{W} < \kappa\right]\right]
\end{aligned}$$

$$\begin{aligned}
&W \geq \kappa \\
&\leftrightarrow\; 1 - W < 1 - \kappa \\
&\leftrightarrow\; \overline{W} \leq 1 - \kappa
\end{aligned}$$

Note that for $k \geq 2, \kappa > 1-\kappa$ and thus $\mathbb{E}\left[\mathbb{1}\left[\overline{W} \leq \kappa\right]\right] \geq \mathbb{E}\left[\mathbb{1}\left[\overline{W} \leq 1-\kappa\right]\right]$, allowing us to continue

$$\begin{aligned}
\ldots &\geq \mathbb{E}\left[(2W^2-1)\,\mathbb{1}\left[W \geq \kappa\right]\right] + \mathbb{E}\left[2\overline{W}^2\,\mathbb{1}\left[\overline{W} < 1-\kappa\right]\right] \\
&= \mathbb{E}\left[1 - 4\overline{W} + 2\overline{W}^2\,\mathbb{1}\left[\overline{W} < 1-\kappa\right]\right] \\
&\geq 0
\end{aligned}$$

□

B.0.3. More motivation for impurity measures

In this section, we give some additional arguments that motivate common impurity measures. The motivation is somewhat less rigorous than the one provided in Section 3.1, but it might be more intuitive and is closer to how impurity measures for decision trees are commonly introduced in teaching materials [8].

Consider a parent node P that, due to some split, was partitioned into the disjoint union $L \dot\cup R$. Let y_P, y_L and y_R be the output values of the parent and the two new leaf nodes, produced by the leaf aggregation function. Since the leaf output is constant over a single cell, the gain in loss due to a split is the difference between the loss of the parent node and the sum of losses of the two individual child nodes. For brevity, we write $\ell_P(y) =_{\text{def}} \sum_{i\in P} \ell(y, y_i)$.

$$\begin{aligned}
\textit{Loss Gain:} \quad & \sum_{i\in P} \ell(y_P, y_i) - \left(\sum_{i\in L} \ell(y_L, y_i) + \sum_{i\in R} \ell(y_R, y_i)\right) \\
&= \ell_P(y_P) - (\ell_L(y_L) + \ell_R(y_R))
\end{aligned}$$

In order for a split to yield positive loss gain, the leaf aggregation function needs to be such that the loss does not increase as constraints are removed, i.e. the set of considered examples is reduced. Recall that $\bar z$ is a centroid with respect to a loss function ℓ and a set of outcomes P if and only if $\bar z = \arg\min_z \sum_{i\in P} \ell(z, y_i) = \arg\min_z \ell_P(z)$ (1.6.1).

Lemma B.0.3 *For a loss function ℓ, if the leaf aggregator of a node P is $y_P =_{def} \arg\min_z \sum_{i\in P} \ell(z, y_i)$, i.e. the centroid with respect to ℓ, the loss gain is nonnegative.*

Proof. Let $\ell_P(y) =_{\text{def}} \sum_{i\in P} \ell(y, y_i)$. Since $P = L\dot\cup R$, we need to show that $\ell_P(y_P) = \ell_L(y_P) + \ell_R(y_P) \geq \ell_L(y_L) + \ell_R(y_R)$. Assume $\ell_L(y_P) < \ell_L(y_L)$. This contradicts the definition of y_L as the minimizer, and as such $\ell_L(y_P) \geq \ell_L(y_L)$. Likewise, we can conclude that $\ell_R(y_P) \geq \ell_R(y_R)$. Combining the two inequalities yields the statement. □

This rigorously motivates the specific choice of leaf aggregation function. The majority vote is a centroid with respect to the 0/1-loss for classification, while the arithmetic mean is the centroid with respect to the squared error loss for regression.

In the best possible case, all training examples in a given cell correspond to the same (classification) or very similar (regression) outcomes. If a query point then falls within that cell, i.e. it has similar features, one can say with high confidence that the query point should have the same (similar) outcome. In the spirit of greedy optimisation, we

aim to split cells such that the resulting child cells are more *pure* with respect to their outcomes.

The notion of local purity is linked to the training error: If a leaf cell is perfectly pure, all training examples in that cell correspond to the same outcome. Hence, the leaf aggregation function, which us usually implemented as some kind of mean, will produce exactly that outcome for any query point that belongs to this cell, in particular any training points. Consequently, for a suitable definition of "error", perfectly pure cells have zero training error.

Consider a split, parameterised by Θ, that partitions a parent node P into the disjoint union $L_\Theta \dot\cup R_\Theta$. Let n, n_L, n_R be the cardinalities of the parent and the two child nodes. Let H be an impurity measure. We will select the split that yields the lowest impurity.

$$\arg\min_\Theta \ \frac{n_L}{n} H(L_\Theta) + \frac{n_R}{n} H(R_\Theta)$$

The gain in purity is then the difference between impurities before and after the split.

$$\textit{Purity Gain}: \quad H(P) - \left(\frac{n_L}{n} H(L_\Theta) + \frac{n_R}{n} H(R_\Theta)\right) \tag{B.1}$$

Note that this is not necessarily the same as the gain in *loss* achieved due to a split.

P is the parent node before the split with combiner y_P. L and R are the two child nodes after a split with combiners y_L and y_R.

Variance Reduction

A commonly used impurity measure for regression is the squared-error variance.

$$H_{\text{var}}(P) =_{\text{def}} \frac{1}{n_P} \sum_{i \in P} (y_i - y_P)^2 \qquad \text{for} \quad y_P =_{\text{def}} \frac{1}{n_p} \sum_{i \in P} y_i$$

To motivate this impurity measure, it remains to be shown that a split guided by this impurity measure actually reduces the value of a specific loss function and which one that is. Luckily, it is easy to see that this holds for the squared error loss. Plugging the definition into the purity gain (eq. (B.1)) yields

$$H(P) = \frac{1}{n_P} \sum_{i \in P} (y_i - y_P)^2 = \frac{1}{n_P} \underbrace{\sum_{i \in L} (y_i - y_P)^2}_{\ell_L(y_P)} + \frac{1}{n_P} \underbrace{\sum_{i \in R} (y_i - y_P)^2}_{\ell_R(y_P)}$$

and

$$\frac{n_L}{n_P} H(L) + \frac{n_R}{n_P} H(R) = \frac{n_L}{n_P} \frac{1}{n_L} \underbrace{\sum_{i \in L} (y_i - y_L)^2}_{\ell_L(y_L)} + \frac{n_R}{n_P} \frac{1}{n_R} \underbrace{\sum_{i \in R} (y_i - y_R)^2}_{\ell_R(y_R)}$$

By lemma B.0.3, we can directly conclude that the loss gain is positive for any split if the arithmetic mean is used as a leaf combiner.

Gini impurity

We will argue that the Gini impurity split criterion, which finds a split such that the Gini impurity is reduced, maximises the classification margins.

Definition B.0.1 *(Classifier margins [8]) The margin for class k of an example X is the difference between the model's confidence that X is of class k and the next-best class:*

$$m(x, y) =_{def} \mathbb{P}[y|x] - \max_{j \neq y} \mathbb{P}[j|x]$$

- *The vector $m(x) = [m_1(x), \ldots, m_K(x)]^\top$, where K is the total number of classes, is called a margin vector iff its components sum to zero.*
- *For a pair (X, y) of example and true outcome, the model's prediction is correct iff $m_y(x) > 0$*

Proposition B.0.4 ⋆ *Minimising the Gini impurity is equivalent to maximising the classification margins as measured by the squared error.*

$$\min H_{Gini} \equiv \max \sum_k (p_k - u_k)^2$$

where $u =_{def} [1/k, \ldots, 1/k]$

Proof. Let p be a probability distribution and $u =_{\text{def}} [1/k, \ldots, 1/k]$ a uniform distribution. Then

$$\sum_k (p_k - u_k)^2 = \sum_k p_k^2 - 2 \sum_k p_k u_k + \sum_k u_k^2$$

$\sum_k u_k^2$ does not depend on p. Further, because p is a probability distribution and sums to one, $2 \sum_k p_k u_k$ is constant. In summary

$$\max \sum_k (p_k - u_k)^2 \equiv \max \sum_k p_k^2$$

Let us now consider the Gini impurity. It holds that

$$H_{\text{Gini}} = \sum_k p_k(1 - p_k) = 1 - \sum_k p_k^2$$

and consequently

$$\min \sum_k p_k(1 - p_k) \equiv \max \sum_k p_k^2$$

□

A common approach in training classification models is *margin maximisation* [23] in which we aim to maximise the margin of the true label $m_y(X)$. A margin loss function $\ell : \mathbb{R} \to \mathbb{R}$ is a *margin-maximising* loss if $\ell'(m_y(X)) \leq 0$ for all values of m_y [59]. A decision tree can also be evaluated based on a margin loss. The empirical error of a decision tree node P with respect to a margin loss ℓ can be written as $L(P) = \frac{1}{|P|} \sum_{i \in P} \ell\left(m_y(x_i)\right)$ where $m_y(x_i)$ is the value of the true margin. Then the following holds [59].

$$\begin{aligned} L(P) &= \frac{1}{|P|} \sum_{i \in P} \sum_k \mathbb{1}[y_i = k] \cdot \ell\left(m_k(x_i)\right) \\ &= \sum_k \frac{1}{|P|} \sum_{i \in P} \mathbb{1}[y_i = k] \cdot \ell(m_k(x_i)) \\ &= \sum_k p_k(x_i) \ell(m_k(x_i)) \end{aligned}$$

An example for a margin-maximising loss function is the *hinge loss* defined as $\ell(m_y(x)) =_{\text{def}} \max\{0, 1 - p\}$. Its subderivative is

$$\frac{\partial \ell}{\partial p} = \begin{cases} -1 & p \leq 1 \\ 0 & \text{else} \end{cases}$$

and hence it is a margin-maximising loss.

In summary, the Gini impurity splitting criterion greedily optimises classification margins and thus margin-maximising losses.

Proposition B.0.5 ⋆ *Let p be a probability distribution and u an arbitrary vector. Let $G = \sum_k p_k(1-p_k)$ be the Gini impurity. Then $-G$ is the generator for the Bregman divergence*

$$B_{-G}(p,u) = \sum_k (p_k - u_k)^2$$

Proof. Let $\phi(q) =_{\text{def}} (-1)\sum_k p_k(1-p_k)$ as assumed.

$$\begin{aligned} B_{-G}(p,u) &= \underbrace{(-1)\sum_k p_k(1-p_k)}_{\phi(p)} \; - \; \underbrace{(-1)\sum_k u_k(1-u_k)}_{\phi(u)} \; - \; \underbrace{\sum_k (2u_k-1)(p_k-u_k)}_{\langle \nabla\phi(u), p-u\rangle} \\ &= \sum_k (p_k - u_k)^2 \end{aligned}$$

the first equality follows by definition of a Bregman divergence (see 1.6.1) and the second equality by arithmetic. □

C. Implementation

Ensemble error decompositions were computed using the *decompose* library [*]. Learners were implemented using *scikit-learn* [†] and *numpy* [‡]. Experiments were managed using DVC [§]. The full source code and configuration can be obtained from the GitHub repository.

Compared datasets We give a brief motivation for the classification datasets we have selected for evaluation. *cover* is a dataset with a relatively high number of examples and low feature dimensionality. *mnist-subset* is a dataset with a moderate number of examples and high dimensionality. *diabetes* is a dataset with relatively high error rates. *bioresponse* is a small dataset with a very high number of features ($d \approx 1/2n$). *qsar-biodeg* is a small dataset used for quick testing. Further, [46, 48] evaluated on *mnist* (although not just a subset of it), *spambase*, *digits* and *diabetes* .

Approximating statistical quantities For a dataset with n examples, $n_{\text{train}} =_{\text{def}} \frac{3}{4}n$ examples were assigned to be part of the *training split*, the other n_{test} for the *testing split*. Examples in the testing split are used for evaluation only and were never used in training a model. If X is a random variable taking values in the space of examples $\mathcal{X}$, expectations over X are approximated as the arithmetic mean over given examples in the testing split, i.e. for a function g:

$$\mathbb{E}_X\left[g(X)\right] \approx \sum_{i=1}^{n_{\text{test}}} g(x_i)$$

If D is a random variable corresponding to the input to a learner, for instance the training dataset or randomness in the learning algorithm, an expectation over D is approximated by an arithmetic mean over results of a fixed number of trials. In our case, we performed 3 trials.

[*] https://github.com/EchoStatements/Decompose/ and [10]
[†] https://scikit-learn.org/
[‡] https://numpy.org/
[§] https://dvc.org

Bibliography

[1] Gérard Biau and Erwan Scornet. 'A Random Forest Guided Tour'. In: *TEST* 25.2 (June 1, 2016), pp. 197–227. DOI: 10.1007/s11749-016-0481-7. (Visited on 12/13/2022) (cited on page 1).

[2] Pengyi Yang et al. 'A Review of Ensemble Methods in Bioinformatics: Including Stability of Feature Selection and Ensemble Feature Selection Methods'. In: *Current Bioinformatics* 5.4 (Dec. 1, 2010), pp. 296–308. DOI: 10.2174/157489310794072508. (Visited on 11/18/2023) (cited on pages 1, 20).

[3] Anders Krogh and Jesper Vedelsby. 'Neural Network Ensembles, Cross Validation, and Active Learning'. In: (1995) (cited on pages 1, 23, 24).

[4] N. Ueda and R. Nakano. 'Generalization Error of Ensemble Estimators'. In: *Proceedings of International Conference on Neural Networks (ICNN'96)*. International Conference on Neural Networks (ICNN'96). Vol. 1. Washington, DC, USA: IEEE, 1996, pp. 90–95. DOI: 10.1109/ICNN.1996.548872. (Visited on 04/13/2023) (cited on pages 1, 25).

[5] Ludmila I Kuncheva. 'Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy'. In: (2003) (cited on pages 1, 24, 39).

[6] Danny Wood et al. *A Unified Theory of Diversity in Ensemble Learning*. Jan. 10, 2023. URL: http://arxiv.org/abs/2301.03962 (visited on 04/30/2023). preprint (cited on pages 1–3, 7, 9, 27, 29, 30, 33, 59).

[7] Ryan Theisen et al. *When Are Ensembles Really Effective?* May 20, 2023. DOI: 10.48550/arXiv.2305.12313. URL: http://arxiv.org/abs/2305.12313 (visited on 06/30/2023). preprint (cited on pages 1, 3, 8, 27, 31, 33–38, 40, 41, 59).

[8] Sami Tibshirani, Harry Friedman, and Trevor Hastie. 'The Elements of Statistical Learning'. In: (2017), p. 764 (cited on pages 7, 8, 49, 60, 62).

[9] J.V. Hansen and T. Heskes. 'General Bias/Variance Decomposition with Target Independent Variance of Error Functions Derived from the Exponential Family of Distributions'. In: *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000* 2 (2000), pp. 207–210. DOI: 10.1109/ICPR.2000.906049. (Visited on 07/25/2023) (cited on page 7).

[10] Danny Wood, Tingting Mu, and Gavin Brown. *Bias-Variance Decompositions for Margin Losses*. Apr. 26, 2022. DOI: 10.48550/arXiv.2204.12155. URL: http://arxiv.org/abs/2204.12155 (visited on 05/10/2023). preprint (cited on pages 7, 33, 64).

[11] Luca Didaci, Giorgio Fumera, and Fabio Roli. 'Diversity in Classifier Ensembles: Fertile Concept or Dead End?' In: *Multiple Classifier Systems*. Ed. by Zhi-Hua Zhou, Fabio Roli, and Josef Kittler. Red. by David Hutchison et al. Vol. 7872. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 37–48. DOI: 10.1007/978-3-642-38067-9_4. (Visited on 07/24/2023) (cited on pages 7, 25).

[12] Pedro Domingos. 'A Unified Bias-Variance Decomposition'. In: () (cited on page 7).

[13] David Pfau. 'A Generalized Bias-Variance Decomposition for Bregman Divergences'. In: () (cited on pages 7, 9, 28).

[14] Gareth James and Trevor Hastie. 'Generalizations of the Bias/Variance Decomposition for Prediction Error'. In: () (cited on page 8).

[15] Erwan Scornet, Gérard Biau, and Jean-Philippe Vert. 'Consistency of Random Forests'. In: *The Annals of Statistics* 43.4 (Aug. 1, 2015). DOI: 10.1214/15-AOS1321. (Visited on 12/01/2022) (cited on pages 8, 20, 41).

[16] Andrew M. Webb et al. 'To Ensemble or Not Ensemble: When Does End-To-End Training Fail?' 2019. DOI: 10.13140/RG.2.2.28091.46880. (Visited on 10/14/2023) (cited on pages 8, 24, 50, 51).

[17] Kilian Weinberger. *Lecture 12: Bias Variance Tradeoff*. URL: https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote12.html (visited on 10/17/2023) (cited on page 8).

[18] Ben Adlam et al. *Understanding the Bias-Variance Tradeoff of Bregman Divergences*. Feb. 9, 2022. URL: http://arxiv.org/abs/2202.04167 (visited on 09/15/2023). preprint (cited on page 9).

[19] Arindam Banerjee et al. 'Clustering with Bregman Divergences'. In: *Proceedings of the 2004 SIAM International Conference on Data Mining*. Proceedings of the 2004 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, Apr. 22, 2004, pp. 234–245. DOI: 10.1137/1.9781611972740.22. (Visited on 10/15/2023) (cited on pages 9, 10, 54).

[20] Zhi-Hua Zhou. *Ensemble Methods: Foundations and Algorithms*. 0th ed. Chapman and Hall/CRC, June 6, 2012. (Visited on 03/24/2023) (cited on pages 11, 23, 54).

[21] Ibomoiye Domor Mienye and Yanxia Sun. 'A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects'. In: *IEEE Access* 10 (2022), pp. 99129–99149. DOI: 10.1109/ACCESS.2022.3207287. (Visited on 07/10/2023) (cited on page 11).

[22] Leo Breiman. 'Random Forests'. In: *Machine Learning* 45.1 (Oct. 1, 2001), pp. 5–32. DOI: 10.1023/A:1010933404324. (Visited on 03/29/2023) (cited on pages 11, 13–15, 20, 42).

[23] Robert E. Schapire and Yoav Freund. *Boosting: Foundations and Algorithms*. Adaptive Computation and Machine Learning Series. Cambridge, MA: MIT Press, 2012. 526 pp. (cited on pages 11, 53, 62).

[24] Gilles Louppe. *Understanding Random Forests: From Theory to Practice*. June 3, 2015. URL: http://arxiv.org/abs/1407.7502 (visited on 06/21/2023). preprint (cited on pages 12–14).

[25] Taiga Abe et al. 'Pathologies of Predictive Diversity in Deep Ensembles'. Version 2. In: (2023). DOI: 10.48550/ARXIV.2302.00704. (Visited on 09/04/2023) (cited on pages 13, 30, 39, 40).

[26] Leo Breiman. *Classification and Regression Trees*. New York: Routledge, Oct. 25, 2017. 368 pp. (cited on page 20).

[27] Sebastian Buschjäger and Katharina Morik. *There Is No Double-Descent in Random Forests*. Nov. 8, 2021. DOI: 10.48550/arXiv.2111.04409. URL: http://arxiv.org/abs/2111.04409 (visited on 04/21/2023). preprint (cited on pages 20, 39, 41, 47, 51).

[28] Sebastian Buschjäger, Lukas Pfahler, and Katharina Morik. *Generalized Negative Correlation Learning for Deep Ensembling*. Dec. 9, 2020. URL: http://arxiv.org/abs/2011.02952 (visited on 05/03/2023). preprint (cited on pages 20, 25, 42, 50).

[29] Ron Kohavi and David H Wolpert. 'Bias Plus Variance Decomposition for Zero-One Loss Functions'. In: () (cited on page 24).

[30] Pádraig Cunningham and John Carney. 'Diversity versus Quality in Classification Ensembles Based on Feature Selection'. In: *Machine Learning: ECML 2000*. Ed. by Ramon López de Mántaras and Enric Plaza. Red. by Jaime G. Carbonell et al. Vol. 1810. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 109–116. DOI: 10.1007/3-540-45164-1_12. (Visited on 11/19/2023) (cited on page 24).

[31] Catherine A. Shipp and Ludmila I. Kuncheva. 'Relationships between Combination Methods and Measures of Diversity in Combining Classifiers'. In: *Information Fusion* 3.2 (June 2002), pp. 135–148. DOI: 10.1016/S1566-2535(02)00051-9. (Visited on 11/19/2023) (cited on page 24).

[32] Gavin Brown et al. 'Managing Diversity in Regression Ensembles'. In: (2005) (cited on pages 25, 26, 49, 51).

[33] Taiga Abe et al. 'The Best Deep Ensembles Sacrifice Predictive Diversity'. In: (2022) (cited on pages 27, 39–41).

[34] Gavin Brown and Ludmila I. Kuncheva. '"Good" and "Bad" Diversity in Majority Vote Ensembles'. In: *Multiple Classifier Systems*. Ed. by Neamat El Gayar, Josef Kittler, and Fabio Roli. Red. by David Hutchison et al. Vol. 5997. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 124–133. DOI: 10.1007/978-3-642-12127-2_13. (Visited on 04/27/2023) (cited on pages 31, 32).

[35] Cha Zhang and Yunqian Ma, eds. *Ensemble Machine Learning: Methods and Applications*. New York, NY: Springer New York, 2012. (Visited on 09/27/2023) (cited on page 39).

[36] Prem Melville and Raymond J. Mooney. 'Creating Diversity in Ensembles Using Artificial Data'. In: *Information Fusion* 6.1 (Mar. 2005), pp. 99–111. DOI: 10.1016/j.inffus.2004.04.001. (Visited on 06/29/2023) (cited on page 39).

[37] Archana R. Panhalkar and Dharmpal D. Doye. 'A Novel Approach to Build Accurate and Diverse Decision Tree Forest'. In: *Evolutionary Intelligence* 15.1 (Mar. 2022), pp. 439–453. DOI: 10.1007/s12065-020-00519-0. (Visited on 07/25/2023) (cited on pages 39, 42).

[38] Souad Taleb Zouggar and Abdelkader Adla. 'Simplifying Random Forests Using Diversity'. In: 18 (2019) (cited on pages 39, 42, 54).

[39] Md Nasim Adnan and M. Islam. 'Complement Random Forest'. In: Australasian Data Mining Conference. 2015. (Visited on 07/25/2023) (cited on pages 39, 40).

[40] Y Liu and X Yao. 'Ensemble Learning via Negative Correlation'. In: *Neural Networks* (1999) (cited on pages 39, 40, 49, 51).

[41] Wenjing Li, Randy C. Paffenroth, and David Berthiaume. *Neural Network Ensembles: Theory, Training, and the Importance of Explicit Diversity*. Sept. 28, 2021. URL: http://arxiv.org/abs/2109.14117 (visited on 06/28/2023). preprint (cited on pages 39, 40).

[42] Chun Yang and Xu-Cheng Yin. 'Diversity-Based Random Forests with Sample Weight Learning'. In: *Cognitive Computation* 11.5 (Oct. 1, 2019), pp. 685–696. DOI: 10.1007/s12559-019-09652-0. (Visited on 09/08/2023) (cited on page 42).

[43] Prashant Gupta et al. 'Guided Random Forest and Its Application to Data Approximation'. In: *ArXiv* (Sept. 2, 2019). (Visited on 09/08/2023) (cited on page 42).

[44] M. A. H. Akhand, M. M. Hafizur Rahman, and K. Murase. 'Decision Tree Ensemble Construction Incorporating Feature Values Modification and Random Subspace Method'. In: *2014 International Conference on Informatics, Electronics & Vision (ICIEV)* (May 2014), pp. 1–6. DOI: 10.1109/ICIEV.2014.6850822. (Visited on 07/25/2023) (cited on page 42).

[45] Md Nasim Adnan and Md Zahidul Islam. 'Effects of Dynamic Subspacing in Random Forest'. In: *Advanced Data Mining and Applications*. Ed. by Gao Cong et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, pp. 303–312. DOI: 10.1007/978-3-319-69179-4_21 (cited on page 42).

[46] Simon Bernard, Sébastien Adam, and Laurent Heutte. 'Dynamic Random Forests'. In: *Pattern Recognition Letters* 33.12 (Sept. 1, 2012), pp. 1580–1586. DOI: 10.1016/j.patrec.2012.04.003. (Visited on 04/21/2023) (cited on pages 42, 43, 56, 64).

[47] Simon Bernard, Laurent Heutte, and Sebastien Adam. 'On the Selection of Decision Trees in Random Forests'. In: *2009 International Joint Conference on Neural Networks*. 2009 International Joint Conference on Neural Networks (IJCNN 2009 - Atlanta). Atlanta, Ga, USA: IEEE, June 2009, pp. 302–307. DOI: 10.1109/IJCNN.2009.5178693. (Visited on 07/18/2023) (cited on pages 42, 47, 54).

[48] Xiaolong Xu and Wen Chen. 'Implementation and Performance Optimization of Dynamic Random Forest'. In: *2017 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*. 2017 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC). Oct. 2017, pp. 283–289. DOI: 10.1109/CyberC.2017.53 (cited on pages 43, 56, 64).

[49] Siyu Zhou and Lucas Mentch. 'Trees, Forests, Chickens, and Eggs: When and Why to Prune Trees in a Random Forest'. In: *Statistical Analysis and Data Mining: The ASA Data Science Journal* 16.1 (2023), pp. 45–64. DOI: 10.1002/sam.11594. (Visited on 06/23/2023) (cited on page 52).

[50] Tianyu Pang et al. *Improving Adversarial Robustness via Promoting Ensemble Diversity*. May 29, 2019. URL: http://arxiv.org/abs/1901.08846 (visited on 11/22/2023). preprint (cited on page 52).

[51] Takuma Amada et al. 'Adversarial Robustness for Face Recognition: How to Introduce Ensemble Diversity among Feature Extractors?' In: () (cited on page 52).

[52] Llew Mason et al. 'Boosting Algorithms as Gradient Descent'. In: () (cited on page 53).

[53] Bela A Frigyik, Santosh Srivastava, and Maya R Gupta. 'An Introduction to Functional Derivatives'. In: () (cited on page 53).

[54] BÉla A. Frigyik, Santosh Srivastava, and Maya R. Gupta. 'Functional Bregman Divergence and Bayesian Estimation of Distributions'. In: *IEEE Transactions on Information Theory* 54.11 (Nov. 2008), pp. 5130–5139. DOI: 10.1109/TIT.2008.929943. (Visited on 11/22/2023) (cited on page 53).

[55] Siddharth Shakya. *What Is K Means Objective Function?* Cross Validated. June 4, 2018. URL: https://stats.stackexchange.com/q/349709/178468 (visited on 11/24/2023) (cited on page 54).

[56] Daniel. *Relation between Pairwise Distance Sum and Sum of Distance to Mean (Gap Statistic)*. Cross Validated. Mar. 21, 2019. URL: https://stats.stackexchange.com/q/398635/178468 (visited on 11/24/2023) (cited on page 54).

[57] ttnphns. *Link between Variance and Pairwise Distances within a Variable*. Cross Validated. Dec. 1, 2015. URL: https://stats.stackexchange.com/q/20108/178468 (visited on 11/24/2023) (cited on page 54).

[58] Fan Yang et al. 'Margin Optimization Based Pruning for Random Forest'. In: *Neurocomputing* 94 (Oct. 2012), pp. 54–63. DOI: 10.1016/j.neucom.2012.04.007. (Visited on 09/08/2023) (cited on page 54).

[59] Christian Leistner et al. 'Semi-Supervised Random Forests'. In: *2009 IEEE 12th International Conference on Computer Vision*. 2009 IEEE 12th International Conference on Computer Vision (ICCV). Kyoto: IEEE, Sept. 2009, pp. 506–513. DOI: 10.1109/ICCV.2009.5459198. (Visited on 05/11/2023) (cited on page 62).