Machine learning for ecosystem assessment in the Amazon, using environmental DNA data

In collaboration with NatureMetrics

Imperial College London

Adamos Spanashis

CID01059045

Supervisor: Dr Ben Calderhead

Dr Oliver Ratmann

Department of Mathematics

Imperial College London

This dissertation is submitted for the degree of

Master of Science in Statistics

September 2019

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

Adamos Spanashis
CID01059045
September 2019

Acknowledgements

I would like to thank NatureMetrics and my supervisor Dr Ben Calderhead for giving me the opportunity to work on this problem and to learn so much about ecology and genomics.

I would also like to thank Dr Oliver Ratmann for taking up the role of supervisor on such a short notice and for giving me valuable advice for writing my thesis.

Finally I would like to thank my friend Kendeas Theofanous with whom I had lengthy conversations on the subject and whose insight on machine learning I found valuable.

Abstract

In this project we show that the application of supervised machine learning methods, for the purpose of classification, on metabarcoding eDNA data is feasible. Samples of water were collected and filtered from Amazonian rivers of Peru by WWF using NatureMetric Kits. The eDNA found in the samples were sequenced using high-throughput sequencing and clustered into OTUs using metabarcodes. We use CSS normalisation, several ordination methods, and other dimensionality reduction techniques to create features out of the OTU table produced from sequencing. Several classifiers trained on these feature sets are used to infer the water colour the samples come from. We outline the workings of both Bayesian and Maximum likelihood Logistic Regression, and of Random Forest. Ways to split the data into train-validation-test sets are devised based on the samples' location; the splits are used to evaluate the classifiers' performance under different scenarios. We find that all of them perform much better, conditional on the features set used, than a naive benchmark score that mimics a weighted 'coin flip'. It is proposed that extensive sampling efforts, spatially and temporally, together with machine learning methods and time series analysis can be used to uncover complex interactions between the species in a community that otherwise go unnoticed when using classical ecological tools.

Table of contents

Chapter 1

Background

1.1 Motivation

Increased human populations during the industrialisation of the 19th century were accompanied by increasing amounts of unmanaged waste that resulted in public health problems. The first attempts at remedying the issues where applied mostly to running waters, and had a bacteriological focus [7]. As time passed, managing freshwater systems became more important and evolved into a more complex procedure that took into account entire aquatic communities (like macro-invertebrates and fish) rather than specific species. Animals inhabiting the system studied were used as indicators of sources of pollution that could no have been identified by chemical analysis. Thus, the link between environmental management and biological monitoring came as response to the needs of human populations.

Nearing the end of the 20th century, "ecological health" became a priority in some human societies; people begun pressuring public authorities to restore freshwater systems to a healthier state. This is also reflected in the political sphere with the rise of Green parties in more economically developed nations across the world. Subsequently, huge budgets have been allocated in the management of freshwater, and other, systems; an example being the restoration of the Emscher river system which has an estimated cost of US$5.5billion [18].

The driving motive today for the assessment of environmental consequences (from plans, policies or projects) around the world is coming from regional legislation, operations of Non-Governmental Organisations and requirements set by the financial backers of projects in the area. Legal procedures, policies and instruments are set up to ensure that decision makers take into consideration the environmental impacts of their projects; examples include the Environmental Impact Assessment Directive of the European Union [19] and the Environmental Protection Agency in the United States [50], both of which became operational around the 1970s.

Environmental studies conducted all around the world test sites such as fish farms (for example in New Zealand, Scotland, Norway and others) [47], rivers that cross various landscapes, land (used for dairy-farming, horticulture, or where different kinds of forests grow)[24] and many others. A good way to infer the health of an environment is by investigating the species inhabiting it, and in particular their relative abundance[1]; some species are very intolerant of pollution (Alderfly Larva) whereas others are tolerant (Leeches, Blood Worms). Thus, the distribution of individuals in the various species (or other higher taxonomic groups) can tell us a lot about the levels of pollution.

The method of using species as indicators to survey the health of an environment is called biomonitoring. The discipline's aim is to find the ideal bioindicators whose presence or behaviour reflects best a stressor's effect on biota. As an example, in rivers, the quality of water can be assessed by examining macro-invertebrates, fishes [30], and bacterial communities [47] found at different sites. In land, soil bacterial community composition can be used to infer soil condition and health [24].

The traditional methods of biomonitoring involve a limited, long-scaled site sampling of individual organisms which are then processed and sorted into sample taxonomic units. This process can take months to years, and usually produces data of low information [4]. Analysis of ecosystems requires taxonomic expertise across many order and several phyla; species-level identification is hindered by problems arising from co-ordinating the inputs of several experts, and differences in taxonomic refinement. As a result, the identified taxa are often very few in numbers, and are usually the ones deemed as critical (indicator organisms), by experts, for the specific study[14].

The resolution of identification (or 'taxonomic penetration') stops, more often that not, to taxonomically higher categories (genus, family etc) than species. The reasons for the reduction of penetration are usually not made explicit; most often a more pragmatic approach is taken which seeks to determine individuals to the species level only if the ease of doing it and the time taken permits it [14]. As a result, most of the species which are more difficult to identify are lumped together to larger categories, loosing information in the process. This is especially the case with specious groups of freshwater organisms that occupy the lower levels of the food web, even though they constitute most of the biodiversity in the System and thus have the greatest potential for response to stressors [5]. For example, lumping together species of the Chironomidae genus, because of the difficulty in separating them, reduces the sensitivity of the biomonitoring scheme used [43].

[1] Species abundance is the number of individuals comprising a species in a particular area. Relative abundance is how individuals in a community are distributed among species.

1.2 Genomics: A new hope?

The morpho-taxonomic[2] identification of species has been the limiting step in biomonitoring efforts because of the short-supply of taxonomists and prohibitive costs in separating and identifying species. The invention in 1977 of Sanger-based DNA sequencing[3] which revolutionised all branches of the biological sciences, could not be used for environmental bulk samples because they contained potentially thousands of species, and separating them for sequencing was prohibitively difficult.

However, DNA sequence-based analysis has enabled ecologists to answer questions they would not have been able to without such data. In particular with the advent of DNA barcoding in 2004 [23], which is a technique of identifying species based on short DNA sequences, international efforts have been made to build a taxonomic reference library (Barcode of Life Initiative), and identify unknown speciments to the species-level by comparing their sequence to known ones already catalogued in a reference database [45].

The emergence of Next-generation sequencing (NGS) platforms brought significant improvements in DNA sequencing technologies [46]. The new platforms can deliver billions of sequence reads per single run, which is orders of magnitude better than traditional Sanger sequencing. As a result there has been a significant drop of sequencing costs per megabases that has been going over the last 2 decades. In particular, the cost of sequencing 1 megabase has gone from $\$10^4$ in 2001, to $\$10^2 - 10^3$ in 2007 and finally to less than $\$10^{-1}$ in 2019 [15]. This, coupled with advancements in DNA- and RNA-based techniques in taxonomic identification [4] have made possible the application of metagenetics, the study of genetic material sourced directly from the environment, in ecological studies.

Normal barcoding standards were designed with the purpose of identifying isolated specimens from intact DNA using Sanger sequencing, so are inapplicable to empirical ecological studies where the samples contain DNA from a mixture of related species [48]. To solve this problem DNA metabarcoding was developed and made possible by NGS techologies. It is a method for high-throughput multi species identification using degraded DNA found in the environment (eDNA) [48]. The method relies on barcode genes which mutate at a rate that makes them stable within a species but different when compared to other ones, and which are used for the purpose of species identification. Examples are the 16S rRNA and the Cytochrome Oxidase 1 [23] genes.

[2]Taxonomic assignment based only on the morphology of the organism. This involves only its form and structure (appearance), but not its functions.

[3]Sequencing is the process by which the order of the Nucleotides (or four bases) in a sample of DNA are found.

After the DNA is extracted from an environmental sample, its barcode region needs to be amplified (multiplied millions of times) before it can be sequenced. This involves selecting the right primers [4] for the targeted taxonomic groups and using PCR for the amplification. The amplified DNA is then sequenced on a high-throughput sequencing platform and the sequences are processed and classified into Operational Taxonomic Units (OTUs).

OTUs are an intentionally vague term used to cluster sequences produced from metabarcoding. Reads with a predetermined percentage of similarity (usually 97%) between them are classified into the same OTU. Thus they are acting as a proxy to traditional 'species'. The most abundant sequence of each OTU is assigned a taxonomy using reference databases. Most of the times, even when cross referencing the OTUs with a taxonomic library, a species-level identification is not available; errors in sequencing reads and the arbitrary cut-off similarity percentile are among some of the factors that prevent identification. However, this might not necessarily be a downside of the metabarcoding approach. OTUs can still act as informative indicators if they respond to environmental gradients and have characteristic signatures. These developments allowed scientists to overcome the bottleneck of morpho-taxonomic identification of species.

An advantage of NGS platforms is that they can sequence DNA from multiple environmental samples, each made of potentially hundreds of species. The sequences clustered into an OTU, per sample, are counted and registered in an sample-OTU table. This can mean, for example, that in the first sample collected from the environment, 30 sequences belonging to OTU1 were read, 0 belonging to OTU2, 20435 in OTU3 and so on until all OTUs are considered. OTUs can be completely taxonomically identified (up to the species level), partly (up to a higher level) or not at all.

1.3 Data

Our data were sampled from the Northern Peruvian rivers in collaboration with WWF Peru. They comprise of a sample-OTU table constructed from water samples collected along the rivers' length, and metadata provided by WWF Peru. The metadata include information on the location, water colour, area of the river, trip number, date of sampling, and details of the sample for some of them.

[4] Short molecules which provide the starting point for DNA amplification (in other words specify the region to be multiplied)

1.3.1 Geography

The Northern Peruvian rivers of interest are made up of the Maranon River on the west, which is joined by its tributary, the Huallaga River. Together they join with the Ucayali River to form the Amazon River, which runs across South America to the Atlantic Ocean. In addition to these rivers, the tributaries of Napo and Tapiche were sampled.

The Peruvian rivers' confluence has the largest annual water discharge rate into the Amazon, making it the mainstream source. The Maranon river flows from the Andes Jungle and mountains, through Pongo de Manseriche, a gorge (narrow steep valley of hills or mountains with a river flowing through it) northwest of Peru. The Pongo is series of torrents, interspersed with rocks, and at points reaches a width of no more than 25m. It acts like a natural barrier between the Upper Part of the Maranon river and the rest of the area, making the fauna upstream potentially different from downstream. For the purposes of the project, the Maranon River has been split into three areas: Upper, Mid and Lower, with the Upper being behind the Pongo.

From the river samples collected, some came from white and others from black waters. White water rivers appear so because of suspended sediment. They have a higher concentration of minerals (especially sodium, potassium, magnesium, calcium) than Black rivers and have a neutral pH, compared to the acidic of Black waters. The dark colour on the other hand comes from tannins leaching from decaying vegetation. These differences have important implications for the rivers' fauna, since some species cannot live in environments with low concentrations of particular minerals. Thus it is expected that different animal communities will be found in these different environments.

A plot of the samples collected along the rivers can be seen in Figure 1.1. The different colours represent the rivers, and the shapes the water colour. There is a clear imbalance in the numbers of black and white water samples and also in their spatial distribution. From the 164 samples collected, 143 are white and 21 black. All of the black water samples are found in the Eastern part of the river, and along Ucayali, Tapiche, Napo and Maranon lower.

1.3.2 Sampling and Preprocessing

The water samples were collected from the sides of a boat using kits provided by NatureMetrics. A volume of water, between 0.5-2L is filtered through a membrane which is then send to the laboratory for DNA extraction. From each site in the river (boat stop), 4 samples were collected (with some sites having a bit more). Information about the sites and samples was also recorded in the metadata (like ID, location, water colour, trip number and date of collection).

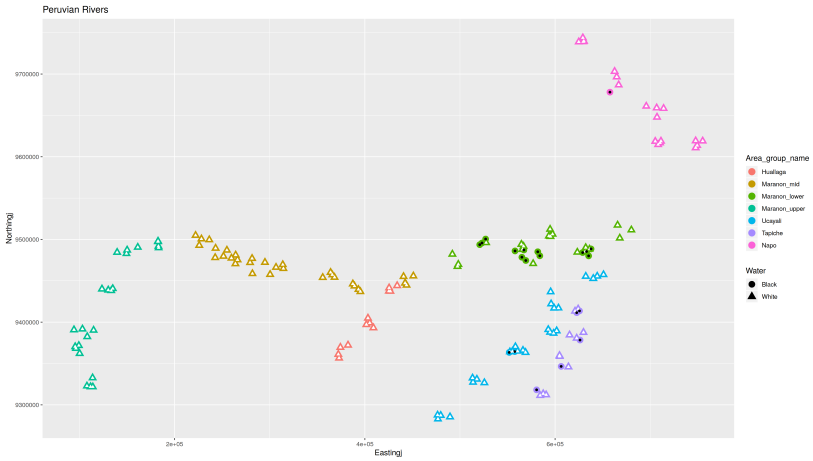
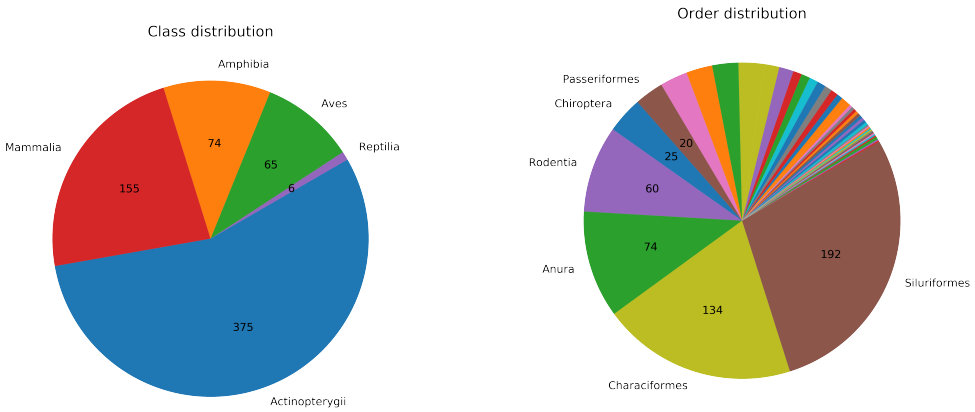


Fig. 1.1 A plot of the Easting and Northing coordinates of the samples collected. Colours represent the rivers, and shapes the water colour (triangle for White and circle for Black). The points are also coloured white and black in the middle to aid in viewing. White noise with a standard deviation of $5 \cdot 10^3$ was added to the coordinates so as to separate points close together.

At the laboratory, DNA trapped inside the membranes is extracted and then PCR amplified. The DNA fragments produced are then sequenced using high-throughput sequencing technology that can handle multiple samples at once. The raw reads from the sequencing are processed (filtered and assembled) and then clustered into OTUs with a similarity cut-off point of 99%. The representative sequences, or the most abundant individual sequences per OTU, were used for taxonomic assignment. After removing the non identified OTUs we were left with 675 of them.

Five taxonomic Classes were kept for our analysis: Actinopterygii (fish), Amphibia, Aves (birds), Mammalia, and Reptilia. The most instances of OTUs found belonged to the Actinopterygii class, and within that, Siluriformes and Characiformes are the most abundant Orders. Class and Order distributions of OTUs can be seen in Figure 1.2. The sample-OTU table produced was very sparse (meaning that a significant number of entries were 0), as is the case for most studies using metabarcoding techniques.

The sampling methods employed to produce our data come with some inherent biases. First of all, species give out different amounts, sizes and kinds of material behind that can be used to identify them. Furthermore, environmental DNA extracted from the samples can be PCR amplified at different rates for species. This can happen because of primer mismatch with the organisms barcode DNA region. This results in OTU reads that do not correlate well with the actual abundance of organisms in the environment and also to uninformative comparisons between OTUs' reads [16].



(a) Distribution of OTUs into Classes. Most abundant one is Actinopterygii.

(b) Distribution of OTUs into Orders. Most abundant ones are Siluriformes and Characiformes

Fig. 1.2

Finally, the OTU table sparsity causes the reads to be concentrated on some samples only, with others having a much lower total read count (sum over the reads of all OTUs in a sample). There are 7 samples (all from white water parts of the river) with under 10000 total reads, of which 3 have less than 50. On the other hand, the sample with the largest total read counts has 219113 reads.

1.4 Literature Review

The advent of metagenetics has opened up the gateway for data driven research in multiple fields. From cataloguing micorbial genes in the human gut [1], to ecological assessment of freshwater [3] and river systems [11] using eDNA metabarcoding. Ecological monitoring has traditionally been done using the morpho-taxonomic identification of species in the environment studied. This involves identifying an organism up to a certain level and then using their presence in the site to come to conclusions about the health of the system.

Biotic indices are used to encapsulate information about the abundance of species identified in a site, and also indicate the degree to which a site is healthy. They are highly specialised in what type of pollution they can quantify and which species they consider as important [51]. To calculate the value of a biotic index, the species found in a site are assigned a weight (or tolerance value) provided by the index and defined from empirical

and experimental data [8]. Weights signify how susceptible an organism is to the pollution studied [10]. Then an analytic formula uses the species weights to calculate the index's value which indicates the environmental quality of the system (usually in categories ranging from 'very poor' to 'very good').

As mentioned previously, the morpho-taxonomic identification of species is time consuming, expensive and limited in scope. Instead, a metabarcoding approach can generate an OTU table with assigned taxonomy from reference databases in significantly less time. The OTU reads can then be used to calculate biotic indices and evaluate the system's environmental health [29]. Due to a lack of taxonomic resolution and limited reference libraries, most metabarcoding data are not used in the calculation of indices. However, taxonomy-free approaches can be used to calculate proxies of biotic indices that have similar evaluation performance [3].

One such approach to calculating biotic indices is through the use of supervised machine learning (SML). The first time that it was employed for biomonitoring surveys was in 2015 by Smith et al [34] (using microbial eDNA) and in 2017 by Cordier et al [12] (using eukaryotes eDNA). Cordier et al trained two SML models (Random Forests and Self Organising Map) to infer several Biotic Indices used often for marine studies. The features were OTUs reads obtained from foroamfinera eDNA sampled from the benthic zone (sediment on the bottom of the river) and processed in a similar way to the one outlined in the previous section. A notable difference is that unassigned OTUs (without taxonomy) were kept and used as features. Furthermore, the authors constructed new feature data sets using standard ecological techniques such as rarefying and alpha diversity metrics. Target values (biotic indices) were calculated using morpho-taxonomic data and associated weights.

The SML models performed better in predicting the Biotic index of a site than a reference model that used the taxon assignments of OTUs and a correlation approach for assigning them weights. Also notable is that the SML models had a very high degree of agreement with the morpho-taxonomic evaluation of habitat quality. Furthermore, discarding low abundance OTUs (low total read counts across samples) did not affect the models' performance. The researchers followed up their paper with a more comprehensive one in 2018 [13], which tested their SML methods on OTUs derived from 5 different ribosomal bacterial and eukaryotic markers[5]. They found that there was no significant difference in the models' performance when using different markers, and that for all of them, SML outperformed all taxonomy-based eDNA biomonitoring methods.

[5]These markers are specifically designed to amplify regions of the genome that would be used in the identification of species. Multiple exist because of the different regions in the DNA needing to be amplified for the identification of specific groups of organisms (like eukaryotes and bacteria).

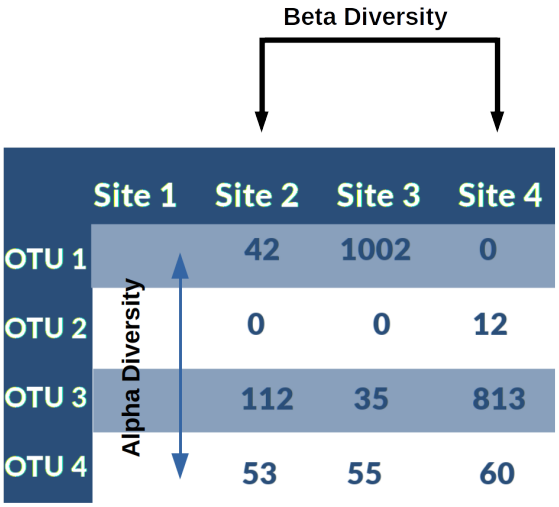


Fig. 1.3 An illustration of Alpha and Beta diversity metrics as calculated from OTU tables

It is very rare that an analysis of metabarcoding (OTU reads per sample) or metagenetic (genes per sample) data involves machine learning approaches. Researchers usually analyse alpha (within) and beta (between) diversity metrics of samples (see Figure 1.3), explore the patterns in beta diversity using ordination techniques, and perform various classical hypothesis tests.

Alpha diversity metrics are a measure of the diversity a sample displays within itself (with respect to the OTUs' reads). An example can be OTU richness which measures the number of OTUs present in the sample (without taking into account the reads) or Shannon index which converts the reads into probabilities (by dividing them with the sample's total reads) and calculates the sample's entropy. The formula is given by:

$$-\sum_{i=1}^{s} p_i \ln(p_i),$$

were s are the number of species present in the sample, and p_i is the read count of the ith OTU divided by the total read count of all OTUs in the sample.

Beta diversity measures how different samples are in terms of their species composition. They usually take the form of (dis)similarity measures, and some of the more common ones used are Bray-Curtis, Chao and Jaccard (to be explained later). The output of these measures is a symmetric matrix whose elements quantify how (dis)similar the samples are. Ordination methods can then be applied to this matrix that explore its structure graphically. Environmental variables, like pH, concentrations of minerals, or pollutants, can be fitted on

top of these plots and help researchers uncover patterns in OTU composition that explain the variables' variation.

If there is a discernible pattern in the ordination plots of the data that separates them along a gradient or grouping of a variable, then differential abundance analysis can be used to investigate it further. For example, if samples are separated into two groups that also happen to coincide with a categorical variable of the data, a model can be fit to test which OTUs are differentially abundant across the grouping that causes the separation. Parametric models include Zero-Inflated Gaussian mixture model, Zero-inflated Log-Normal mixture model [39], and other generalised linear models (like overdispersed Poisson model) [42]. The choice of distributions to model read counts is highly dependent upon the problem and also very debatable [39, 53, 31]; if their assumptions about the data are not met they can yield a high level of false negatives/positives.

Non-parametric tests exist that test if groupings of samples (that divides them into two or more groups) result in populations that have significantly different distributions. Such tests include the Mann-Whitney test and permutational multivariate analysis of variance (PERMANOVA). To use such tools the data have to be transformed in some way first, either using alpha or beta diversity.

1.5 Aim

The aims of this project are to explore the data set obtained from eDNA metabarcoding of samples collected from Peruvian rivers, and use the OTU table to predict the water colour of samples. Data processing techniques like feature selection through correlation and normalisation of the read-counts are presented in Chapter 2. These modifications to the OTU table are used as new features for the classification of samples. Their spatial distribution along the rivers is taken into account when designing train-validation-test splits; the emulate different conditions under which the models' weaknesses and strengths can be uncovered. These are outlined and explained in the Chapter.

Ordination methods are used with a variety of distance measures to identify patterns in the data. These methods can also be used for dimensionality reduction, and thus more features are created for classification. An introduction to the most popular ordination methods is presented in Section 3.1. Also, we prove that Principal coordinate analysis is a general case of Principal components analysis.

Permutation Analysis of variance is presented in section 3.2. This classical ecological test hypothesis method is used to evaluate if the grouping by water colour divides the samples into populations with significantly different distributions.

Supervised machine learning models are trained to predict water colour of samples using a variety of features derived from OTUs. An outline of Bayesian and maximum likelihood logistic regression, as well as of Random forests is presented in section 3.3. The Hamiltonian Monte Carlo sampling algorithm is introduced in section 3.4 and is used for the Bayesian approach to Logistic Regression.

These models are then cross-validated and tested under the schemes devised previously; their performance is compared for all the features created. Furthermore, the taxonomic Orders with the most explanatory power are identified.

Chapter 2

Data Processing and Splitting

2.1 Processing

In this Chapter, we explore in what ways we can process our data before using them for classification and how we might test our classifiers while taking into account the spatial distribution of the samples. Datasets obtained through metabarcoding or metagenetic methods usually display high variability in read counts between and within samples that stems from systematic biases. Several normalisation techniques have been developed to deal with this issue; we present some of them and will evaluate their usefulness in classification. Furthermore, inspired by the interdependent nature of species in animal communities, we use correlations between features for feature selection.

The samples' location in the river surely introduces some dependencies between them; animals tend to wander around their habitat, and thus eDNA collected from sites close together will be more similar than those far apart. The way our data is split for training, validating and testing will influence our classifiers' performance. We present two methods of splitting them, using their location as the deciding variable, that will test the classifiers' ability to extrapolate information gained through training. Furthermore, we outline the cross-validation procedure to be used for the evaluation.

2.1.1 Normalisation

Count data from amplicon sequencing display a very high degree of variability in total read counts per sample [32]. A histogram, Figure 2.1, of the samples' total count reads for our data exemplifies this variability. The sums range from 17 to 219113, with a median and mean of 63672 and 77152 respectively. This much variation between samples makes it hard to

identify which OTUs' difference in abundance between samples is significant and also might negatively impact the performance of the classifiers.

The increased variation comes from a systematic variability affecting multiple samples and OTUs in a similar manner. Sources of such variability can be the inconsistencies in DNA extraction and handling of samples, a varying quality of sequencing runs [41], and other PCR-specific amplification biases (like primer mismatch, GC-content etc.) [16, 25].

Furthermore, these biases can cause the distribution of read counts obtained from high-throughput amplicon sequencing to diverge significantly from the actual distribution of species abundances inhabiting in the same regions where the samples were collected. The Pearson correlation between read counts and actual species frequencies is close to zero [17].

The removal of this systematic variability is called *Normalisation* and it's original aim in the bioinformatics literature is to increase the statistical power and false positive rates of differential abundance analysis [1]. We will be testing if normalisation methods have any effect on the classifiers' scores.

A normalisation method that produced promising results in differential analysis on datasets similar to ours is Cumulative sum scaling (CSS). This method is an extension to the quantile normalisation approach which divides read counts by the Qth percentile of each sample's non-zero count distribution. CSS determines that percentile using a data-driven approach [39].

To illustrate how normalisation methods work we define $X_{i,j}$ as the read counts of sample $i = 1,...,n$ and OTU $j = 1,...,p$. The normalisation factor of the total sum scaling (TSS) method is found by summing the read counts in a sample i:

$$N_i = \sum_{j=1}^{p} X_{ij}. \tag{2.1}$$

Then the counts in row i of the read matrix X are divided by the factor N_i. TSS is the most commonly used method of normalisation but it has been shown to introduce biases in differential analysis estimates [9].

Quantile scaling computes the normalisation factor by taking into account how OTU total counts (for all samples) vary, and choosing a percentile that produces desirable properties

[1] Differential abundance analysis aims at finding OTUs (or genes in the case of metagenomic studies) whose variation between groups of samples (e.g. black and white water river samples) is statistically significant.

(such as robustness from highly abundant OTUs). The scaling factor is defined as:

$$N_i = Q\text{th quantile}_{j \in G} X_{ij} \tag{2.2}$$

$$G = \left\{ j : \sum_{i=1}^{n} X_{ij} > 0 \right\}. \tag{2.3}$$

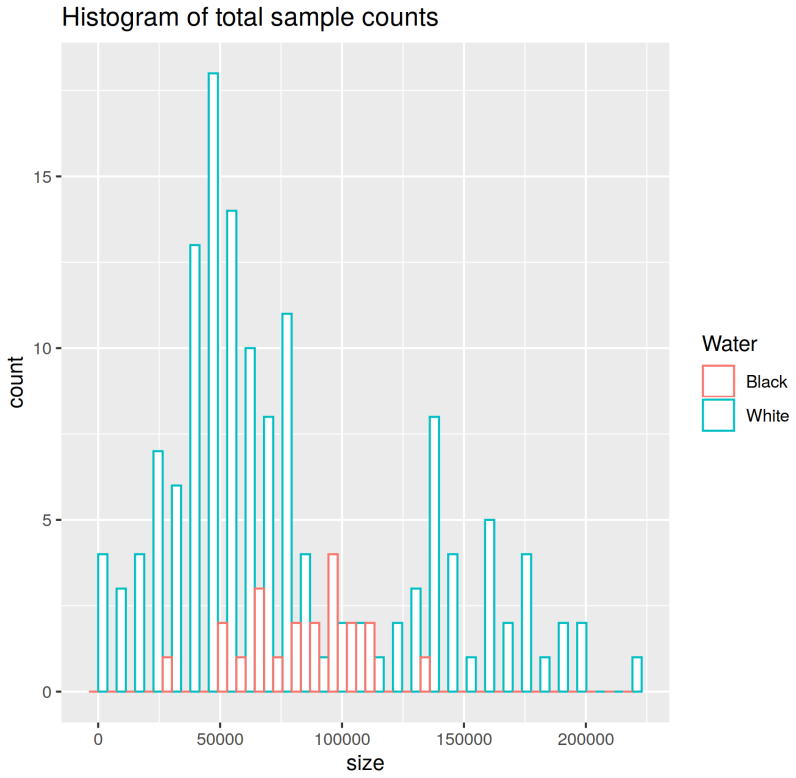


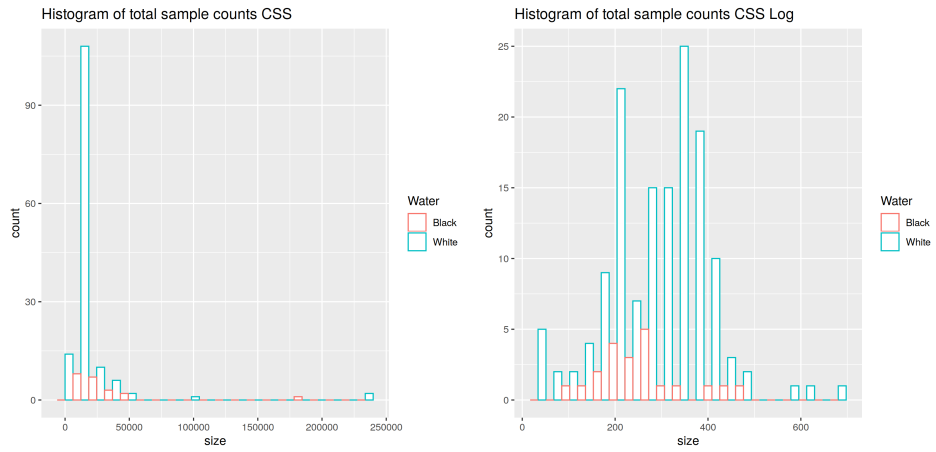
Fig. 2.1 Histogram of the sum of OTU counts of each sample. The cyan colour represents white water samples and the red colour black water samples.

CSS normalisation involves the calculation of a data-driven quantile We define the lth quantile of sample i as q_i^l, which means that l OTUs have a read count lower than q_i^l. Also we define the sum of counts per samples i up to the lth quantile as

$$s_i^l = \sum_{j | X_{ij} \leq q_i^l} X_{ij}. \tag{2.4}$$

With this notation, the total sum normalising factor (2.1) is given by $N_i = s_i^p$ (where p is the total number of OTUs). CSS chooses a value $\hat{l} \leq p$ using a data driven approach to calculate the scaling factor ($N_i = s_i^{\hat{l}}$) for each sample and get normalised counts. In particular, the

quantile $\hat{l}$ for the threshold $q_i^{\hat{l}}$ is chosen to be the smallest value where the median absolute deviation of sample-specific quantiles q_i^l from a reference point (the median quantile q_i^l across all samples) shows high instability. [39].



(a) (b)

Fig. 2.2 Histogram of the read counts per sample after 2.2a CSS normalisation and with 2.2b a log_2 transformation. The cyan colour represents white water samples and the red colour black water samples.

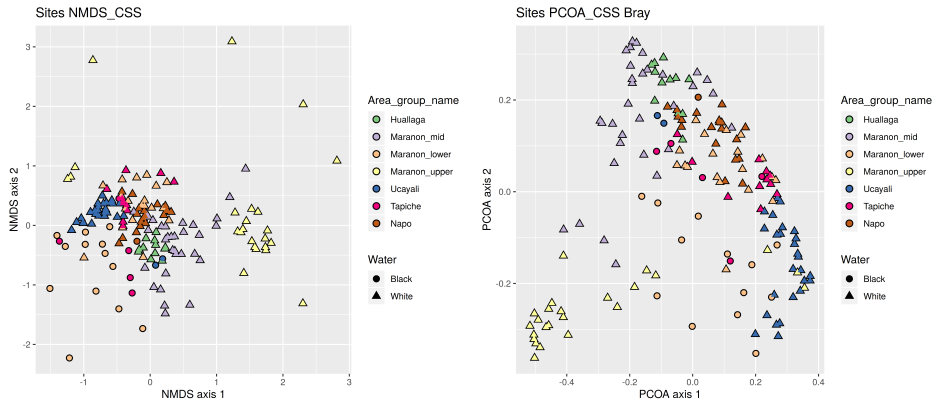
Using CSS normalisation on our data causes the variation of read counts per sample to drop significantly (except for some samples whose total read counts increase, see Figure 2.2a). Applying a further log_2 transformation to the data after normalising reduces the variation even more (see Figure 2.2b). The transformation does not apply the logarithm to 0 valued read counts. Furthermore, using Principal coordinate analysis and Non-metric multidimensional scaling (Figure 2.3b and 2.3a respectively) ordination on the normalised data separates white and black water samples better than if the normalisation was not applied[2]. However, doing the same on the log_2 transformed data does not have the same effect.

We will be checking if this normalisation method and transformation produces any significant improvements (over the unnormalised dataset) in the classifiers ability to determine the river colour.

2.1.2 Feature Correlation

A big minority of OTUs in our data set have been found to have an absolute Spearman correlation of more than 0.9 with at least one other OTU. This correlation has a biological

[2]These methods are similar to Principal Components Analysis in that they reduce the dimensions of the data. An outline of their use and how they are performed will be presented in the next Chapter.



(a) (b)

Fig. 2.3 Ordination plots for the CSS normalised data. Figure 2.3a shows an NMDS and Figure 2.3b a PCoA plot of the data using the Bray-Curtis measure. Both methods separate white and black water samples better than when applied to non-normalised data.

underpinning as it is expected that the abundance of species in a river environment would co-vary along it's length. Relationships between species in an environment can take many forms, like parasitic (the abundance of OTU1 is increased and for OTU2 is decreased when they are both present), competitive (the abundance of both OTU1 and OTU2 is decreased when they are both present), and mutual (the abundance of both OTU1 and OTU2 is increased when they are both present). More complicated, non-linear correlation networks exist in nature between more than two species that are very difficult to capture even with large amounts of unbiased data [52].

For our problem at hand, recovering such correlation networks is not of interest. We are more interested in selecting the best features from our data that would aid in the classification. Thus, we used the Spearman correlation with a threshold of 0.9 to remove 122 OTUs and create a new data set without highly correlated OTUs.

2.2 Splitting

Together with the OTU table, our data also include the location of the samples in the rivers (in Easting and Northing coordinates) and in which part of the river they belong to. Because of this location attribute, our samples cannot be said to be independent. Thus, the way we choose to split our data into training and testing sets will surely affect the accuracy of the classifier. For example, testing on a set that is composed of samples maximally distant from the ones in the training set (see Figure 2.5) will produce different results than testing on a set with all samples in close proximity to the ones in the training set (see Figure 2.4).

To avoid choosing a split method, several ones are employed that represent different splitting conditions. The classifiers are then tested on all of them so as to evaluate how well they can perform under various circumstances.

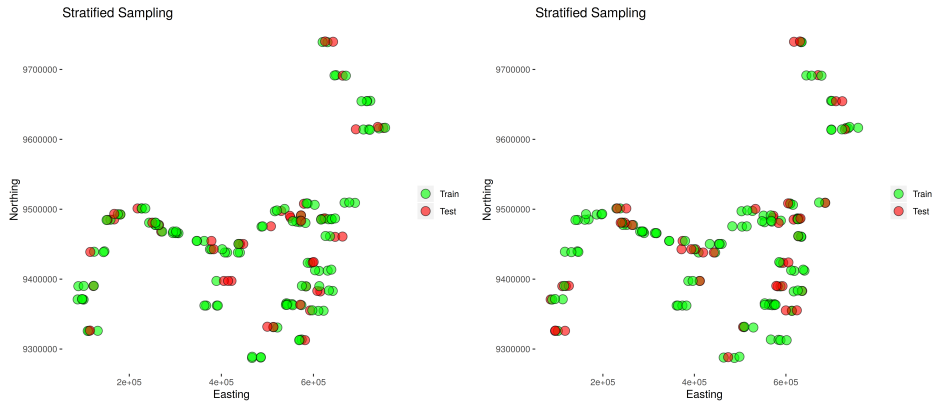


Fig. 2.4 Test set samples are selected such that they are located among the Train set. This represents a maximally similar split which is created by ensuring constant distribution of samples in each part of the river in both the Train and Test set.

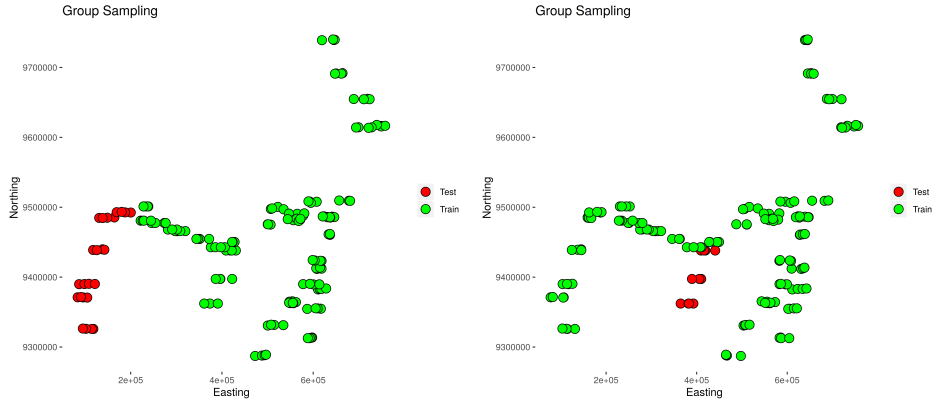


Fig. 2.5 Test set samples are geographically distinct from Train set samples. This represents a maximally dissimilar split which is created by choosing a different part of the river as the Testing set.

In addition to splitting the data into training and test sets, the train set is further split into validation sets so as to tune the hyperparameters of the models. The splitting into validation sets follows the same principle/method as the train-test split. So if, for example, the train and test sets are split using the maximally similar approach, the validation sets created from the train set are chosen so as to be maximally similar to the remaining set.

The steps of testing a classifier using a specific split method are as follows:

1. A set of hyperparameters values are chosen for the cross-validation procedure. For example, in Logistic Regression a set of numbers ranging from 0.001 to 20 is chosen for the sparsity parameter and the set $\{True, False\}$ is chosen for the intercept parameter.

2. The data set is split into train and test sets based on the splitting method of choice.

3. The train set is split into K-folds, each being a validation set, using the same method as for the train-test split.

4. The classifier is trained on the $K-1$ folds and tested on the remaining validation set for all possible combinations of hyperparameters (e.g. the Logistic Regressor is trained with hyperparameters $\{0.001, True\}, \{0.001, False\}, \{1.001, True\}, \{1.001, False\}$ etc...).

5. Step 4 is repeated for all the K-folds (by training on the $K-1$ and testing on the one left out). The average score across the validation sets is found for each hyperparameter combination and the classifier is retrained on the train set using the best hyperparameters set.

6. The retrained model is tested on the test set and the confusion matrix is calculated.

7. Steps 2 through 6 are repeated a number of times for different train-test splits using the same split principles. The confusion matrix for each repetition is stored and added at the end of the process.

The best hyperparameters are the ones that maximise either the F score or accuracy. The former combines both the precision and recall of the validation test. Precision is the number of correctly identified positive results divided by the number of all positive results returned by the classifier. Recall is the number of correctly identified positive results divided by the true number of positive results. The score is the harmonic mean of the two

$$F_1 = \frac{2}{\text{recall}^{-1} + \text{precision}^{-1}}. \tag{2.5}$$

Choosing black water as the positive result would lead to cases were recall is undefined; in the case of maximum dissimilarity, a validation set might end up with only white water samples (thus only negatives). So recall, which measures how many positive observations were correctly identified from the total number of positive observations, would be $\frac{0}{0}$. Thus, white samples are chosen as the positive observations.

The whole testing procedure is repeated using accuracy as the metric for picking the best hyperparameters. This is because the F score might lead to choosing classifiers that

Table 2.1 Features used in Classification

Features used	Description of dataset
OTU	The OTU table as it is
OTU LOW	The OTU table without highly correlated features
OTU CSS	The CSS normalised OTU table
OTU Min CSS	The CSS normalised OTU table without samples with total read counts of less than 10000 reads
OTU CSS LOG	A $\log_2$ transformed OTU CSS
PCoA Bray-Curtis	The transformation of the OTU table with PCoA using the Bray-Curtis measure
PCoA Bray-Curtis CSS	The transformation of the OTU CSS table with PCoA using the Bray-Curtis measure

minimise False Negatives (maximising recall), and as such, have an increased concentration of mistakes in identifying black water samples.

This procedure can be repeated for different features (data sets), so as to evaluate how feature selection and transformations can alter the results. The various data sets used to test the classifiers on are summarised in table 2.1.

A baseline benchmark for this problem is also set so that the classifiers' evaluation is meaningful. We are essentially setting the 'coin flip'/naive rate; prediction based on prior probabilities of the classes. Using their distribution on the whole data set (87.2% of samples come from white water) would be an unfair baseline for the classifiers since they are trained only on a fraction of it, and the proportions of black and white water samples change significantly with the train set.

Thus, for every train-test split we calculate the prior probability $P(C = 1)$ of a sample being white, using the train set, by dividing the number of white samples by the total number of samples. Then, for each observation in the test set $y_i \in \{1 for White, 0 for Black\}$ we calculate the expected times the 'coin' would identify it correctly and falsely

$$E(\text{Correct Prediction}|y_i) = y_i * P(C = 1) + (1 - y_i) * (1 - P(C = 1)) \tag{2.6}$$

$$E(\text{False Prediction}|y_i) = (1 - y_i) * P(C = 1) + y_i * (1 - P(C = 1)). \tag{2.7}$$

A confusion matrix is constructed (see table 2.2) using the total number of white N_{white} and black N_{black} water samples in the test set.

Maximising Similarity

Aim: Evaluate how well the classifiers perform when they are tested on a set which is similar geographically to the train set.

Method: Testing and validation sets are made up of samples coming from every part of the

	Predicted Black	Predicted White
Actual Black	$N_{black} * E(\text{Correct}\|\text{Black})$	$N_{black} * E(\text{False}\|\text{Black})$
Actual White	$N_{white} * E(\text{False}\|\text{White})$	$N_{white} * E(\text{Correct}\|\text{White})$

Table 2.2 Confusion matrix of baseline benchmark

river. Care is taken to ensure that no geographical area is over represented. This is done using Stratified sampling (using the method StratifieKfold) where the strata (or groups) are the areas of the rivers the samples belong to. Stratified sampling ensures that the distribution of test or validation samples in each area is approximately the same as for train samples.

Maximising Dissimilarity

Aim: Evaluate how well the classifiers perform when they are tested on a set which is dissimilar (or far away) geographically to the train set.

Method: Testing and Validation sets are made up of all the samples which belong to a particular area of the river. For example, the test set might be constituted by all the samples in the Upper Maranon area, and the validation sets by all the remaining areas (Huallaga, Min Maranon, Lower Maranon, Ucayali, Tapiche, and Napo). The method we use to produce these splits is GroupKfold.

Random Splits

Aim: Evaluate the performance of the classifiers on train and test sets obtained by random splitting

Method: Test and Validation sets are obtained by randomly splitting the data. Care is taken to ensure that the balance of white and black water samples is the same across splits. This is done using Stratified sampling with the colour of the river as the strata.

Ideal Splits An ideal sampling method would take into account the spatial correlation structures between the samples, besides just their location. Since the river flows eastwards from Maranon Upper down to all other streams, samples collected upstream would inevitably affect those from downstream, but the opposite might not be true. Thus, euclidean proximity of the river samples is not always a good enough indicator for detecting similarity between samples. For example, sample A collected near the opening of the Tapiche stream and sample B collected a bit further South of the opening, in the Ucayali stream, will have a smaller distance between them than with other samples collected further south in the Tapiche and Ucayaly streams. However, since the streams diverge, A and B might be more similar to samples found downstream their part of the river (Tapiche and Ucayaly respectively) than between them.

To achieve this, a directed graph of the river can be constructed, with vertices representing the samples, and edges the river path between them. Then a sampling scheme can take into

account the stream direction and split the dataset in more sophisticated ways. One such way can test a classifier's ability to predict river colour if the test set is downstream from the train set, and the opposite, if the test set is upstream from the train.

Chapter 3

Methods

In this chapter we introduce all the methods we used in this project. The explanations are not exhaustive but act as a good primer if more details are of interested. In the first section we present ordination methods, which are very frequently used as part of exploratory data analysis in ecological studies. We use them to sketch plots and search for patterns in our data and also as dimensionality reduction techniques.

The classical hypothesis test permutational multivariate analysis of variance is introduced in the following section. It is used to check if the groupings of samples along the levels of a categorical variable significantly different from each other. A distance metric is used to quantify the dissimilarity between samples, and a permutation scheme to avoid assuming the F-statistic's distribution.

Classification models are introduced in the next section together with a brief outline of their working. In particular, the models presented are Logistic Regression and Random Forest. A Bayesian interpretation of the former is also given which can be implemented with Markov chain Monte Carlo methods.

An outline of the Hamiltonian Monte Carlo method for sampling from normalisable distributions is given in the final section of this Chapter. We motivate its use by transforming the problem of sampling to physics terms. Then, we introduce Hamiltonian dynamics and their properties which allow for an efficient exploration of the state space of the target distribution.

3.1 Ordination

Ordination can be thought of as series of operations/transformations performed on a data matrix (samples and species table) with the purpose of representing the relationships between the species and samples as faithfully as possible. These methods are made up of two stages;

the data are converted into a dissimilarity/similarity matrix of the samples or species and are subsequently transformed into a lower dimensional space where the interpoint distances are related to the distance matrix through the scaling method used.

Metric scaling methods try to maximise the linear correlation between the distances in the dissimilarity matrix and in the lower dimensional space. Non-metric methods on the other hand aim at maximising rank-order correlation between those distances instead.

Some of the reasons for carrying out ordination are :

- To make the dissimilarities between the sites or species more evident in a visual manner.
- To reduce the noise from the data, since the projection of the ordination method used will be of a lower dimensional space.
- To interpret environmental gradients along the dimensions.

Ordination methods can be categorised based on how they use external environmental variables; when the method only considers the sites by species table then it is classified as indirect gradient analysis (or unconstrained ordination), when it takes in to account environmental data from the start then it is classified as direct gradient analysis (or constraint ordination).

Indirect gradient analysis finds the most important gradients in the sites by species data (or more descriptive dimensions), in the absence of environmental data. After the procedure is complete, environmental gradients can be fitted on the new basis and explore how the variables change with sites or species. Direct gradient analysis on the other hand explores how much environmental variables are related to the variation of species composition across sites, by taking into consideration only the variation that can be explained by the environmental variables. It can be thought of as a regression technique testing the null hypothesis that the species composition is not related to environmental gradients.

3.1.1 Principal Components Analysis

Principal Component analysis (PCA) is a clustering technique that can be used as an ordination method as well. It works by finding the direction that explains the most variance in the data and setting it as the first axis. Then it finds the direction with the second highest explanation of variance that is orthogonal to the first one, and sets it as the second axis, and the direction that is uncorrelated with both the first and the second axis and explains the most variance as the third, and so on. For ordination, since all ecological community data are measured in the same units, the data do not need to be standardised to unit variance, and thus

the method involves finding the eigenvectors of the covariance matrix of the data (and not the correlation matrix).

Let the row vector x_i with p columns represent the ith sample of our data, with each element denoting how many individuals of that particular column (species/OTU) are encountered in the sample. We can place the vectors into an $n \times p$ matrix X, where n is the number of samples collected. For ecological abundance data we only have to centre the data by finding the mean abundance of each species and subtracting it from matrix X:

$$u_j = \frac{1}{n}\sum_{i=1}^{n} X_{ij} \tag{3.1}$$

$$B = X - hu^T, \tag{3.2}$$

where u_j is the $p \times 1$ vector of mean values, h is a $n \times 1$ vector of ones and B is the centred matrix. This expression can be rewritten using the $n \times n$ centering matrix J which is defined as

$$J = I - \frac{1}{n}O, \tag{3.3}$$

where I is the identity matrix and O is $n \times n$ matrix of ones. Multiplying this matrix on the left with X produces the same result as centering the matrix by columns

$$B = JX. \tag{3.4}$$

The $p \times p$ empirical covariance matrix is then computed as

$$C = \frac{B^T B}{n}, \tag{3.5}$$

which is diagonalised to find its eigenvectors and eigenvalues [20]

$$C = VDV^{-1}, \tag{3.6}$$

where D is the diagonal matrix with the eigenvalues of C in its diagonal, and V is the matrix with the eigenvectors of C as its columns. The projections of the data onto the eigenvectors is given by

$$Y = BV. \tag{3.7}$$

To reduce the number of features, we can select the m eigenvectors of C that correspond to the m largest eigenvalues, sort them in decreasing order (so the first column of V corresponds

to the eigenvector with the largest eigenvalue, the second column to the second largest eigenvalue and so on), and create a new matrix of the sorted eigenvectors

$$W_{ij} = V_{ij} \text{ where i = 1,...,} p \text{ and j = 1,...,} m. \tag{3.8}$$

Then the projection to the reduced eigenvector space is given by

$$Y_r = BW, \tag{3.9}$$

which is a $n \times m$ matrix. When the number of features p is greater than the number of observations n, it is computationally more efficient to diagonalise the Gram matrix G to find the eigenvectors of C

$$G = \frac{BB^T}{n}. \tag{3.10}$$

The Gram matrix has dimensions $n \times n$ and can be diagonalised to give

$$G = USU^{-1}. \tag{3.11}$$

In this instance, U columns are the eigenvectors of the Gram matrix. The connection between this diagonalisation and of the Covariance matrix can be seen more clearly through the singular value decomposition of B

$$B = U\Sigma V^T, \tag{3.12}$$

where U and V are $n \times n$ and $p \times p$ matrices respectively with orthogonal unit vectors as columns. The matrix Σ is an $n \times p$ rectangular diagonal matrix of positive numbers, the singular values of B. Using this representation of B, we can rewrite the covariance and Gram matrix as such

$$C = \frac{B^T B}{n} = \frac{(U\Sigma V^T)^T U\Sigma V^T}{n} = \frac{V\Sigma^T U^T U\Sigma V^T}{n} = \frac{V\Sigma^T \Sigma V^T}{n} \tag{3.13}$$

$$G = \frac{BB^T}{n} = \frac{U\Sigma V^T (U\Sigma V^T)^T}{n} = \frac{U\Sigma V^T V\Sigma^T U^T}{n} = \frac{U\Sigma\Sigma^T U^T}{n}. \tag{3.14}$$

Since matrices U and V are orthogonal, their inverse is equal to their transpose (e.g. $U^{-1} = U^T$) [20]. The projection of the centred data onto the eigenvectors of the covariance matrix C can be reformulated using the SVD

$$Y = BV = U\Sigma V^T V = U\Sigma. \tag{3.15}$$

This reformulation tells us that the projections can also be found using the eigenvectors of the Gram matrix (i.e. the left-singular vectors of B). The projection to the reduced m-dimensional eigenvector space can be given in a similar manner by

$$Y_r = U_m \Sigma_m, \tag{3.16}$$

where Σ_m is the matrix of the m largest singular values, and U_m the matrix with their corresponding eigenvectors as columns.

The centred $n \times p$ data matrix B has maximum rank r equal to the minimum of the numbers n and p. This means that there are at most r linearly independent row (or column) vectors, and consequently, at most r singular values in the $n \times p$ matrix Σ. Comparing the two decompositions of the Covariance and Gram matrix, we note that the matrices $\Sigma^T\Sigma$ and $\Sigma\Sigma^T$ have the same number of non-zero elements in their diagonals, even if they are of dimension $p \times p$ and $n \times n$ respectively. Thus, we can conclude that the Covariance and Gram matrices have the same eigenvalues.

An important assumption of PCA is that there are linear relationships between all the features. Therefore, the principal components are constructed such that they decouple the linear correlations between them. This assumption, however, makes it unsuitable for use in Ecological data sets. The species composition varies non linearly across samples, and the relationship between species is also not linear. The presence of a species in a sample is usually much more important that the actual number of read counts of that species. Thus, when the data are projected onto the eigenvectors of the covariance matrix (Figure 3.1), they are distorted into a horseshoe shape (i.e. like an arch) [21]. PCA was performed using the rda function in the Vegan package [37, 36].

3.1.2 Principal Coordinates Analysis

A more general method of ordination, of which PCA is a special case, is called Principal Coordinates Analysis (PCoA), otherwise known as classical multidimensional scaling. The method aims to represent dissimilarities between samples (in the species space) in a lower dimensional space, so that they can be easily interpreted. This is done by creating a distance matrix using whatever metric we wish (with the condition that it returns a scalar given two vectors of arbitrary dimensions) that quantifies the dissimilarities between our samples. The data are then projected to a lower dimensional space by maximising the linear correlation between the distances in the distance matrix and the inter-point distances in the lower dimensional projection. Thus, the method assumes that the distances used are meaningful

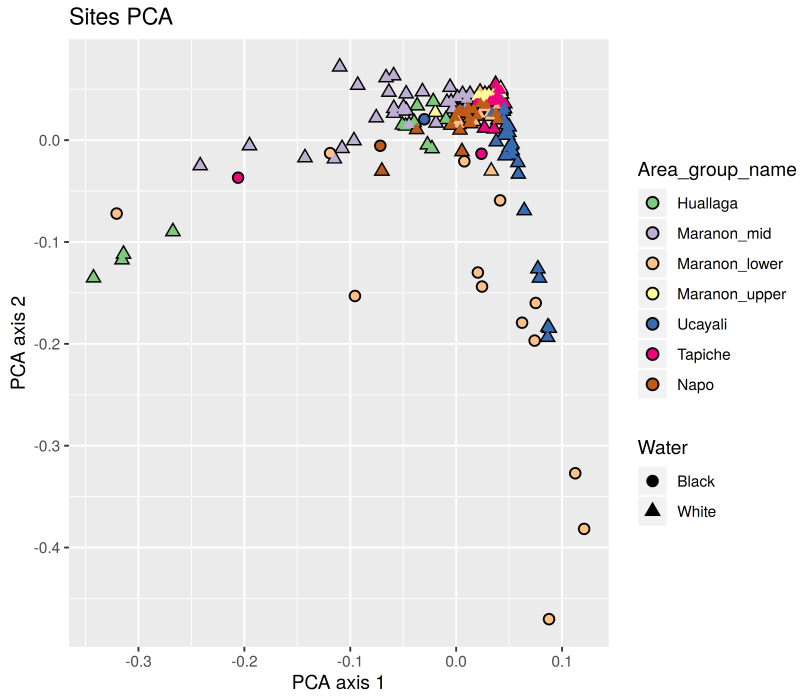


Fig. 3.1 First 2 dimensions of PCA performed on the full OTU table. The 2 axes account for 47.8% of the variance.

and thus try to reserve them. It was first introduced by Torgerson in 1952 as a tool in psychometrics [49].

Lets define by D_{ij} the $n \times n$ distance matrix between the samples (rows) in X (where D_{ij} indicates the distance between sample i with j). It is evident from its construction that it is symmetric, since the distance between sample i and j is the same as the distance between j and i. When using a euclidean metric to quantify the distance the diagonals are zero. The euclidean metric is given by

$$D_{ij} = ||\mathbf{x}_i - \mathbf{x}_j||, \tag{3.17}$$

where $\mathbf{x}_i$ is a row vector of the data matrix X containing the abundance reading for the sample i, and $||\cdot||$ is the L2 norm.

The algorithm first double centres the element squared distance matrix D^2, or in other words subtracts the row and column mean, and multiplying it by $-\frac{1}{2}$

$$K = -\frac{1}{2} J D^2 J. \tag{3.18}$$

Then the matrix K is decomposed into its eigenvectors, which are the columns of matrix E, and eigenvalues, which make up the diagonals of matrix Λ

$$K = E\Lambda E \tag{3.19}$$

The m largest eigenvalues with their corresponding eigenvectors are collected and sorted in a descending manner in matrices E_m and Λ_m (so the largest eigenvalue is $\Lambda_{m,11}$ with corresponding eigenvector $E_{m,\bullet 1}$, the first column of the E_m matrix). The projection of the data onto the reduced m-dimensional space is given by

$$Y_r = E_m \Lambda_m^{(1/2)}, \tag{3.20}$$

where the exponent is applied element-wise on all eigenvalues (the square root of each eigenvalue is used). This representation looks very similar to the one obtained in equation 3.16, where the projection is calculated using the eigenvectors of the Gram matrix. We will show how, when using an Euclidean metric, PCoA is equivalent to PCA.

We can expand the euclidean distance matrix into

$$D_{ij}^2 = ||\mathbf{x}_i - \mathbf{x}_j||^2 = ||\mathbf{x}_i - \bar{\mathbf{x}} + \bar{\mathbf{x}} - \mathbf{x}_j||^2 \tag{3.21}$$

$$= ||\mathbf{x}_i - \bar{\mathbf{x}}||^2 + ||\mathbf{x}_j - \bar{\mathbf{x}}||^2 - 2(\mathbf{x}_i - \bar{\mathbf{x}}) \cdot (\mathbf{x}_j - \bar{\mathbf{x}}), \tag{3.22}$$

where the last term is the dot product between the vectors of the mean-centred samples i and j. The $||\mathbf{x}_i - \bar{\mathbf{x}}||^2$ term is an additive constant over the columns of D_{ij}, and so is the corresponding term with $\mathbf{x}_j$ over the rows of the matrix. To uncover the connection between the last term and the Gram matrix, we have to reformulate the later to an equivalent representation of the former.

The mean centred matrix B can be interpreted as row vectors $\mathbf{x}_i - \bar{\mathbf{x}}$ stacked on top of each other vertically. As before, $\mathbf{x}_i$ denotes the row vector of data matrix X (i.e. the species abundance of sample i) and $\bar{\mathbf{x}}$ denotes the row vector with the mean number of reads of each species across all samples

$$\bar{\mathbf{x}} = [u_1, u_2, ..., u_p], \tag{3.23}$$

where u_j is defined as in equation (3.1). To construct the Gram matrix, we multiply the mean-centred matrix with its transpose

$$\frac{1}{n}\begin{pmatrix} — (x_1-\bar{x}) — \\ — (x_2-\bar{x}) — \\ \vdots \\ — (x_n-\bar{x}) — \end{pmatrix} \cdot \begin{pmatrix} | & | & & | \\ (x_1-\bar{x}) & (x_2-\bar{x}) & \dots & (x_n-\bar{x}) \\ | & | & & | \end{pmatrix} = \frac{1}{n}\begin{pmatrix} (x_1-\bar{x})\cdot(x_1-\bar{x}) & \dots & (x_1-\bar{x})\cdot(x_n-\bar{x}) \\ (x_2-\bar{x})\cdot(x_1-\bar{x}) & \dots & (x_2-\bar{x})\cdot(x_n-\bar{x}) \\ \vdots & \ddots & \vdots \\ (x_1-\bar{x})\cdot(x_n-\bar{x}) & \dots & (x_n-\bar{x})\cdot(x_n-\bar{x}) \end{pmatrix}, \tag{3.24}$$

and get the expression

$$G_{ij} = \frac{1}{n}(x_i-\bar{x})\cdot(x_j-\bar{x}). \tag{3.25}$$

The Gram matrix rows' and columns' means are equal to zero since it is double-centred by construction

$$G = \frac{1}{n}BB^T = \frac{1}{n}JX(JX)^T = \frac{1}{n}JXX^TJ^T = \frac{1}{n}JG_{uc}J, \tag{3.26}$$

where J is the centring matrix (which is symmetric), and G_{uc} the uncentered Gram matrix. Therefore, when we double centre the Distance matrix, we end up with the Gram matrix scaled by a constant number

$$JD_{ij}J = J\left(||x_i-\bar{x}||^2 + ||x_j-\bar{x}||^2 - 2nG_{ij}\right)J \tag{3.27}$$

$$\propto -2G_{ij} \tag{3.28}$$

$$K \propto G, \tag{3.29}$$

Where K is the matrix we decompose into its eigenvalues and eigenvectors in the PCoA method (equation (3.18)). The two terms preceding the Gram matrix go to zero when double-centred since they are constants over rows and columns. As mentioned earlier, the Gram matrix itself stays the same when double centred since its rows' and columns' mean is zero.

Therefore, diagonalising the K matrix when using a euclidean distance metric (PCoA method) is equivalent to diagonalising the Gram matrix of the mean-centred data. The projections produced by the two methods are the same since the eigenvector (E for PCoA and U for PCA) and eigenvalue ($\Lambda^{(1/2)}$ for PCoA and Σ for PCA) matrices are the same.

A Euclidean distance metric, however, is not very useful when it comes to ecological abundance data. This is because it suffers from the same drawbacks that PCA does (see earlier discussion in section 3.1.1). The framework of PCoA was developed so as to enable the use of other measures which are more suitable to ecological data; as mentioned earlier, the number of read counts for each species might suffer from biases and thus not much

emphasis should be place on their value. Such a measure is the Bray-Curtis dissimilarity statistic, which quantifies how dissimilar two samples are by taking the absolute value of their difference. The measure B_{ij} is defined as

$$B_{ij} = \frac{\sum_{k=1}^{p} |X_{ik} - X_{jk}|}{\sum_{k=1}^{p} X_{ik} + X_{jk}}, \tag{3.30}$$

where X_{ik} is the data matrix which denotes the number of reads of speciment k in sample i. The statistic ranges from 0, where the samples have the same species composition, to 1 where the samples do not have any species in common, and is semimetric[1]. If the data matrix is instead made up of presence and absence values (1 if an OTU was found in a sample, 0 otherwise), the Bray-Curtis index quantifies the fraction of species not in common between the samples.

If we use this statistic to calculate the distance matrix D_{ij} and then carry out PCoA, the arch effect is absent from the 2 dimensional ordination plot and can better separate the different river samples. As it can be seen in Figure 3.2, the method can separate well samples from the upper Maranon part of the river (yellow points in the upper right corner of the plot). Samples from other parts are grouped together on the opposite side from the upper Maranon points. Black and white river samples are not very well separated, and occupy the same space (except on the upper right corner where no black water samples are found). Even when the ordination method produces a better spread of results than when using PCA, the variance explained by the first 2 axes is only 19.8%.

To illustrate the effect of measures on the PCoA algorithm, we produced 2 dimensional ordination plots for the Euclidean and the Jaccard metrics. The Euclidean metric has the form given in equation (3.17) and the ordination plot (see Figure 3.3a) has the same form as the one obtained using PCA. The only difference is the scale (the covariance matrix is divided by the number of samples) and the reflection of the points on the x-axis. The Jaccard metric is given by

$$Jac_{ij} = \frac{2C_{ij}}{1 - C_{ij}}, \tag{3.31}$$

where C_{ij} is the Bray-Curtis statistic between sample i and j. The ordination plot produced using this index is shown in Figure 3.3b. PCoA was performed using the pcoa function in the ape package in R [38].

[1] Semimetric measures do not satisfy the triangle inequality.

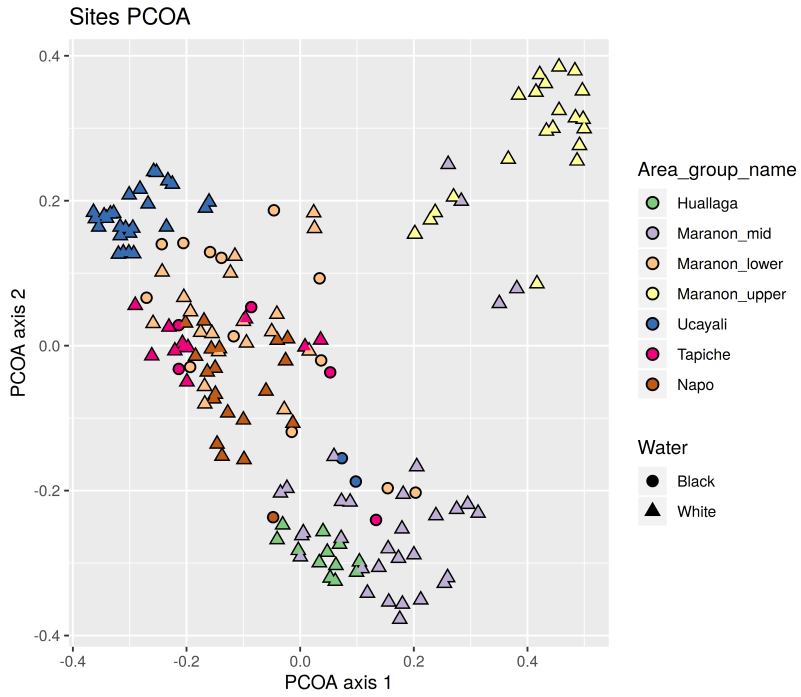
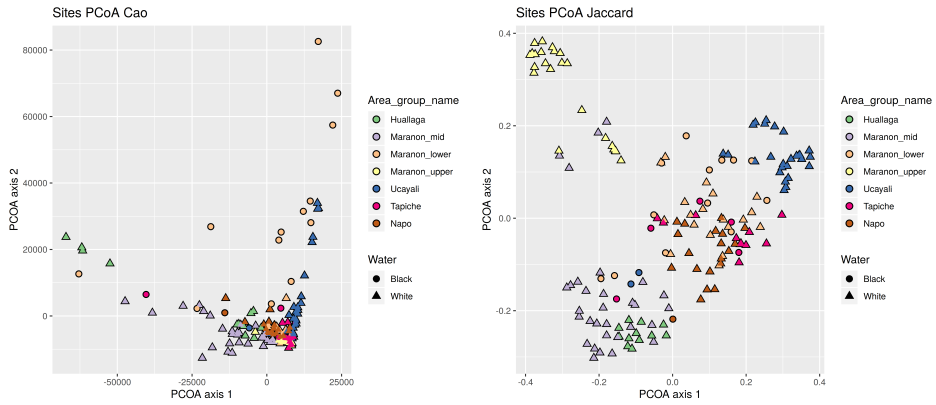


Fig. 3.2 First 2 dimensions of PCoA performed on the full OTU table using the Bray Curtis statistic as the distance metric. The 2 axes account for 19.8% of the variance.

3.1.3 Non-metric multidimensional scaling

Non-metric multidimensional scaling (NMDS) is distinct from classical multidimensional scaling methods, like PCA and PCoA, in that it does not assume there is a linear relationship between the actual ecological distance of two samples and their dissimilarity (e.g. using Bray-Curtis). In contrast it assumes only that there is a rank order relationship between the two (i.e. the most different samples should also have the largest dissimilarity). Thus, it derives a configuration of points representing the samples, in which the inter-point distance of the pairs are in rank order with the dissimilarities of the samples. Furthermore, it is a random iterative approach and the number of dimensions of the configuration of points must be decided from beforehand. Like PCoA, NMDS was first introduced in the psychometrics field for use in psychology [26].

The goal of the algorithm is to find n points in an m dimensional spaces whose inter-point distances are somewhat related to the experimental dissimilarities of our n objects/samples. In this instance, dissimilarity is given by the distance matrix between samples Δ_{ij} which is calculated using a dissimilarity measure. In NMDS ordination we are not interested in the numerical distance between our objects but rather their ordinal relationship (the rank of dissimilarities between samples is of importance); that's why the word dissimilarity is used.



(a) (b)

Fig. 3.3 PCoA using the Euclidean 3.3a and Jaccard 3.3b metric. The result for the Euclidean metric is the same as with PCA; only the scales have different values and the points are flipped over the X-axis. The Jaccard metric produces a similar result with Bray-Curtis since they are rank-order similar.

In contrast, PCoA or classical MDS assume a linear relationship between the calculated dissimilarity measure and the ecological distances of the samples. What we label as distance D_{ij} in NMDS is between the points in the m dimensional projection of our object. A measure of relatedness between distance and dissimilarity is given by a goodness-of-fit statistic which is an integral part of the procedure; in classical MDS however no such statistic is used.

This statistic is called Stress and it measures how the configuration of points in the m dimensional space match the data, through a monotonic relationship between them. A solution to the iterative algorithm, the best-fitting configuration of points, is found when the stress is minimised. The best solution is achieved when there is a perfect monotone relationship between the dissimilarities and distances.

The dissimilarities we are dealing with are symmetric; $\Delta_{ij} = \Delta_{ji}$. Therefore, only the upper triangular matrix is needed without the diagonal (because samples are perfectly similar between themselves). We want to achieve a configuration of points in an m dimensional space such that the rank ordering of dissimilarities

$$\Delta_{i_1 j_1} \leq \Delta_{i_2 j_2} \leq \ldots \leq \Delta_{i_M j_M}, \tag{3.32}$$

where $M = \frac{n(n-1)}{2}$ is the number of interactions between the samples, is matched by the rank ordering of distances D_{ij}.

The objects are represented by points/vectors in an m-dimensional space $\mathbf{x}_i \in \mathbb{R}^m$ or by the matrix X_{ij}, and their distance by any metric we want. In this case we choose the

Euclidean:

$$D_{ij} = \sqrt{\sum_{k=1}^{i} \left(X_{ik} - X_{jk}\right)^2}. \tag{3.33}$$

An important element of the procedure are the numbers $\hat{D}_{ij}$, which are used as proxies for the dissimilarities when it comes to calculating the stress. The reason we do not use the numerical values of the dissimilarities is because we are not interested in them, only in their rank order relationship. Thus, the numbers $\hat{D}_{ij}$ are constructed as such to be monotonically related to Δ_{ij}; when arranged in the order

$$\hat{D}_{i_1 j_1}, \hat{D}_{i_2 j_2}, ..., \hat{D}_{i_M j_M}, \tag{3.34}$$

each number is greater than or equal to the one before it. These numbers are chosen so as to be as 'nearly equal' to D_{ij} as possible, while satisfying the monotonicity of Δ_{ij} (3.32).

Now we can introduce the stress formula for a configuration of points X_{ij}. This is given by

$$S(X_{ij}) = \text{stress of the fixed configuration } \mathbf{x}_1, ..., x_n \tag{3.35}$$

$$= \min_{\text{numbers } \hat{D}_{ij} \text{ satisfying (3.32)}} \sqrt{\frac{\sum_{i<j} \left(D_{ij}, -\hat{D}_{ij}\right)^2}{\sum_{i<j} D_{ij}^2}}, \tag{3.36}$$

where the sum is taken over all ij pairs such that $i < j$ is satisfied (sum over all upper triangular matrix elements), and the minimum is over the numbers $\hat{D}_{ij}$. This minimisation can also be seen as monotonic regression, if the dissimilarities Δ_{ij} are plotted against the distances D_{ij} and the proxies are the points that make up the free-form line. The regression step is finding new $\hat{D}_{ij}$ numbers when the distances change.

The algorithm then aims at finding the configuration of points that minimises (3.36). The minimisation over the configuration is done by a 'method of gradients' (similar to gradient descent) outlined in [27]. That over the $\hat{D}_{ij}$ is done using a method of blocks outlined in [33, 27].

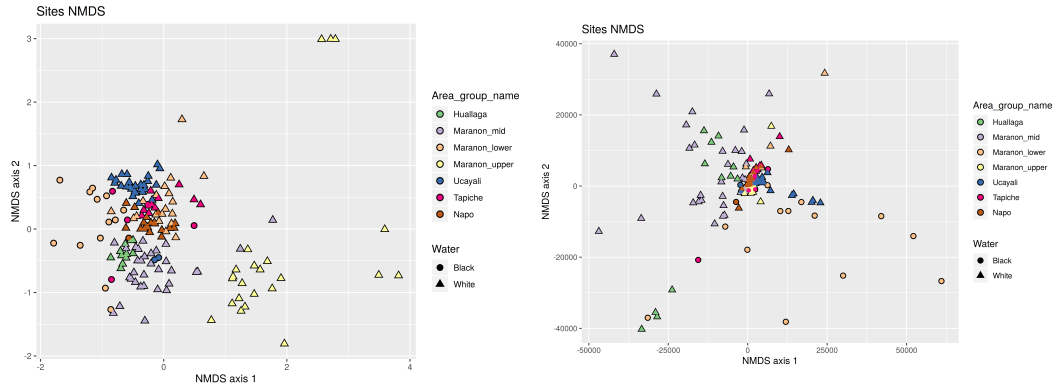
The algorithm first starts by generating a random configuration of n non-identical points that span an m-dimensional space. This can also be done by running another ordination method and using it's configuration of points limited to the m first dimensions. This set is then normalised; the centre of gravity is set to the origin, and the points are uniformly stretched or shrank such that their root-mean-square distance from the origin is equal to one. This transformation does not affect the algorithm since the euclidean metric is invariant to

translations and rotations, and the denominator of stress also makes it invariant to a uniform stretch of the points.

The interpoint distance D_{ij} is then calculated and the stress for the configuration is found by minimising the dissimilarity proxies $\hat{D}_{ij}$ in the way described earlier. Then, the gradient of the configuration is calculated and, if some stopping conditions are not met that signify the algorithm reached an appropriate minimum, the points are moved towards the direction of decreasing stress. A new configuration is thus obtained and the procedure is repeated until some minimum conditions are met; either the stress is low enough that finding another minimum would not improve the solution significantly, or several attempts from different starting configurations have been attempted and the lowest stress solution is chosen.

NMDS ordination was performed using the metaMDS function in vegan [37] package for R. As mentioned previously, the number of dimensions have to be predetermined before running the algorithm. There is also a maximum number of iteration to try if a non-convergent solution is found. Ordination plots for Bray-Curtis and Euclidean dissimilarity measures are shown in Figures 3.4a and 3.4b. The same legends and colouring schemes have been used as in the PCoA case. The Bray-Curtis measure can separate river colour much better than when using the Euclidean measure.

The stress of the 2 dimensional NMDS configuration was 15.8% and 12.8% for Bray-Curtis and Euclidean respectively. This is considered 'fair' in the literature, and is good enough for discerning patterns in a plot. With increasing dimensions the stress falls, however, the number of iterations needed to get a convergent solution also increases considerably. After 100000 iterations a convergent solution for 20 dimensions was not found. Thus, its inclusion as a feature selection method might not be warranted.



(a) (b)

Fig. 3.4 NMDS ordination plots using Bray-Curtis 3.4a and Euclidean 3.4b measure for dissimilarities. Bray-Curtis separates the rivers and colour much better than the Euclidean measure.

3.2 Permutational multivariate analysis of variance

Permutational multivariate analysis of variance is a hypothesis testing technique that involves permuting the data set such that normality assumptions are avoided. Its use in ecology is to test if the OTU (read counts) distribution of two (or more) groups of samples is significantly different. Distance measures are used to define the relationships between the samples.

Particularly, the null hypothesis is: The centroids and dispersion of the groups, defined by a distance measure, are equivalent for all groups. A rejection of the null means that the centroid and/or the dispersion is different between groups.

In the case of a one-way test, testing for the effect of a single grouping variable (our case), with a groups and n samples, we construct the distance matrix D_{ij} using any measure relevant to our data. The matrix is constructed in the same way as in section 3.1 and it quantifies the dissimilarity between observations i and j. The index of samples in each group k is given by I_k. We are interested in testing if the grouping of samples by water colour produces significantly different distributions is OTU abundance.

Some statistics first need to be calculated, so that the test-statistic can be constructed. The total sum of squares is

$$SS_T = \frac{1}{n}\sum_{i<j} D_{ij}^2, \tag{3.37}$$

where the sum is taken over the lower triangular distance matrix, excluding the diagonal. The group k sums of squares is given by

$$SS_{w,k} = \frac{1}{n_k}\sum_{i<j} \varepsilon_{ij} D_{ij}^2 \tag{3.38}$$

$$\varepsilon_{ij} = \begin{cases} 1, & \text{if } i,j \in I_k \\ 0, & \text{otherwise} \end{cases}, \tag{3.39}$$

where the sum is taken over the samples belonging to the particular group. This amounts to summing up all the entries in the squared distance matrix that belong to group k. Summing up the group sums gives us the within groups sum-of-squares

$$SS_w = SS_{w,white} + SS_{w,black}. \tag{3.40}$$

With this we can define the among-group sum-of-squares $SS_A = SS_T - SS_w$ which can be used to construct a pseudo F-ratio

$$F = \frac{SS_A/(a-1)}{SS_w(n-a)}, \tag{3.41}$$

where $(a-1)$ are the degrees of freedom associated with the grouping factor and $(N-a)$ are the residual degrees of freedom [2].

The distribution of this F-ratio under the null hypothesis is unknown, and we do not want to assume any. That is why permutations of the samples are used to generate a distribution of F. If the groupings have no discernible effect on the samples, then it is equally likely that the group labels were associated with any of the samples. So, under a true null hypothesis, we can shuffle the labels onto the samples, calculate F-ratios (denoted by F_p) and construct an empirical (discrete) distribution $\{F_p\}_p$.

If the null hypothesis is true, then the F-ratio calculated using the original ordering of labels relative to the samples will be similar to the values obtained under permutation. If the grouping however, produces a significant effect, then the original F value will be larger than those obtained from the permutation $\{F_p\}_p$. To test for significance we calculate the probability associated with F under the true null by counting the number of permuted ratios F_p that have larger value than F and dividing by the total number of permutations

$$P = \frac{\#\{F_q \in \{F_p\}_p | F_q \geq F\} + 1}{\#\{F_p\}_p + 1}. \tag{3.42}$$

We include the original F value in the distribution as well, that is why $+1$ is included in both denominator and numerator.

Not all possible permutations of the labels are necessary, a subset of them is enough to calculate a valid P-value. If it is calculated to be less than 5% then we can reject the null. Of course the permutation affects greatly the calculated value so it is important that an appropriate scheme is used. The one we choose involves permuting the labels within sites. As explained earlier, 4-6 samples were collected for each site along the rivers. This means that even if samples collected close together came from different water colour, their dissimilarity would be relatively low when compared to far away sites. Thus, this shuffling will make it less probable for water colour to have a significant effect.

The P-value calculated when using the OTU table and the permutation scheme outlined previously, with 2000 permutations, was 2.6%. When no restriction is put on the shuffling, the P-value is 0.05%, significantly smaller. This means the null hypothesis is rejected, and the two distributions have different centroids and/or dispersion, defined by the distance matrix. This is of course evident when one looks at the ordination plots in section 3.1, which show a different spread of black and white water samples. This is especially true for NMDS.

The test was performed using the function adonis from the Vegan package in R.

3.3 Classification Models

3.3.1 Logistic Regression

Bayesian approach

Binary classification involves deciding in which of two groups (classes) an object (point) should be categorised. A point is constituted of a response variable and features. The purpose of a classifier is then to correctly identify the response variable given it's features and any other relevant information.

Our data set's response variable is the water colour of the sample, defined here as $y_i \in \{1(White), 0(Black)\}$ for the ith sample, where $i = 1, ..., N$. The notation $\mathbf{x}_i = [x_{i1}, x_{i2}, ..., x_{im}]$ is used to represent the m dimensional row vector of features for the ith sample. Stacking up the feature vectors we construct the matrix X_{ij}, with rows denoting the samples and columns the features.

Assuming we do not know the response variable of a point we would like to calculate its probability of belonging to one of the classes given its features

$$P(y_i = 1(\text{White})|\mathbf{x}_i). \tag{3.43}$$

We can use Bayes' theorem to expand this probability into

$$P(y_i = 1|\mathbf{x}_i) = \frac{P(\mathbf{x}_i|y_i)P(y_i)}{P(\mathbf{x}_i)}, \tag{3.44}$$

where the likelihood is given by $P(x_i|y_i)$, the prior by $P(y_i)$, and the marginal likelihood by $P(y_i)$. The prior indicates the point's probability of being in a class without taking into account its features. All the information about the point's features is encoded in the likelihood, where the probability of obtaining them is conditioned on the class. Finally, the marginal is the probability of obtaining the data, whatever the class of the point. It can thus be obtained by integrating out the class variable

$$P(\mathbf{x}_i) = \sum_{y_i=0}^{1} P(\mathbf{x}_i|y_i)P(y_i). \tag{3.45}$$

To decide in which class a point belongs to, we construct a decision boundary using the discriminant function

$$f(\mathbf{x}_i) = \log \frac{P(y_i = 1|\mathbf{x}_i)}{P(y_i = 0|\mathbf{x}_i)}. \tag{3.46}$$

If the function is greater than 0 that means that the probability of the point coming from White water is greater than from Black water.

We can adopt either a generative or a discriminative approach. The generative approach requires modelling the class conditional distributions $P(\mathbf{x}_i|y_i = 1)$ and $P(\mathbf{x}_i|y_i = 0)$, and using Bayes' theorem to obtain the discriminant function directly. Typical samples could also be generated by drawing data from $P(\mathbf{x}_i|y_i)$. The discriminant approach involves modelling the discriminant function (3.46) directly, using a linear model for example. We will be using the discriminative approach so as to avoid modelling the complex interaction between water colour and OTU-derived features.

A linear function (in parameters) can be used to model (3.46). The features can be transformed by a non-linear function $\phi(\mathbf{x}_i) = [\phi_1(\mathbf{x}_i), \phi_2(\mathbf{x}_i), ..., \phi_D(\mathbf{x}_i)]$ which can span a higher or lower dimensional space. The model is given by

$$a = \log \frac{P(y_i = 1|\mathbf{x}_i)}{P(y_i = 0|\mathbf{x}_i)} = \phi(\mathbf{x})\boldsymbol{\theta}^T \tag{3.47}$$

where $\boldsymbol{\theta} = [\theta_1, \theta_2, ..., \theta_D]$ are the model parameters.

We can simplify this expression by making the observation that the conditional class probabilities sum up to one

$$\frac{P(y_i = 1|\mathbf{x}_i)}{P(y_i = 0|\mathbf{x}_i)} = \exp(a) \tag{3.48}$$

$$\frac{P(y_i = 1|\mathbf{x}_i)}{1 - P(y_i = 1|\mathbf{x}_i)} = \exp(a) \tag{3.49}$$

$$P(y_i = 1|\mathbf{x}_i) = \frac{\exp(a)}{1 + \exp(a)}, \tag{3.50}$$

and making use of the sigmoid function. The probability of point i belonging to class $y_i \in \{1, 0\}$ is thus

$$\begin{aligned} P(C = y_i|\mathbf{x}_i, \boldsymbol{\theta}) &= P(C = 1|\mathbf{x}_i)^{y_i}(1 - P(C = 1|\mathbf{x}_i))^{1-y_i} \\ &= \left(\frac{\exp(\phi(\mathbf{x})\boldsymbol{\theta}^T)}{1 + \exp(\phi(\mathbf{x})\boldsymbol{\theta}^T)}\right)^{y_i} \left(\frac{1}{1 + \exp(\phi(\mathbf{x})\boldsymbol{\theta}^T)}\right)^{1-y_i} \\ &= \frac{\exp(\phi(\mathbf{x})\boldsymbol{\theta}^T)^{y_i}}{1 + \exp(\phi(\mathbf{x})\boldsymbol{\theta}^T)}. \end{aligned} \tag{3.51}$$

The aim now is to figure out which distribution of parameters $\boldsymbol{\theta}^T$ model a decision function best so that it classifies the most points correctly. To do that, we agglomerate all

of our samples into two sets; train and test. The train set will be used to 'learn' the best parameters and the test set to evaluate it's performance on unseen data. Thus the response variables $\mathbf{y}$ are grouped into $\mathbf{y}_{train}$ and $\mathbf{y}_{test}$, and the features X into X_{train} and X_{test}. The index set of samples belonging to train and test is given by I_{train} and I_{test} respectively.

Because of the non-trivial interactions between our samples, we will be assuming that they are independent and identically distributed

$$P(\mathbf{y}|X,\boldsymbol{\theta}) = \prod_{n=1}^{N} P(C = y_i|\mathbf{x}_n,\boldsymbol{\theta}) \tag{3.52}$$

$$= \prod_{n=1}^{N} \frac{\exp\left(\phi(\mathbf{x}_i)\,\boldsymbol{\theta}^T\right)^{y_n}}{1+\exp\left(\phi(\mathbf{x}_i)\,\boldsymbol{\theta}^T\right)}. \tag{3.53}$$

We can thus construct the posterior distribution of the parameters $\boldsymbol{\theta}$ by utilising Bayes' theorem and conditioning on the train sets

$$P(\boldsymbol{\theta}|X_{train},\mathbf{y}_{train},s) = \frac{P(\mathbf{y}_{train}|\boldsymbol{\theta},X_{train})\,P(\boldsymbol{\theta}|s)}{P(\mathbf{y}_{train}|X_{train},s)}. \tag{3.54}$$

The s denotes the (hyper)parameters of the prior that we have to assign from before hand. The marginal likelihood is obtained by integrating out the parameters from the likelihood and the prior using the law of total probability

$$P(\mathbf{y}_{train}|X_{train},s) = \int P(\mathbf{y}_{train}|\boldsymbol{\theta},X_{train})\,P(\boldsymbol{\theta}|s)\,d\boldsymbol{\theta}. \tag{3.55}$$

The integral is usually intractable analytically because the parameters span a high-dimensional space.

Once we have the posterior distribution we can use it on a new data set (test set), make predictions, and evaluate the model's performance using the predictive posterior distribution. This is done using the law of total probability

$$P(C=1|X_{train},\mathbf{y}_{train},\mathbf{x}_{i,test},s) = \int P(C=1|\mathbf{x}_{i,test},\boldsymbol{\theta})P(\boldsymbol{\theta}|X_{train},\mathbf{y}_{train},s)d\boldsymbol{\theta} \tag{3.56}$$

$$= E(P(C=1|\mathbf{x}_{i,test},\boldsymbol{\theta})|X_{train},\mathbf{y}_{train},s), \tag{3.57}$$

where $\mathbf{x}_{i,test} = \mathbf{x}_{i\in I_{test}}$ is the features vector of any sample in the test set. This is the probability of observation i being in group 1 (White water) given the train data sets and the parameter of the prior. The probability $P(C=1|\mathbf{x}_{i,test},\boldsymbol{\theta})$ is the likelihood we encountered earlier and has

the analytic form (3.51). Note that the distribution over the Class is not dependent on the parameters $\boldsymbol{\theta}$ since we integrated them out.

A Bayesian approach to train and predict with such a model is to use Markov chain Monte Carlo (MCMC) methods to sample from the posterior and then approximate the posterior predictive probability (3.56) using a Monte Carlo estimator. The M samples obtained are denoted with $z^{(j)}$ and have the same dimension as the parameters $\boldsymbol{\theta}$. The estimate,

$$\int P(C=1|\mathbf{x}_{i,test},\boldsymbol{\theta})P(\boldsymbol{\theta}|X_{train},\mathbf{y}_{train},s)d\boldsymbol{\theta} = \frac{1}{M}\sum_{j=1}^{M} P(C=1|\mathbf{x}_{i,test},z^{(j)}) \tag{3.58}$$

$$= \frac{1}{M}\sum_{j=1}^{M} \frac{\exp(\phi(\mathbf{x}_{i,test})\cdot(z^{(j)})^T)}{1+\exp(\phi(\mathbf{x}_{i,test})\cdot z^{(j)T})}, \tag{3.59}$$

can be evaluated with the generated samples for each point in the test set.

The idea of MCMC is to generate samples $z^{(i)}$ while exploring the state space of interest (the space on which the posterior is defined, Θ) using a markov chain mechanism. The chain is constructed such that it spends more time in more important regions. In particular, the aim is for the samples generated to mimic samples drawn from the target distribution ($P(\boldsymbol{\theta}|X_{train},\mathbf{y}_{train},s)$). The reason MCMC methods are of interest in our case is that they allow drawing samples from the target distribution if it can be evaluated up to a normalising constant (but does not have to be normalised, so we avoid calculating the marginal likelihood).

The M resulting samples can be used to approximate the target density

$$P_M(\boldsymbol{\theta}|X_{train},\mathbf{y}_{train},s) = \frac{1}{M}\sum_{i=1}^{M}\delta(\boldsymbol{\theta}=z^{(i)}), \tag{3.60}$$

where $\delta(\boldsymbol{\theta}=z^{(i)})$ denotes the Dirac delta function centred at the vector $z^{(i)}$. Using this point-mass function, integrals $I(f)$ can be approximated with sums $I_M(f)$ that converge in the following manner

$$I_M(f) = \frac{1}{M}\sum_{i=1}^{M} f(z^{(i)}) \underset{M\to\infty}{\to} \int_\Theta f(\theta)P_M(\boldsymbol{\theta}|X_{train},\mathbf{y}_{train},s)d\theta = I(f). \tag{3.61}$$

If the $z_{(i)}$ samples were i.i.d, the estimator $I_M(f)$ is unbiased and by the strong law of large numbers it almost surely converges to $I(f)$. MCMC samples need to satisfy some conditions for them to be used in monte carlo estimates.

The ergodic theorem generalises the law of large numbers. Suppose that $\{z^{(i)}\}_{i\in\mathrm{N}}$ is an ergodic Markov Chain with stationary distribution our target distribution and function $f:\mathrm{R}^D\to\mathrm{R}$ defined on the state space of our sample. Then (3.61) holds.

This motivates the use of Metropolis-Hastings, an MCMC sampling method. A step of the algorithm involves drawing a sample z^* from a proposal distribution (that aims at exploring the state space) $q(z^*|z^{(i-1)})$ given the current value of the chain $z^{(i-1)}$. Then the chain moves to the candidate value z^* with an acceptance probability

$$min\left(1, \frac{p(z^*)q(z^*|z^{(i-1)})}{p(z^{(i-1)})q(z^{(i-1)}|z^*)}\right), \tag{3.62}$$

where $p(z)$ is the unnormalised target density (or the target density up to a constant, the marginal likelihood, independent of z).

If the candidate z^* is not accepted then a new one is drawn from the proposal conditioned on the previously accepted sample. The process goes on up until we have generated a sufficient number of samples. More detailed explanation of the algorithm and the theory behind Markov Chain can be found in [28].

Instead of using Metropolis-Hastings (MH), Hamiltonian Monte Carlo (HMC) was used. When compared to a MH using a Gaussian random-walk as proposal distribution, it reduces the correlation between successive sampled states, and thus it needs fewer samples to approximate integrals. An outline of the method is given in section 3.4.

Maximum Likelihood Estimation

The Maximum Likelihood approach to fitting a logistic regression model to the data involves, as the name suggests, maximising the likelihood of response variables given the features and the parameters , with respect to the parameters (3.53).

Instead of looking at the likelihood we can look at its log transform, which has the same order relations as the likelihood

$$P(a_1|b) > P(a_2|b) \Leftrightarrow \log P(a_1|b) > \log P(a_2|b). \tag{3.63}$$

We can define the loss function by taking the negative logarithm of the likelihood, which gives what is called the cross-entropy loss function

$$L(\boldsymbol{\theta}) = -\left(\sum_{n=1}^{N} y_n \phi\left(\mathbf{x}_i\right) \cdot \boldsymbol{\theta}^T - \log\left(1 + \exp\left(\phi\left(\mathbf{x}_i\right)\boldsymbol{\theta}^T\right)\right)\right). \tag{3.64}$$

Thus, instead of maximising the likelihood, we can minimise the loss, which allows for more computational accuracy. The parameters $\boldsymbol{\theta}^*$ that minimise this loss are then used for the

prediction of new features. If the discriminant function,

$$\log \frac{P(C=1|\mathbf{x}_i)}{P(C=0|\mathbf{x}_i)} = \phi(\mathbf{x})(\boldsymbol{\theta}^*)^T, \tag{3.65}$$

is bigger than 0 then the point is classified as coming from white water, and if it is smaller then from black.

Regularisation can also be introduced that would reduce over fitting on the training data. If not employed, the model might become too complicated, modelling the noise in our data, and thus not generalisable. Using regularisation in machine learning amounts to penalising model complexity by adding a term to the loss function. This will cause some of the parameters to shrink towards zero, thus creating a simpler model.

Ridge, or L_2 regularisation, involves adding the L_2 norm of the parameters to the loss function

$$L(\boldsymbol{\theta}, s) = -\left(\sum_{n=1}^{N} y_n \phi\left(\mathbf{x}_i\right) \cdot \boldsymbol{\theta}^T - (1-y_n) \log\left(1+\exp\left(\phi\left(\mathbf{x}_i\right)\boldsymbol{\theta}^T\right)\right)\right) + s||\boldsymbol{\theta}||_2. \tag{3.66}$$

A sparsity parameter s is used to control how much the model complexity will be penalised. With Ridge the parameters are prevented from taking large values.

Lasso, or L_1 regularisation, involves adding the L_1 norm of the parameters to the loss function

$$L(\boldsymbol{\theta}, s) = -\left(\sum_{n=1}^{N} y_n \phi\left(\mathbf{x}_i\right) \cdot \boldsymbol{\theta}^T - \log\left(1+\exp\left(\phi\left(\mathbf{x}_i\right)\boldsymbol{\theta}^T\right)\right)\right) + s||\boldsymbol{\theta}||_1. \tag{3.67}$$

Use of the L_1 norm allows some parameters to become zero during training, thus reducing the model's complexity. This is in contrast to Ridge which generally will not force any parameters to take a zero value, but something close to it. This ability of Lasso makes it also useful as a feature selection method, which is especially important in data sets with a large number of features. However, Lasso tends to select only one feature from a group of highly correlated ones, even if they are all descriptive. For our data set, where linear relationships might arise accidentally and not bear ecological significance, this tendency of the algorithm might mask the contribution of otherwise significant species.

Furthermore, adding an L_1 norm to the loss function means it is not longer differentiable. Therefore, minimisation algorithms have to be employed to get the best parameters.

Choosing one of these regularisation techniques is equivalent to maximising the posterior distribution $P(\boldsymbol{\theta}|X_{train}, \mathbf{y}_{train}, s)$, and choosing an appropriate prior for the parameters. To demonstrate, let's assume that the prior is a multivariate normal distribution with mean zero

and the identity matrix multiplied by a constant as covariance. Then the log posterior can be written as

$$P(\boldsymbol{\theta}|s) = (2\pi)^{-\frac{M}{2}} s^{-\frac{1}{2}} \exp\left(-\frac{\boldsymbol{\theta}\cdot\boldsymbol{\theta}^T}{2s}\right) \tag{3.68}$$

$$\log P(\boldsymbol{\theta}|X_{train}, \mathbf{y}_{train}, s) = \left(\sum_{n=1}^{N} y_n \phi\left(\mathbf{x}_i\right)\cdot\boldsymbol{\theta}^T - \log\left(1+\exp\left(\phi\left(\mathbf{x}_i\right)\boldsymbol{\theta}^T\right)\right)\right) - \frac{\boldsymbol{\theta}\cdot\boldsymbol{\theta}^T}{2s}, \tag{3.69}$$

by ignoring constants, like the marginal likelihood $P(\mathbf{y}_{train}|X_{train}, s)$ and the prior constants that do not depend on the parameters. Therefore, maximising the posterior with multivariate Normal prior is equivalent to minimising the loss with an L_2 regularisation.

Using instead independent Laplace priors centred at zero for each parameter

$$P(\boldsymbol{\theta}|s) = \prod_{i=1}^{M} \frac{1}{2s} \exp\left(\frac{|\theta_i|}{2s}\right), \tag{3.70}$$

and maximising the posterior, is equivalent to minimising the loss with lasso regularisation.

An advantage of the Bayesian method is that we get a distribution over the parameters instead of a single value estimate. This gives us a better idea of how important a feature is and how considerable the effect. A disadvantage is that it needs to sample a large number of values from the posterior to approximate it well. This is more so when the parameters' dimension is larger. Therefore, it is usually a significantly slower algorithm.

The hyperparameters we cross-validated included the sparsity parameter s, the intercept, and a class balance term that gives more weight to the minority class in the loss function.

Linear regression was performed using the package scikit-learn in python [40]. The MCMC approach was carried out using PyMC3, a python package for Bayesian statistical modelling [44].

3.3.2 Random Forest

Random Forest is an ensemble method that works by training multiple decision trees and averaging their predicted class when it comes to classification. The combination of trees is done to reduce the overfitting of single decision trees on the train set [22]. In particular, trees grown very deeply to learn complex patterns in the data have often low bias but very high variance.

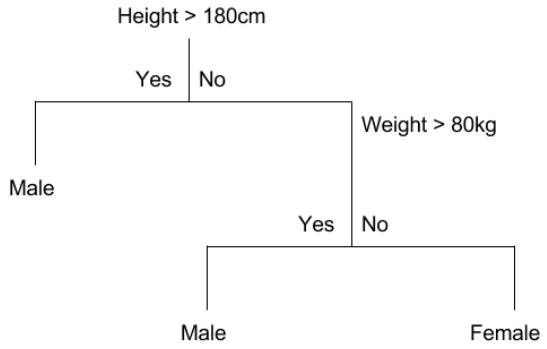


Fig. 3.5 An example of a decision tree constructed for the classification to males and females

To combat this the features and observations used by each tree are randomly sub-sampled from the training set (with or without replacement) [2]. Thus the random forest algorithm averages out deep decision trees trained on different parts of the training set, so as to reduce their variance.

Decision Trees

To illustrate how binary decision trees work, let's consider the classification of humans into male and female using two features: height in centimetres and weight in kilograms. An example of a decision tree applicable for such a problem is given in figure 3.5. The root node splits the features space of height into two regions, one above 180cm and one below. In the first region, the tree classifies all points as male. The second region is further split into two based on weight; above and below 80kg.

Decision trees constructed by Random Forest are done so using the Classification and Regression Trees (CART) algorithm. These are binary trees were each node represents a single input feature and a split condition in that feature's space. The leaf nodes of the trees (end points) contain an output variable (class label). The selection of which features to be used at each node is done using a cost function. The construction of a tree ends when a predefined stopping condition is met.

Our data sets is made up of N response variables $y_i \in \{1,0\}$ and features $\mathbf{x}_i \in \mathrm{R}^p$. Let's also define the regions our feature space is split into by each node of the tree by $R(j,s)$, where j is the splitting feature and s is the split point. The pair of regions is given by

$$R_1(j,s) = \{X|X_j \leq s\} \text{ and } R_2(j,s) = \{X|X_j > s\}. \tag{3.71}$$

[2] Sampling only the observations with replacement is called bagging (bootstrap aggregating)

The proportion of class $k \in \{1,0\}$ observations in region m which has a total of N_m observations is given by

$$\hat{p}_{m,k}(j,s) = \frac{1}{N_m} \sum_{\mathbf{x}_i \in R_m(j,s)} I(y_i = k), \tag{3.72}$$

where $I(y_i = k)$ is the indicator function (equals to one when $y_i = k$, zero otherwise) and the sum is over all the points which lie in region R_m. The proportion depends on the splitting feature and point (j,s) of the region R_m.

To create a tree we need to divide up the feature space; a greedy approach called recursive binary splitting is used. It involves searching over all available features (to that node) and all split points to find the split (or regions) with the lowest cost. To create this cost function, impurity measures are used, which quantify how pure a node is. One commonly used measure is the Gini index which takes the form

$$G_m = 2\hat{p}_{m,1}(1 - \hat{p}_{m,1}), \tag{3.73}$$

in region m for binary classification. When samples at a node (or region) belong with equal proportions to both classes (thus a bad/impure split), the index takes the value 0.5. When the node has samples in only one class it takes the value 0 (a perfect/pure split).

To construct the cost function we need to take into consideration the number of observations in the two child nodes $N_{mL},\ N_{mR}$ created by splitting the m node, and weight the corresponding node impurity measure. This gives the cost function of node m

$$\Delta G_m = G_m - G_{mL}\frac{N_{mL}}{N_m} - G_{mR}\frac{N_{mR}}{N_m} \tag{3.74}$$

$$= 2\hat{p}_{mL,1}(j,s)(1 - \hat{p}_{mL,1})\frac{N_{mL}}{N_m} + 2\hat{p}_{mR,1}(1 - \hat{p}_{mR,1})\frac{N_{mR}}{N_m}, \tag{3.75}$$

which depends on the choice of region m, and consequently on the split feature and point (j,s). The subscripts mL and mR denote the two child nodes created after the split, and $\hat{p}_{mR,1}$ the proportion of samples in class 1 in the right node. This cost function is called 'decrease in Gini impurity' at node m.

The feature (from those available to the tree) and split point that maximises the decrease in impurity ΔG_m are used to construct child nodes, increasing the depth of the tree.

The most common stopping criterion is to set a minimum number of observables that each node needs to satisfy before it is further split. If the count is less than that then the node becomes a leaf node.

Aggregating

As mentioned earlier, trees used in random forest are not trained on the whole data set. In particular, bagging is usually employed so that each tree fits a model on a different set of observables. Furthermore, at each node of a tree, a predetermined number of features are sampled without replacement; only these are considered in the splitting condition. Usually this number is set to the square root of the total number of features in the train set (rounded up).

When it comes to prediction, the output class probability predicted by each tree $T_i(\mathbf{x})$ is averaged over all B trees to give the class probability of the ensemble

$$T_i(\mathbf{x}_{test}) = P(C = 1|\mathbf{x}_{test})_i \tag{3.76}$$

$$= \text{Proportion of class 1 observables in leaf node reached by } \mathbf{x}_{test} \tag{3.77}$$

$$P(C = 1|\mathbf{x}_{test}, \{T_i\}_1^B) = \frac{1}{B}\sum_{i=1}^{B} T_i(\mathbf{x}_{test}). \tag{3.78}$$

The construction of the algorithm allows it to determine which features offer most of the explanatory power in separating samples. There are several ways to do this and none of them outperforms the other in all cases (since explanatory power is a difficult concept to define). One approach uses the out-of-bag samples that were not used in the construction of a tree (if bagging was used). Its prediction accuracy on these samples is measured and used as a benchmark for feature evaluation. Then the values for a feature in the out-of-bag data set are randomly permuted, and the prediction accuracy of the tree is tested again. The drop in accuracy is associated with the predictive power of the feature whose values where permuted. This is done for all features, and the loss in accuracy is averaged over all trees. The larger the drop, the more predictive the feature.

Another way is to measure the mean decrease in impurity of each feature. For a feature X_j this is done by adding up the weighted decrease in impurity for all nodes where X_j is used, and then average over all trees in the forest. The formula is given by

$$Importance(X_j) = \frac{1}{B}\sum_{b=1}^{B}\sum_{t\in\phi_b} I(j_t = j)\left[prop(t)\Delta G_t\right], \tag{3.79}$$

where the sum is over all trees b and all nodes of the tree ϕ_b, and $I(j_t = j)$ is the indicator function that takes the value 1 when node t is split using variable X_j. The proportion of samples, from the initial number, at node t is given by $prop(t)$ [.] **Gilles**

The random forest algorithm was implemented using scikit-learn in python. The gini impurity measure is used both for tree growing and for feature importance (using also trees' depth as an indicator). The number of trees, the stopping criterion and whether to bootstrap the observations were cross validated.

3.4 Hamiltonian Monte Carlo

3.4.1 Motivation

Hamiltonian Monte carlo (HMC) is a method for sampling from the distribution of interest, using properties of Hamiltonian dynamics. Because of the way new states are proposed in HMC, it can explore the state space faster (with less samples) than if a Metropolis-Hastings sampler was used.

Usually the quantities we wish to evaluate in Bayesian Inference are the expectations of a function f over the density of interest $p(q)$ in the parameter space $\mathscr{Q}$

$$E_p(f) = \int_{\mathscr{Q}} p(q)f(q)dq. \tag{3.80}$$

As was discussed previously, these integrals can be analytically intractable for any non-trivial target distribution, and that is why numerical methods are employed to approximate them. If the variation of the function of interest does not strongly affect the inegral, we might assume that most of it comes from the neighbourhood around the mode of the target density (where it is maximised).

This assumption however is misplaced, as the integral is calculated by accumulating the integrand over a volume of parameter space. The volume plays an increasingly important role the more dimensions of the parameter space we consider. This can be better understood by partitioning a parameters space into rectangles centred around the mode of the distribution.

Figure 3.6 summarises the effect of increasing dimensions to the weight of the mode. In one dimension, there are two partitions of our space, a line to the left and to the right of the mode. Thus it makes up for $1/3$ of the volume. In two dimensions we can fit eight squares/partitions around the mode, where it accounts for $1/9$th of the volume. In 3 dimensions, there are 26 partitions adjacent to the mode. If instead we consider the volume outside this neighbourhood of cubes, we find even more volume.

We can conclude that most of the volume exists in the tails of the distribution of interest, far away form the mode, and it grows exponentially as the dimensions increase. Densities on

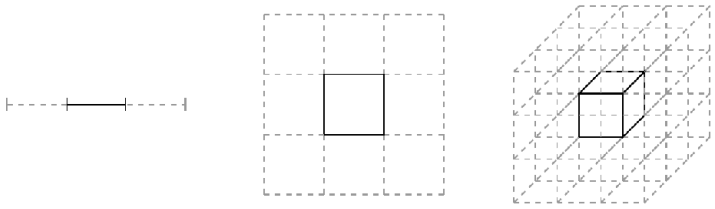


Fig. 3.6 We partition the neighbourhood of the mode with rectangular regions. As the dimensions of the parameter space increase, the volume of the mode accounts for less of the total volume of its neighbourhood regions [6].

the other hand have exactly the opposite property, since they have to integrate to unity (or a constant).

Therefore, the regions of parameter space which contribute most to the expectation are not found either close to the mode, or very far away from it. These regions are usually called the typical set, which is found around the mode, and which is what we want to explore with an MCMC sampler.

The method we need to explore the typical set should do it fast enough and not get stuck at pathological points of the distribution. The Metropolis-Hastings algorithm, given enough time, will explore most of the space of interest. However, because of the necessary exploration of the parameter space in states of low probability density, and the way the algorithm proposes and accepts new samples, the markov chain sometimes gets stuck and rarely moves. Because of limited computational power, a new method is needed that explores the typical set much faster.

This is where Hamiltonian dynamics becomes useful. The problem of exploring the typical set can be recast into a physics problem; the orbit of a satellite around a planet. The gravitational potential energy exerted by the planet can be viewed as the target density. Its gradient points towards the planet and thus going along it will cause the satellite to crash. To keep it in orbit we have to pump it with enough momentum to keep it going around the planet, but also not too much so that it flies away to the depths of space. Because of the conservative dynamics of this system, giving the satellite just enough energy at the start of the orbit will keep it at approximately the same zone.

The key to exploring the typical set, is thus by introducing auxiliary momentum parameters to our probabilistic system. They have to be added, however in such a way that the conservative dynamics of the system are ensured.

3.4.2 Formulation

The Hamiltonian of a system is completely defined by its position q_i and momentum p_i, and it describes how these variables change with time

$$\begin{aligned}\frac{dq_i}{dt} &= \frac{\partial H}{\partial p_i} \\ \frac{dp_i}{dt} &= -\frac{\partial H}{\partial q_i}\end{aligned} \tag{3.81}$$

It can be expressed as the sum of the potential $U(q)$ and kinetic $K(p)$ energy of the system

$$H(q,p) = U(q) + K(p), \tag{3.82}$$

and thus the equations of motion can be rewritten to

$$\begin{aligned}\frac{dq_i}{dt} &= \frac{\partial K}{\partial p_i} \\ \frac{dp_i}{dt} &= -\frac{\partial U}{\partial q_i}\end{aligned} \tag{3.83}$$

The dynamics of the system satisfy some important properties without which the construction of valid MCMC updates would not be possible. First of all, the Hamiltonian of the system is time invariant

$$\frac{dH}{dt} = \sum_{i=1}^{d}\left[\frac{dq_i}{dt}\frac{\partial H}{\partial q_i} + \frac{dp_i}{dt}\frac{\partial H}{\partial p_i}\right] = \sum_{i=1}^{d}\left[\frac{\partial H}{\partial p_i}\frac{\partial H}{\partial q_i} - \frac{\partial H}{\partial q_i}\frac{\partial H}{\partial p_i}\right] = 0. \tag{3.84}$$

Another fundamental property is that the dynamics are reversible; there is a one-to-one map from the state $(q(t), p(t))$ at time t to the state $(q(t+s), p(t+s))$ at time $t+s$, and thus the reverse map exists as well. Finally, the phase space's (q,p) volume is conserved, therefore, any mapping from one region in the phase space to another must conserve the initial volume.

The introduction of the auxiliary momentum variables transforms the target distribution into a joint one in phase space

$$P(q,p) = P(p|q)P(q) = P(p)P(q). \tag{3.85}$$

The joint density is equal to the probability of momentum given the position times the target distribution. This can be further simplified to the product of the two marginals, since we do not assume any dependence of the momentum to the position. The joint probability is related

to the Hamiltonian via a canonical distribution

$$P(q,p) = \frac{1}{Z} exp\left(-H(q,p)\right), \tag{3.86}$$

where Z is the normalising constant. Therefore we can rewrite the density as

$$P(q,p) = \frac{1}{Z} \exp\left(-U(q)\right) \exp\left(-K(p)\right). \tag{3.87}$$

Under this formulation, we can see how the potential energy $U(q)$ can be viewed as the negative log of the target density

$$U(q) = -\log(p(q)). \tag{3.88}$$

The kinetic energy on the other hand can take the form

$$K(p) = \frac{p^T M^{-1} p}{2}, \tag{3.89}$$

where M is a symmetric positive-definite matrix which is usually a diagonal multiplied by a constant. This is the usual definition of kinetic energy in physics, but under our probabilistic perspective (and the canonical distribution) it can be viewed as a multivariate Gaussian distribution over the momentum particles, centred at zero and with covariance matrix M. The partial derivative of the kinetic energy with respect to momentum p_i is given by

$$\frac{\partial K(p)}{\partial p_i} = [M^{-1} p]_i = \frac{p_i}{m_i}. \tag{3.90}$$

The last equality is given when the covariance (mass) matrix is diagonal with m_i as its ith diagonal element.

Any trajectory in the phase space satisfying Hamilton's equations and the dynamics of the system leaves the joint probability density constant. Therefore, to draw samples of position from a previously unexplored region of phase space, we could generate a random momentum vector and use the previous value of the position. Then we could evolve the state (q,p) using Hamilton's equations (3.81) until we reach the desired space, and accept the new samples. The process can be repeated by generating a new random momentum vector.

The problem with this method is that Hamilton's equations must be approximated by discretising them in time. Then the evolution of states is from time t $(q(t), p(t))$ to time $t+\varepsilon$ $(q(t+\varepsilon), p(t+\varepsilon))$, and whatever numerical method we use to do this, will not be perfect and have some numerical errors. The method most commonly used is the leapfrog which

evolves in time the momentum variable first by half a step, then the position by a whole step using the evolved momentum, and finally completes the evolution of the momentum by one more half step

$$p_i(t+\varepsilon/2) = p_i(t) - (\varepsilon/2)\frac{\partial U}{\partial q_i}(q(t)) \tag{3.91}$$

$$q_i(t+\varepsilon) = q_i(t) + \varepsilon\frac{p_i(t+\varepsilon/2)}{m_i} \tag{3.92}$$

$$p_i(t+\varepsilon) = p_i(t+\varepsilon/2) - (\varepsilon/2)\frac{\partial U}{\partial a_i}(q(t+\varepsilon)). \tag{3.93}$$

Because of numerical approximation errors, the new state q^* found by the evolution of the Hamiltonian using the leapfrog method is accepted with a probability of

$$\text{acceptance probability of new state} = \min(1, \exp(H(q,p) - H(q^*,p^*))). \tag{3.94}$$

If the proposed state is not accepted then the next state is set to the current one.

The algorithm starts with a specified initial set of parameters q, that can be predetermined or randomly generated. For every iteration of the algorithm new momentum p parameters are generated and, together with the current q, are updated according to Hamiltonian dynamics using the leapfrog method performed L number of times with discretisation time ε. A metropolis acceptance step is then applied with probability (3.94); if the new state q^* of the parameter is accepted then it is appended to the markov chain and set as the current q. If the new state is not accepted then the current q is unchanged and appended again to the markov chain.

A crucial part of the method is the random generation of a new momentum vector at each iteration. It is what moves the chain to (q,p) points with different joint density, thus allowing for state phase exploration. The generation of new momentum for each step can change the density by a large amount, thus producing q values with a much different density.

A more extensive treatment and some theoretical results can be found in Chapter 5 of the MCMC Handbook [35]. For a conceptual introduction to the matter see [6].

Chapter 4

Results and Discussion

In this chapter we present and comment upon the performance of the classifiers under the different split schemes outlined earlier. The various features used and their applicability on different splitting conditions is also to be evaluated.

The confusion matrix used throughout this chapter has the form

$$\begin{bmatrix} \text{True Black/ Black Predicted correctly} & \text{False White/ Black Predicted falsely} \\ \text{False Black/ White Predicted falsely} & \text{True White/ White Predicted correctly} \end{bmatrix} \tag{4.1}$$

The results of the benchmark method we devised in section 2.2 are presented in table 4.1 for maximum similarity, table 4.2 for maximum dissimilarity, and table 4.3 for random splits. The two row names differentiate between the set of response variables used for the benchmark's evaluation; All Labels indicates the use of all the river samples, Min Labels of those whose total OTU read-count is more than 10000. The accuracy is also included in a separate row so as to make comparisons between methods easier. A quick glance shows that the benchmark has a much lower accuracy than the class prior of the data set (87.2%), as was suggested in section 2.2.

Table 4.1 Benchmark for Maximum Similarity

	Confusion Matrix		Accuracy
Features used	Predicted Black	Predicted White	
All Labels	2.64 18.36	18.36 124.64	77.6%
Min Labels	2.75 18.25	18.25 117.75	76.8%

Furthermore, the benchmark is lower in the maximally dissimilar case. This is expected since black water samples are concentrated in only some rivers, and calculating the prior on a set excluding them will lead to more errors in prediction. It is also important to note that mistakes are made disproportionally in the direction of 'False White' (black samples predicted as white).

Table 4.2 Benchmark for Maximum Dissimilarity

	Confusion Matrix		Accuracy
Features used	Predicted Black	Predicted White	
All Labels	1.75 18.96	19.25 124.04	76.7%
Min Labels	1.83 18.83	19.17 117.17	75.8%

Table 4.3 Benchmark for Maximum Dissimilarity

	Confusion Matrix		Accuracy
Features used	Predicted Black	Predicted White	
All Labels	2.69 18.31	18.31 124.69	77.7%
Min Labels	2.81 18.19	18.19 117.81	76.8%

4.1 Maximum Similarity

The results for Logistic regression (with L_1 regularisation) and Random forest tested on the maximum similarity scheme are shown in tables 4.4 and 4.5 respectively. Both methods performed relatively well when compared to the baseline. Furthermore, the PCoA features produced poorer results than the OTU ones. This was also the case when fewer dimensions, describing 99% and 90% of the variance, were chosen as features. The results for these can be found in the appendix in tables A.3 and A.2. The results for PCA and a 20-dimensional NMDS configuration were also included there and not in this Chapter. The NMDS method failed to converge to a minimum stress.

Using the F score to find the best model in the cross validation step resulted in a lower number of black water samples correctly identified. This was especially the case for Random Forest. Therefore, we present the performance of the models selected using accuracy.

Just from the accuracy score we can see that Logistic regression outperforms Random Forests. The best score for the former is 98.78% using OTU LOW and close second is 98.17% using OTU, OTU CSS, and OTU CSS LOG. For Random Forests, OTU CSS LOG has the best score of 96.95% and close second is OTU CSS and OTU LOW with 96.34%. When using the L_2 penalty, the algorithm had lower accuracy except for OTU CSS LOG, where it attained a score of 98.78%. The results for L_2 will not be presented in a table but mentioned when they outperform L_1. Both methods have significantly better results than the baseline's 77.6% and 76.8% accuracy (the second is used to compare the OTU MIN CSS set).

It is also interesting that Random Forest has an equal or lower Recall for Black water samples for all feature sets. In other words, given that a sample comes from black waters, the classifier predicts it as such with a lower probability than Logistic regression.

Using the L_1 penalty in Logistic regression has reduced the coefficients of many features to zero. From the best performing sets, the percentage of zero coefficients to the total number is 82.34% (OTU LOW), 85.29% (OTU), 88.95% (OTU CSS), and 95.32% (OTU CSS LOG). From the remaining species, 3.79% of them have non zero coefficients in three sets (OTU CSS, OTU CSS LOG, and OTU). This means that there are only a limited number of informative species that help us identify the water colour.

Table 4.4 Results from maximising similarity using Logistic Regression

Features used	Confusion Matrix		Accuracy
	Predicted Black	Predicted White	
OTU	19 1	2 142	98.17%
OTU LOW	19 0	2 143	98.78%
OTU CSS	18 0	3 143	98.17%
OTU Min CSS	18 0	3 136	98.09%
OTU CSS LOG	19 1	2 142	98.17%
PCoA Bray-Curtis	16 3	5 140	95.12%
PCoA Bray-Curtis CSS	16 1	5 142	96.34%

Table 4.5 Results from maximising similarity using Random Forest

Features used	Confusion Matrix Predicted Black	 Predicted White	Accuracy
OTU	17 3	4 140	95.73%
OTU LOW	18 3	3 140	96.34%
OTU CSS	18 3	3 140	96.34%
OTU Min CSS	18 3	3 133	96.18%
OTU CSS LOG	19 3	2 140	96.95%
PCoA Bray-Curtis	6 4	15 139	88.41%
PCoA Bray-Curtis CSS	4 0	17 143	89.63%

To explore which of these species contribute most to classification we used the feature importance of Random Forest. However, instead of checking only for individual species' predictive ability, we used the taxonomic order as an aggregating factor, to get a sense which contributes the most. This was done by averaging the importance of each species, as reported by Random Forests, within their Order. The results are presented for some feature sets in the form of pie charts in Figures 4.1a and 4.1b. For other features were this evaluation of taxonomic importance was possible are presented in the Appendix, in Figures A.1a and A.1b.

For all feature sets used, species in the *Perissodactyla* Order have on average the most explanatory power. However, this Order is only composed of two (sub)species; the South American and Mountain tapir. This method of aggregating feature importance thus favours orders with few species. It is evident in the next most important taxonomic Order as well, the *Cetacea*, which is again composed of two species; the Amazon river dolphin (which is technically a whale) and Tucuxi (a freshwater dolphin).

Therefore, another aggregating method has to be used to get a better idea of the explanatory power of the whole taxonomic Order. We chose to do this by summing the importance of each species, within their Order. The results are shown in Figures 4.1c and 4.1d (A.1c and A.1d in the Appendix), and highlight different Orders. In particular, *Characiformes* and *Siluriformes* are in sum the most explanatory ones. This is to be expected however, since

they are the most populous as well (although in reverse order) and summing importance will inevitably favour the most numerous groups.

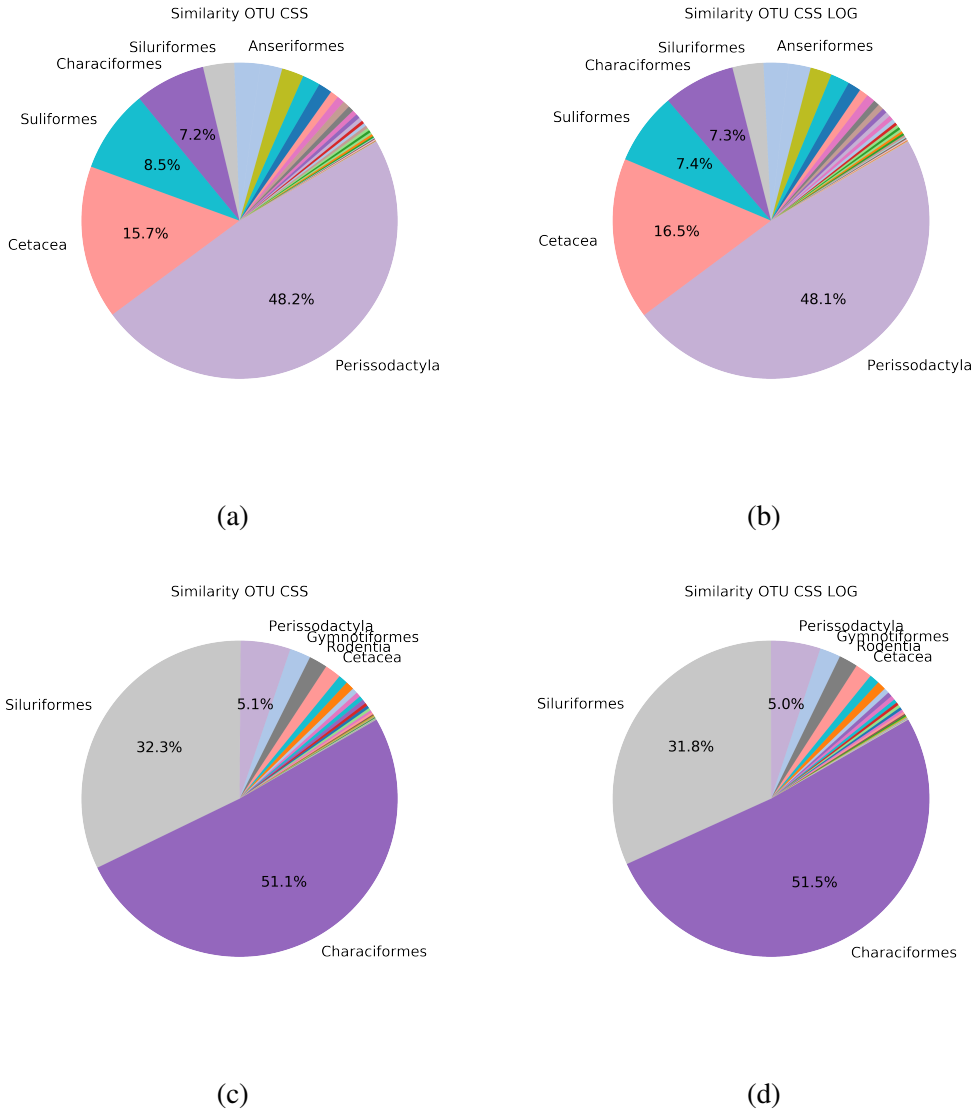


Fig. 4.1 Species' importance per taxonomic order as calculated by Random Forest in the maximum similarity test. Averaging the importance for the sets: OTU CSS 4.1a, and OTU CSS LOG 4.1b. Summing the importance for the sets: OTU CSS 4.1c, and OTU CSS LOG 4.1d.

Conclusions can be drawn despite the seemingly contradictory representations of the two methods. This is because all four important Orders are in the top 6 in both methods, and when combined they amount for a substantial part of explanatory power (more than 70%).

4.2 Maximum Dissimilarity

The classifiers' performance in a maximum dissimilarity setting is worse than in maximum similarity. The results are summarised in tables 4.6 and 4.7 for Logistic regression and Random Forests respectively. In this setting, Random forests have on average a higher accuracy score. The highest is obtained with OTU MIN CSS (90.45%), but OTU CSS and OTU CSS LOG come close (90.24%). Logistic regression has the highest accuracy with PCoA CSS and OTU CSS LOG (87.80%), and with PCoA coming close (86.58%). If we consider only some of the axes of PCoA and PCoA CSS, the method attains an even higher accuracy (see table A.3 in the Appendix). Using the L_2 norm instead, the classifier reaches 93.90% accuracy with OTU CSS LOG, with 9 true blacks and 1 true white being wrongly classified.

Some of the feature sets had an accuracy score close to the baseline, meaning that the classifiers were not significantly better than random guessing. Furthermore, both classifiers had black water Recall scores lower than 0.5 for all of their feature sets, and made a disproportionate amount of errors in classifying black samples. Results for the PCoA sets explaining 99% and 90% of the variance, for PCA, and for the NMDS configuration, are given in tables A.3 and A.4 of the Appendix.

Again, the L_1 penalty reduced many coefficients to zero; among the best performing sets, PCoA CSS had 88.33% of it's coefficients go to zero, OTU CSS LOG had 95.58%, and PCoA 90.78%. The sets OTU, OTU CSS, and OTU CSS LOG shared 3.05% of their non zero coefficients. This means that once again only a limited number of OTUs contribute to the classification.

The feature importance method of Random Forest was again employed to explore which species contributed most to their predictive power. Averaging the importance within taxonomic Order for OTU CSS and OTU CSS LOG is shown in Figures 4.2a and 4.2b respectively. For the feature sets OTU and OTU MIN CSS the Figures are A.2a and A.2b respectively, presented in the Appendix. As was the case for the maximum similarity setting, *Perissodactyla* is on average the Order with the most explanatory species, with *Cetacea* coming second.

Summing instead of averaging produces the same result seen previously in the maximum similarity setting. Once again *Characiformes* are in sum the most explanatory Order, and *Siluriformes* the second most explanatory. The pie charts for OTU CSS and OTU CSS LOG can be seen in Figures 4.2c and 4.2d, for OTU and OTU MIN CSS in Figures A.2c and A.2d in the Appendix.

The same conclusions are drawn for this setting as well; all four taxonomic Orders are within the six most explanatory ones in both aggregating methods. This suggests that they are the most important when it comes to prediction.

Table 4.6 Results from maximising dissimilarity using Logistic Regression

Features used	Confusion Matrix Predicted Black	Confusion Matrix Predicted White	Accuracy
OTU	5 18	16 125	79.27%
OTU LOW	5 17	16 126	79.88%
OTU CSS	7 14	14 129	82.93%
OTU Min CSS	3 14	18 122	79.61%
OTU CSS LOG	5 4	16 139	87.80%
PCoA Bray-Curtis	7 8	14 135	86.59%
PCoA Bray-Curtis CSS	9 8	12 135	87.80%

4.3 Random Splits

The classifiers' performance under the Random splitting setting is between than of maximum similarity and dissimilarity. The results are summarised in tables 4.8 for Logistic regression and 4.9 for Random Forest. Both methods have accuracy scores significantly better than the baseline.

The best features set for Logistic regression is OTU CSS LOG with 96.34% accuracy. For Random Forest it is OTU CSS LOG and OTU CSS with a score of 95.73%. Neither classifier performs better than the other consistently for all features sets.

Overall, when taking into consideration all the splitting settings, the features set that produces on average the highest accuracy score is OTU CSS LOG. Furthermore, it seems that the sets produced using PCoA fair better when used in Logistic regression, whereas the NMDS configuration is better used by Random Forest (see Appendix). PCA (or PCoA with a euclidean metric) performs generally poorly for all classifiers; usually it has the lowest accuracy, but it always has the lowest black water recall score, classifying almost all the samples as white.

Table 4.7 Results from maximising dissimilarity using Random Forest

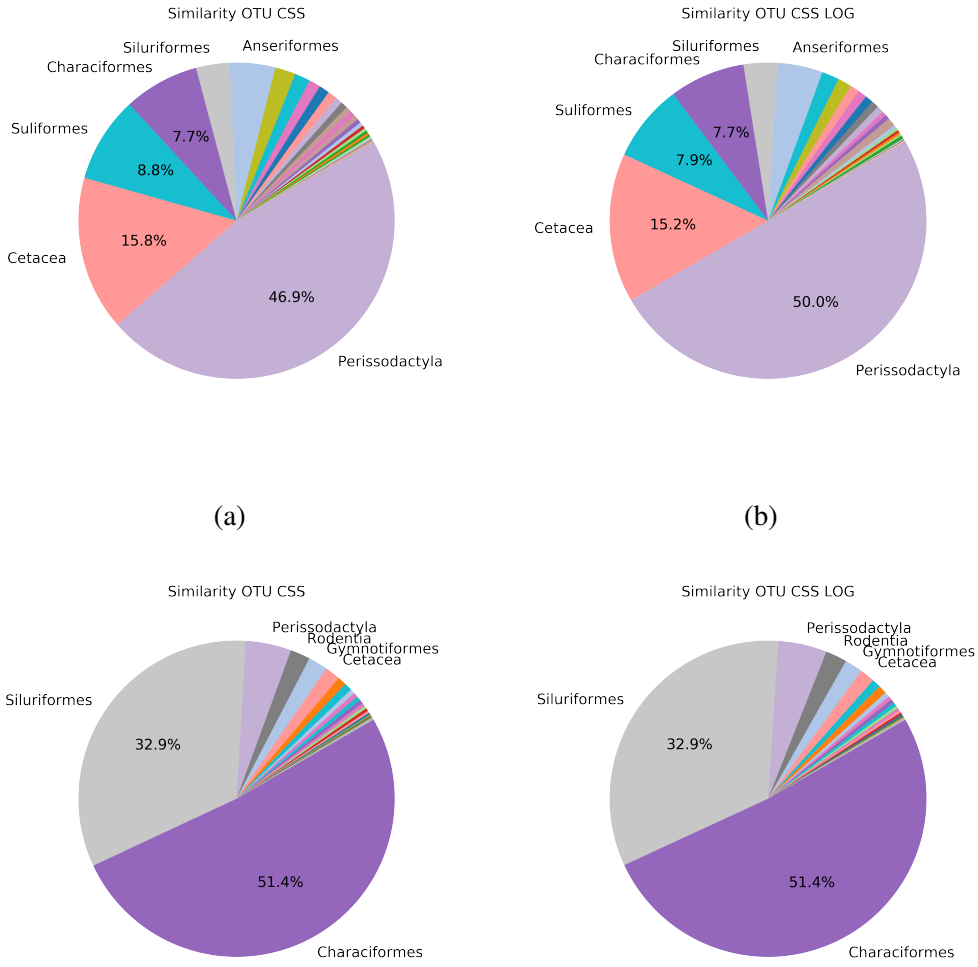
Features used	Confusion Matrix		Accuracy
	Predicted Black	Predicted White	
OTU	4 10	17 133	83.50%
OTU LOW	2 10	19 133	82.32%
OTU CSS	7 2	14 141	90.24%
OTU Min CSS	7 1	14 135	90.45%
OTU CSS LOG	7 2	14 141	90.24%
PCoA Bray-Curtis	0 1	21 142	86.59%
PCoA Bray-Curtis CSS	0 5	21 138	84.15%

4.4 Bayesian vs MLE Logistic Regression

As mentioned previously, a downside of MCMC methods is their need for a large number of samples, especially when the parameter space is high dimensional. This is the case for our model as well, which needed almost one hour to get 5000 samples from the posterior, when using the OTU features set. The parameter space was even then not explored adequately and the prediction accuracy was lower than the benchmark.

That is why we will be presenting the results of training a logistic regression model on the NMDS features set; it has the smallest number of features from all others in consideration. Moreover, cross-validating the model takes prohibitively long; for each train-test split we fit 240 models to choose the best hyperparamters (6 validation folds times 60 combinations of hyperparameters). Therefore, we choose to present a model with an intercept parameter and $s = 1$ as the sparsity. We also show results for two priors over the parameters, Gaussian and Laplacian.

The results for testing under the maximum similarity setting (without cross-validation) are presented in table 4.10. The feature set used is NMDS with L_1 and L_2 denoting the use of a Laplacian and Gaussian priors. Compared to the L_1 MLE implementation (see table A.3), the MCMC method achieves a higher accuracy score even without cross-validation (using



(a) (b)

(c) (d)

Fig. 4.2 Species' importance per taxonomic order as calculated by Random Forest in the maximum dissimilarity test. Averaging the importance for the sets: OTU CSS 4.2a, and OTU CSS LOG 4.2b. Summing the importance for the sets: OTU CSS 4.2c, and OTU CSS LOG 4.2d.

Laplacian prior). Using a ridge penalty we get an accuracy score of 95.12% using the MLE method, which is higher than that obtained through the MCMC.

We can also check if the two methods reduced the same features' coefficients to zero. To do this we arbitrarily choose the L_1 model (and its parameters) which trained on the first split of our procedure. We plot the posterior distribution of the parameters in Figure 4.3; the beta terms denote the parameters $\boldsymbol{\theta}$. On the same plot we show with red the coefficients

Table 4.8 Results from random splits using Logistic Regression

Features used	Confusion Matrix Predicted Black	Confusion Matrix Predicted White	Accuracy
OTU	18 6	3 137	94.51%
OTU LOW	17 9	4 134	88.41%
OTU CSS	16 16	5 127	87.20%
OTU Min CSS	15 22	6 114	82.17%
OTU CSS LOG	18 6	3 137	96.34%
PCoA Bray-Curtis	14 7	7 136	91.46%
PCoA Bray-Curtis CSS	11 9	10 134	88.41%

of features obtained by an MLE approach of the same model (sparsity parameter and L_1 regularisation). Most coefficients do not line up exactly, but some are very close. Even though it is slower, the Bayesian approach is much richer in terms of information gained with regards to the parameters' distribution.

Comparing the MCMC method, table 4.11, in the maximally dissimilar setting with the MLE, we see that the L_1 regularisation has similar performance for both. When we use L_2 however, the MLE's score is 84.15%, much lower than that obtained through MCMC.

It is also noteworthy that Bayesian logistic regression using a Gaussian prior and the NMDS features set outperformed almost all other classifiers in the maximally dissimilar case (MLE logistic regression with OTU CSS LOG has an accuracy score of 93.90%).

It is possible that with much more MCMC samples and time, the Bayesian approach would be able to achieve an even greater accuracy by utilising the much larger feature sets, like OTU CSS LOG.

Table 4.9 Results from random splits using Random Forest

Features used	Confusion Matrix Predicted Black	Confusion Matrix Predicted White	Accuracy
OTU	15 2	6 141	95.12%
OTU LOW	15 10	6 143	90.24%
OTU CSS	17 3	4 140	95.73%
OTU Min CSS	17 3	4 133	95.54%
OTU CSS LOG	17 3	4 140	95.73%
PCoA Bray-Curtis	4 7	17 136	85.37%
PCoA Bray-Curtis CSS	4 2	17 141	85.37%

Table 4.10 Results from Maximum Similarity using MCMC Logistic Regression

Features used	Confusion Matrix Predicted Black	Confusion Matrix Predicted White	Accuracy
NMDS L_1	13 0	8 143	95.12%
NMDS L_2	8 0	13 143	92.07%

Table 4.11 Results from Maximum Dissimilarity using MCMC Logistic Regression

Features used	Confusion Matrix Predicted Black	Confusion Matrix Predicted White	Accuracy
NMDS L_1	3 9	18 134	83.54%
NMDS L_2	0 21	21 140	92.07%

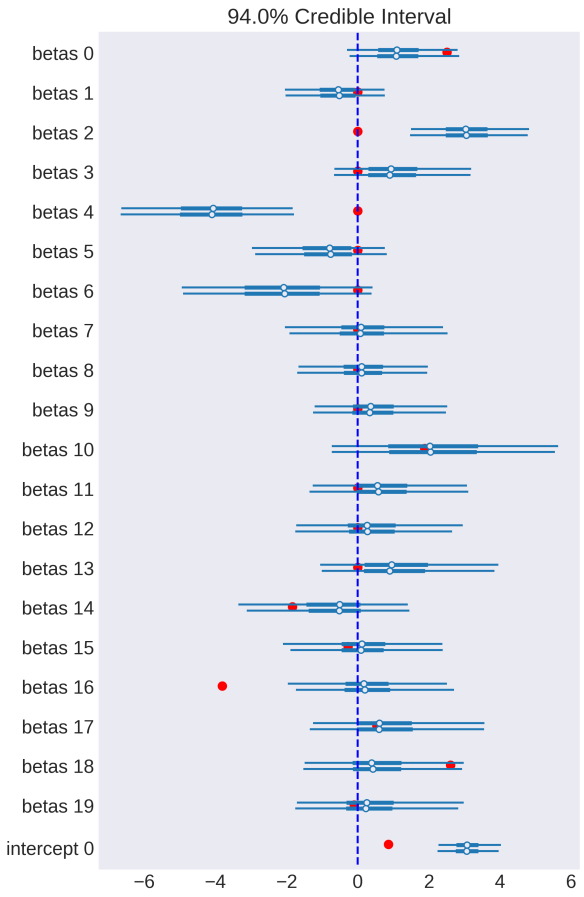


Fig. 4.3 A forest plot of the parameters and their 94% credible interval. Red points denote the coefficients obtained through the MLE approach.

Chapter 5

Conclusion

The three splitting settings devised earlier highlight different classifiers. In the maximum similarity setting, which ensured that observation from all parts of the rivers were included in the train, validation, and test sets, L_1 Logistic regression outperformed Random Forest. It achieved accuracy scores above 98% with five features sets.

In the maximum dissimilarity setting, which kept out whole parts of the rivers for testing and validation, Random Forest proves to be on average the better model. It has an accuracy score of 90.45% when using the OTU MIN CSS set and 90.24% with OTU CSS and OTU CSS LOG. Logistic regression has relatively low scores, with most models being close to naive guessing. Using L_2 regularisation however, we get an accuracy score of 93.90% using the OTU CSS LOG set, outperforming Random Forest.

Randomly splitting the data (but ensuring that the balance of classes was constant between the train, test, and validation sets) did not favour any method. Both performed equally well, with scores ranging between the maximum similarity and dissimilarity settings.

In all cases the classifiers (with best features set) produce better results than naive guessing, even though most of the prediction errors are concentrated in identifying black water samples.

Using Ordination methods as features produces only one relatively (to other sets) high accuracy score, for Logistic regression in the maximum dissimilarity setting. This is achieved by selecting the subset of its axes explaining 90% of the variance in the data. In all other cases, the methods are not useful for dimensionality reduction. Furthermore, using the 20-dimensional NMDS configuration does not produce significant results, and it is highly unlikely that if it had converged better scores would be produced. On the other hand, using the OTU CSS LOG set we obtain the highest average accuracy score across all classifiers and split schemes.

Finally the Bayesian logistic regression framework takes a very long time to sample from posteriors where the parameter's dimension is very large (i.e. all OTU and PCoA sets). Using the much smaller NMDS set however, we get promising result; it is comparable with the cross-validated MLE logistic regression model in the maximally similar setting and outperforms it in the dissimilar.

Further Work

This work showcases that further machine learning exploration of this particular data set, but also of any other derived from metabarcoding eDNA methods, can be very fruitful but also challenging. As a next step, species importance could be identified using parametric models often used in ecology (like zero-inflated Gaussian mixture) and check if they agree with those obtained through random forest.

More classifiers and dimensionality reduction techniques that take into account the spatial distribution of samples could also be employed. Furthermore, other sampling approaches can be developed that take into account correlation structures of river samples, and thus evaluate where each classification method fails.

Problems were encountered because of the unbalanced class distribution. Collecting more samples from the rivers further to the east is one way to combat this, which will also aid in the evaluation of the classifiers.

Bayesian methods (or versions of), even linear ones like Logistic regression, might be promising if more time is spend on sampling and choosing the right models for our data set.

Finally, the scope does not have to be limited to water colour classification. The effect of anthropogenic and environmental factors on the Amazonian community composition could be investigated. The process can begin by additional sampling efforts along and around the rivers, collecting data on other ecological variables, like minerals, pollution levels, river size, water flow, proximity to settlements, and land use, to name a few. Data can also be collected on a temporal basis so that the changes can be better understood. Then, time series analysis together with machine learning could be utilised to uncover the complex interactions between species abundance and their environment, and thus the role humans play in the ecosystem.

References

[1] "A human gut microbial gene catalogue established by metagenomic sequencing". In: *Nature* 464.7285 (2010), p. 59.

[2] Marti J. Anderson. "Permutational multivariate analysis of variance A computer program". In: 2005.

[3] Laure Apothéloz-Perret-Gentil et al. "Taxonomy-free molecular diatom index for high-throughput eDNA biomonitoring". In: *Molecular Ecology Resources* 17.6 (Nov. 1, 2017).

[4] Donald J. Baird and Mehrdad Hajibabaei. "Biomonitoring 2.0: a new paradigm in ecosystem assessment made possible by next-generation DNA sequencing". In: *Molecular Ecology* 21.8 (2012). (Visited on 08/20/2019).

[5] Donald J. Baird and Mehrdad Hajibabaei. "Biomonitoring for the 21st Century: new perspectives in an age of globalisation and emerging environmental threats". In: *Molecular Ecology* 32.8 (2012), pp. 2039–2044.

[6] Michael Betancourt. "A Conceptual Introduction to Hamiltonian Monte Carlo". In: *arXiv:1701.02434 [stat]* (Jan. 9, 2017). URL: http://arxiv.org/abs/1701.02434 (visited on 08/29/2019).

[7] Núria BONADA et al. "Developments in Aquatic Insect Biomonitoring: A Comparative Analysis of Recent Approaches". In: *Annual review of entomology* 51 (Feb. 2006), pp. 495–523.

[8] A Borja, J Franco, and V Pérez. "A Marine Biotic Index to Establish the Ecological Quality of Soft-Bottom Benthos Within European Estuarine and Coastal Environments". In: *Marine Pollution Bulletin* 40.12 (Dec. 1, 2000), pp. 1100–1114.

[9] James H. Bullard et al. "Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments". In: *BMC bioinformatics* 11 (Feb. 18, 2010), p. 94.

[10] James L. Carter, Vincent H. Resh, and Morgan J. Hannaford. "Chapter 38 - Macroinvertebrates as Biotic Indicators of Environmental Quality". In: *Methods in Stream Ecology (Third Edition)*. Ed. by Gary A. Lamberti and F. Richard Hauer. Academic Press, Jan. 1, 2017, pp. 293–318. ISBN: 978-0-12-813047-6. (Visited on 08/23/2019).

[11] Anthony A. Chariton et al. "Ecological assessment of estuarine sediments by pyrosequencing eukaryotic ribosomal DNA". In: *Frontiers in Ecology and the Environment* 8.5 (June 2010), pp. 233–238. (Visited on 08/23/2019).

[12] Tristan Cordier et al. "Predicting the Ecological Quality Status of Marine Environments from eDNA Metabarcoding Data Using Supervised Machine Learning". In: *Environmental Science & Technology* 51.16 (Aug. 15, 2017), pp. 9118–9126.

[13] Tristan Cordier et al. “Supervised machine learning outperforms taxonomy-based environmental DNA metabarcoding applied to biomonitoring”. In: *Molecular Ecology Resources* 18.6 (Nov. 2018), pp. 1381–1391.

[14] P. S. Cranston. “Biomonitoring and invertebrate taxonomy”. In: *Environmental Monitoring and Assessment* 14.2 (May 1, 1990), pp. 265–273.

[15] *DNA Sequencing Costs: Data*. Genome.gov. URL: https://www.genome.gov/about-genomics/fact-sheets/DNA-Sequencing-Costs-Data (visited on 08/20/2019).

[16] Robert Edgar. *Abundance and amplification bias in amplicon sequencing*. URL: https://drive5.com/usearch/manual/amplification_bias.html (visited on 08/14/2019).

[17] Robert C. Edgar. “UNBIAS: An attempt to correct abundance bias in 16S sequencing, with limited success”. In: *bioRxiv* (Apr. 4, 2017), p. 124149. URL: https://www.biorxiv.org/content/10.1101/124149v1.

[18] *Emschergenossenschaft (EG) und Lippeverband (LV)*. URL: https://www.eglv.de (visited on 08/20/2019).

[19] *Environmental Impact Assessment*. URL: https://ec.europa.eu/environment/eia/index_en.htm (visited on 08/20/2019).

[20] Joel N. Franklin. *Matrix Theory*. Courier Corporation, July 31, 2012. 319 pp.

[21] Hugh G. Gauch. “Noise Reduction By Eigenvector Ordinations”. In: *Ecology* 63.6 (1982), pp. 1643–1649.

[22] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. 2nd ed. Springer Series in Statistics. New York: Springer-Verlag, 2009. ISBN: 978-0-387-84857-0. URL: https://www.springer.com/gp/book/9780387848570 (visited on 07/05/2019).

[23] Hebert Paul D. N. et al. “Biological identifications through DNA barcodes”. In: *Proceedings of the Royal Society of London. Series B: Biological Sciences* 270.1512 (Feb. 7, 2003), pp. 313–321.

[24] Syrie M. Hermans et al. “Bacteria as emerging indicators of soil condition”. In: *Applied and Environmental Microbiology* (Oct. 2016), AEM.02826–16.

[25] Henrik Krehenwinkel et al. “Estimating and mitigating amplification bias in qualitative and quantitative arthropod metabarcoding”. In: *Scientific Reports* 7.1 (Dec. 15, 2017), pp. 1–12.

[26] J. B. Kruskal. “Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis”. In: *Psychometrika* 29.1 (Mar. 1964), pp. 1–27.

[27] J. B. Kruskal. “Nonmetric multidimensional scaling: A numerical method”. In: *Psychometrika* 29.2 (June 1964), pp. 115–129.

[28] Kenneth Lange. “Markov Chain Monte Carlo”. In: *Numerical Analysis for Statisticians*. Statistics and Computing. New York, NY: Springer New York, 2010, pp. 527–550. ISBN: 978-1-4419-5945-4.

[29] Franck Lejzerowicz et al. “High-throughput sequencing and morphology perform equally well for benthic monitoring of marine ecosystems”. In: *Scientific Reports* 5 (Sept. 10, 2015), p. 13932. (Visited on 05/29/2019).

[30] Li Li, Binghui Zheng, and Lusan Liu. "Biomonitoring and Bioindicators Used for River Ecosystems: Definitions, Approaches and Trends". In: *Procedia Environmental Sciences* 2 (2010), pp. 1510–1524.

[31] Paul J. McMurdie and Susan Holmes. "Waste Not, Want Not: Why Rarefying Microbiome Data Is Inadmissible". In: *PLoS Computational Biology* 10.4 (Apr. 3, 2014). Ed. by Alice Carolyn McHardy, e1003531.

[32] Paul J. McMurdie and Susan Holmes. "Waste Not, Want Not: Why Rarefying Microbiome Data Is Inadmissible". In: *PLOS Computational Biology* 10.4 (Apr. 2014), pp. 1–12.

[33] R. E. Miles. "The Complete Amalgamation into Blocks, by Weighted Means, of a Finite Set of Real Numbers". In: *Biometrika* 46.3 (1959), pp. 317–327. URL: https://www.jstor.org/stable/2333529 (visited on 08/25/2019).

[34] "Natural Bacterial Communities Serve as Quantitative Geochemical Biosensors". In: 6 (July 1, 2015).

[35] Radford Neal. "MCMC Using Hamiltonian Dynamics". In: *Handbook of Markov Chain Monte Carlo*. Vol. 20116022. Chapman and Hall/CRC, May 10, 2011.

[36] Jari Oksanen. "Multivariate Analysis of Ecological Communities in R: vegan tutorial". In: (May 10, 2015), p. 43.

[37] Jari Oksanen. *vegan package | R Documentation*. URL: https://www.rdocumentation.org/packages/vegan/versions/2.4-2 (visited on 08/26/2019).

[38] Emanuel Paradis. *ape package | R Documentation*. Analyses of Phylogenetics and Evolution. URL: https://www.rdocumentation.org/packages/ape/versions/5.3 (visited on 08/30/2019).

[39] Joseph N. Paulson et al. "Robust methods for differential abundance analysis in marker gene surveys". In: *Nature methods* 10.12 (), pp. 1200–1202.

[40] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[41] Mariana Buongermino Pereira et al. "Comparison of normalization methods for the analysis of metagenomic gene abundance data". In: *BMC Genomics* 19 (Apr. 20, 2018).

[42] Mark D. Robinson, Davis J. McCarthy, and Gordon K. Smyth. "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data". In: *Bioinformatics* 26.1 (Jan. 1, 2010), pp. 139–140. URL: https://academic.oup.com/bioinformatics/article/26/1/139/182458.

[43] Les Ruse. "Classification of nutrient impact on lakes using the chironomid pupal exuvial technique". In: *Ecological Indicators - ECOL INDIC* 10 (May 1, 2010), pp. 594–601.

[44] John Salvatier, Thomas V. Wiecki, and Christopher Fonnesbeck. "Probabilistic programming in Python using PyMC3". In: *PeerJ Computer Science* 2 (Apr. 2016), e55. URL: https://doi.org/10.7717/peerj-cs.55.

[45] Savolainen Vincent et al. "Towards writing the encyclopaedia of life: an introduction to DNA barcoding". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 360.1462 (Oct. 29, 2005), pp. 1805–1811.

[46] Jay Shendure and Hanlee Ji. “Next-generation DNA sequencing”. In: *Nature Biotechnology* 26.10 (Oct. 2008), pp. 1135–1145.

[47] Thorsten Stoeck et al. “Environmental DNA metabarcoding of benthic bacterial communities indicates the benthic footprint of salmon aquaculture”. In: *Marine Pollution Bulletin* 127 (Feb. 2018), pp. 139–149.

[48] Pierre Taberlet et al. “Towards next-generation biodiversity assessment using DNA metabarcoding: NEXT-GENERATION DNA METABARCODING”. In: *Molecular Ecology* 21.8 (Apr. 2012), pp. 2045–2050.

[49] Warren S. Torgerson. “Multidimensional scaling: I. Theory and method”. In: *Psychometrika* 17.4 (Dec. 1, 1952).

[50] OA US EPA. *Our Mission and What We Do*. US EPA. URL: https://www.epa.gov/aboutepa/our-mission-and-what-we-do (visited on 08/20/2019).

[51] H. G. Washington. “Diversity, biotic and similarity indices: A review with special relevance to aquatic ecosystems”. In: *Water Research* 18.6 (Jan. 1, 1984), pp. 653–694. (Visited on 08/23/2019).

[52] Sophie Weiss et al. “Correlation detection strategies in microbial data sets vary widely in sensitivity and precision”. In: *The ISME Journal* 10.7 (July 2016), pp. 1669–1681.

[53] James Robert White, Niranjan Nagarajan, and Mihai Pop. “Statistical methods for detecting differentially abundant features in clinical metagenomic samples”. In: *PLoS computational biology* 5.4 (Apr. 2009), e1000352.

Appendix A

Appendix

Here we present additional results for different features sets than the ones presented in Chapter 4, so as to complete and complement the discussion without burdening the reader with long, difficult to read tables and figures.

The PCoA sets presented here were computed using the Bray-Curtis distance measure; their only difference with the ones presented in the Results & Discussion Chapter is that only some of the axes where used for classification. In particular, the first axes to describe 99% and 90% of the variance were used. Also, the configuration of a 20-dimensional NMDS ordination was used as a feature set. The procedure was run for 10000 steps and did not converge, so it does not represent the best 20-dimensional NMDS configuration. Finally the PCA projection of the data was used as features.

The results for maximum similarity are presented in tables A.1 for Logistic regression and A.2 for Random Forest. As with the other feature sets, Logistic regression is outperforming Random Forest (except for NMDS). Both methods are above the baseline. PCA has the lowest score when used in Logistic regression, and second lowest when used in Random Forest.

The results for maximum dissimilarity are presented in tables A.3 for Logistic regression and A.4 for Random Forest. Unlike the other feature sets in the same setting, Logistic regression is outperforming Random Forest (except for NMDS). Both methods are above the baseline. It is interesting to note that the sets PCoA 90% and PCoA CSS 99% produce a better accuracy for Logistic regression than its best set in Chapter 4.

The logistic regression model using the PCA set classified everything as white, and got a score of 87.20%. This might look like a good score, but its black water recall is zero, which makes it practically unusable. The case is similar in the Random Forest case.

We used the feature importance method of Random Forest to determine which taxonomic Orders contribute the most to the predictive power of the classifier. Figure A.2 shows the

Features used	Confusion Matrix Predicted Black	Predicted White	Accuracy
PCoA 99%	16 3	5 140	95.12%
PCoA 90%	16 3	5 140	95.12%
PCoA CSS 99%	16 1	5 142	96.34%
PCoA CSS 90%	16 2	5 141	95.73%
NMDS	13 8	8 135	90.24%
PCA	4 2	17 141	88.41%

Table A.1 Results from maximising similarity using Logistic Regression. The percentages indicate the total variance the axes of PCoA explain.

mean and sum aggregation for the OTU and OTU MIN CSS feature sets in the maximum similarity setting. Figure A.1 for the maximum dissimilarity.

Features used	Confusion Matrix Predicted Black	Predicted White	Accuracy
PCoA 99%	4 5	17 138	86.59%
PCoA 90%	9 4	12 139	90.24%
PCoA CSS 99%	10 0	11 143	93.29%
PCoA CSS 90%	12 0	9 143	94.51%
NMDS	10 2	11 141	92.07%
PCA	2 0	19 143	88.41%

Table A.2 Results from maximising similarity using Random Forest. The percentages indicate the total variance the axes of PCoA explain.

Features used	Confusion Matrix Predicted Black	Predicted White	Accuracy
PCoA 99%	7 8	14 135	86.59%%
PCoA 90%	7 5	14 138	88.41%
PCoA CSS 99%	8 5	13 138	89.02%
PCoA CSS 90%	8 11	13 132	85.37%
NMDS	3 10	18 133	82.93%
PCA	0 0	21 143	87.20%

Table A.3 Results from maximising dissimilarity using Logistic Regression. The percentages indicate the total variance the axes of PCoA explain.

Features used	Confusion Matrix Predicted Black	Predicted White	Accuracy
PCoA 99%	0 4	21 139	84.76%
PCoA 90%	0 4	21 139	84.76%
PCoA CSS 99%	0 4	21 139	84.76%
PCoA CSS 90%	0 5	21 138	84.15%
NMDS	1 2	20 141	86.59%
PCA	0 9	21 134	81.71%

Table A.4 Results from maximising dissimilarity using Random Forest. The percentages indicate the total variance the axes of PCoA explain.

Fig. A.1 Species' importance per taxonomic order as calculated by Random Forest in the maximum similarity test. Averaging the importance for the sets: OTU A.1a, and OTU MIN CSS A.1b. Summing the importance for the sets: OTU A.1c, and OTU MIN CSS A.1d.

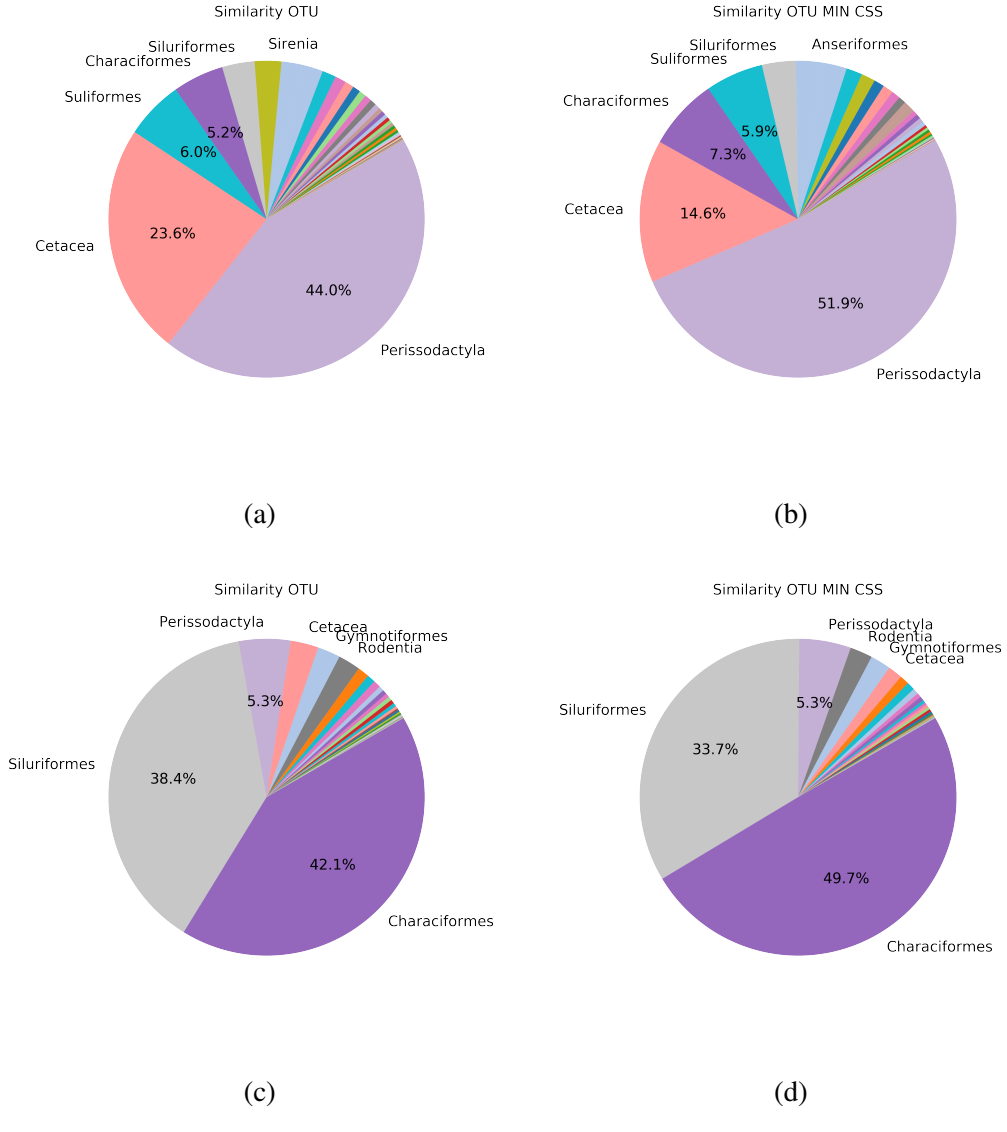


Fig. A.2 Species' importance per taxonomic order as calculated by Random Forest in the maximum similarity test. Averaging the importance for the sets: OTU A.2a, and OTU MIN CSS A.2b. Summing the importance for the sets: OTU A.2c, and OTU MIN CSS A.2d.