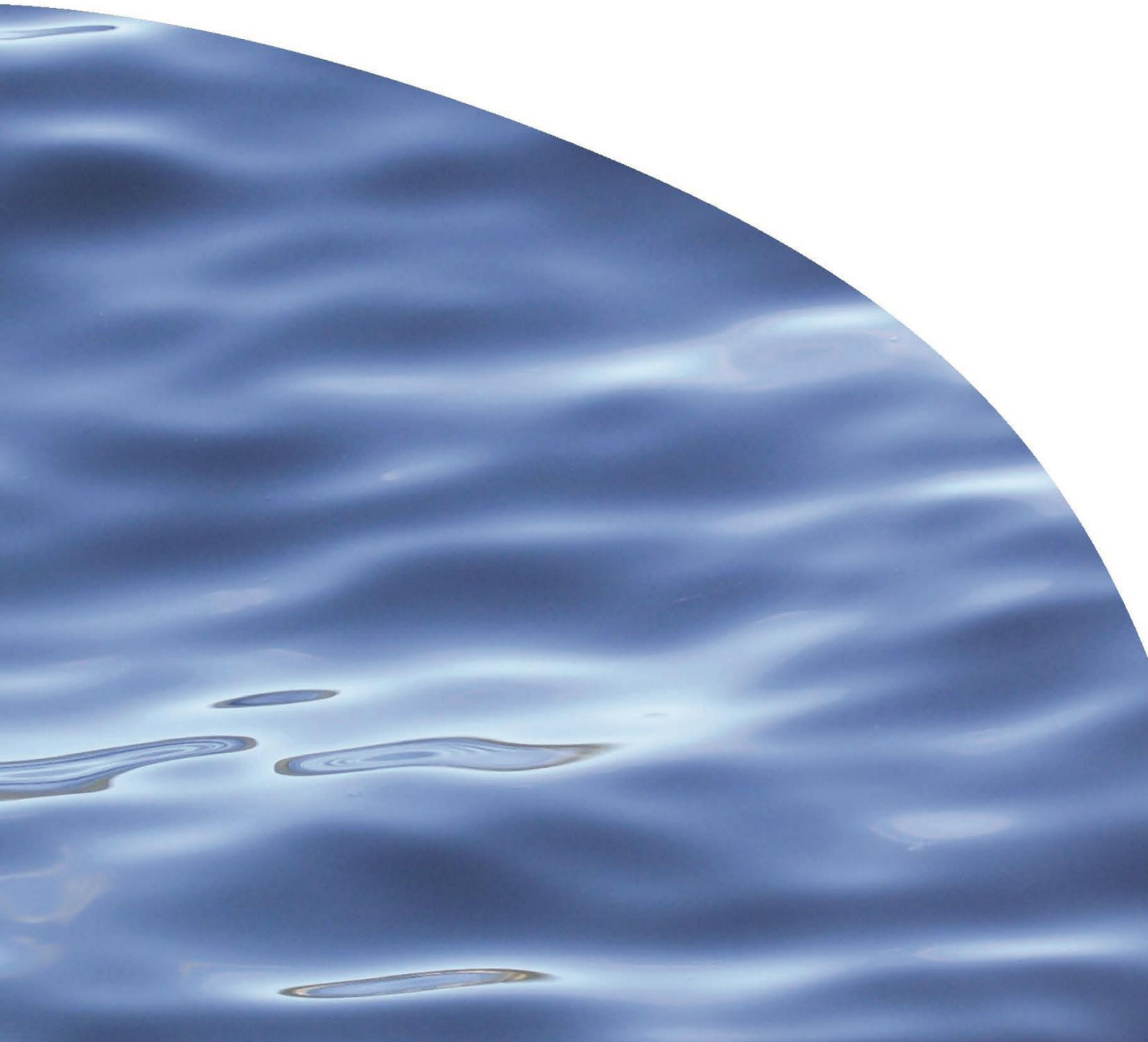
CAWTHRON
The power of science®

REPORT NO. 3573

MOLECULAR TOOLS FOR CHARACTERISING FRESHWATER FISH COMMUNITIES IN NEW ZEALAND



MOLECULAR TOOLS FOR CHARACTERISING FRESHWATER FISH COMMUNITIES IN NEW ZEALAND

JONATHAN BANKS, LAURA KELLY, JOANNE CLAPCOTT

Prepared for Ministry of Business, Innovation and Employment
Envirolink CAWX1802

CAWTHRON INSTITUTE
98 Halifax Street East, Nelson 7010 | Private Bag 2, Nelson 7042 | New Zealand
Ph. +64 3 548 2319 | Fax. +64 3 546 9464
www.cawthron.org.nz

REVIEWED BY:
Anastasija Zaiko

APPROVED FOR RELEASE BY:
Roger Young

ISSUE DATE: 13 November 2020

RECOMMENDED CITATION: Banks J, Kelly L, Clapcott J 2020. Molecular tools for characterising freshwater fish communities in New Zealand. Prepared for Ministry of Business, Innovation and the Environment, Envirolink CAWX1802. Cawthron Report No. 3573. 66 p. plus appendices.

EXECUTIVE SUMMARY

The objective of this research was to develop a standardised and cost-effective method for monitoring freshwater fish species in wadeable streams using environmental DNA (eDNA). Environmental DNA is DNA isolated from an environmental sample such as water or soil rather than directly from the organism(s) of interest. In this report we describe the development of a whole community assessment method (eDNA metabarcoding). Steps included i) development of a fish eDNA sample collection protocol, ii) selection of a universal primer pair for high throughput sequencing of fish communities, iii) sequencing of missing genes to build a fish reference database, iv) development of a bioinformatics pipeline to analyse community sequences and v) field validation of the eDNA methods alongside standard fishing methods.

Development of a fish eDNA sample collection protocol

We found that a purpose-built backpack pump eDNA sampler facilitated the filtering of larger volumes of water (> 1 L) through 5µm self-preserving filters compared with other methods tested (e.g. bench top, lab-based vacuum pumps) and filter sizes and filter types and maximised fish eDNA detection. However, sample replication at each site is important to account for the patchy distribution of eDNA and maximise the probability of detecting the species present. We recommend that, when using the backpack sampler, the volume of water filtered is maximised (≥ 3 L per filter) using a minimum of three replicate filters for characterising fish communities in wadeable streams; greater filtration volumes will be required if target DNA concentrations are low or the probability of detection required is high. Further research is required to determine the number of replicate filters required to maximise the probability of detection across different environments (e.g. lakes, braided rivers).

Selection of a universal primer pair for high throughput sequencing of fish communities, and sequencing of missing genes to build a fish database

We downloaded sequences for three mitochondrial genes (cytochrome C oxidase subunit 1, cytochrome b, and 12S rRNA), and the d-loop of the mitochondrial control region. These regions are commonly used to characterise fish species because they vary between species and have conserved regions among species that allow the design of PCR primers to amplify all species. After an in silico assessment of available sequences for the four gene regions from as many NZ freshwater fish species as possible, we selected a region of the 12S rRNA gene that contained sequences of a suitable length to identify the fish species present within the limitations of instruments currently available. We extracted fish DNA from morphologically identified specimens to add to the published sequences to build a reference DNA database.

We conducted field trials of two primer pairs ‘MiFish’ and ‘teleo’ that each target individual portions of the fish 12S rRNA gene because previous studies found that in silico and in vitro results (i.e. simulated and field trials) were not always in agreement in their comparison of gene regions and primer sets. Results of field trials showed differences in the discrimination of species between primer pairs. In general, the MiFish primer set performed better at species level, discriminating most taxonomic groups, although it performed poorly for

taxonomic assignments within bullies; in contrast, the teleo primers distinguished some of the bully species. Ideally, both primers would be used to characterise fish communities in New Zealand. However, as the reference database coverage increases, we suggest that MiFish is a useful primer pair set for eDNA metabarcoding of New Zealand fish communities when used with Illumina DNA sequencing. The identification of the optimal primer pair for metabarcoding all New Zealand freshwater fish requires more research and more than one primer pair may be required to achieve full species resolution.

Development of a bioinformatics pipeline to analyse community sequences

We developed an extensive bioinformatic pipeline to analyse sequences which included multiple data filtering steps. The bioinformatics pipeline was designed to target fish taxa with a high level of certainty, with a trade-off being a higher potential for false negatives (where a species is incorrectly defined as not being present). The level of certainty is adjustable, but we recommend it is kept consistent among repeated samplings to ensure results are comparable. Open source pipelines should be used for bioinformatic analysis and code is available to enable analyses to be updated if new pipelines or database sequences are added in future.

Field validation of the eDNA methods alongside standard fishing methods

Field trials at 13 sites demonstrated that the developed methods confidently assigned most of the eDNA-derived sequences to species detected using electric-fishing methods; at some sites additional taxa that were not physically captured were nevertheless detected using eDNA metabarcoding. However, some sequences could not be confidently assigned beyond genus level; possibly because all the intra-species sequence variation for the 12S rRNA gene has yet to be included in the reference database. As more sequences are obtained from more species and more geographical areas, the proportion of sequences that can be assigned to species will increase over time as the sequence database coverage is extended. Thus, the reference database must be viewed as a ‘living’ database until full geographical coverage of reference fish species are obtained.

In summary, each step of the workflow developed for the eDNA monitoring of freshwater fish can be considered as a series of modules (DNA capture, DNA extraction, gene region choice and amplification, data processing) that will almost certainly change as new technologies develop. However, before changes are made to the workflow, new methods should be validated by running them alongside the existing protocols to ensure the results are comparable. We recommend the development of a clear framework to guide the choice of eDNA analysis depending on the research question or required outcome of the analysis (e.g. species versus community composition) and to identify how to exchange modules as new developments emerge. For now, the protocols, and using the database and code provided in this report, is a ‘first step’ in the use of eDNA to characterise freshwater fish communities in New Zealand, with the caveat that currently some sequences can be assigned to genus but not to species. Given the changing nature of techniques and instrument capability in the eDNA area, and as experience is gained with the use of the tool and more research is conducted, our recommendations may be superseded.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

GLOSSARY

Amplicon is a short piece of an organism's genome that has been amplified to a measurable amount by PCR.

DNA is deoxyribonucleic acid. A long molecule made up of two chains of nucleotides coiled around each other in a helix. DNA carries the instructions for the development, functioning, growth, and reproduction of most organisms.

DNA polymerase is an enzyme used to copy DNA.

environmental DNA (eDNA) is DNA isolated from an environmental sample such as water or soil rather than from the organism itself.

High throughput sequencing (HTS) is sequencing a mixture of DNA so that each of the constituent DNA copies in the mixture can be read.

In silico is an expression meaning performed on computer or via computer simulation.

Mitochondrial genes are 37 genes that encode 13 proteins, 22 tRNAs, and 2 rRNAs**.**

Metabarcoding is using PCR to amplify a region of the genome to produce amplicons that will distinguish each taxon in a community.

Nucleotides are four chemical bases (adenine, cytosine, guanine, thymine) usually abbreviated as A, C, G, T) that make up each DNA strand.

Oligonucleotide is a fragment of DNA made up of a few nucleotides.

PCR is polymerase chain reaction, a method of amplifying a piece of an organism's genome to a measurable amount.

Primers are oligonucleotides that are complementary to a region of an organism's genome. Primers are used to initiate the DNA polymerase copying of DNA.

RNA is ribonucleic acid, a polymeric molecule essential in various biological roles in coding, decoding, regulation and expression of genes.

rRNA is ribosomal ribonucleic acid, a ribozyme which carries out protein synthesis in ribosomes.

tRNA is transfer ribonucleic acid, a type of RNA molecule that helps decode a messenger RNA (mRNA) sequence into a protein.

1. INTRODUCTION

1.1. Current fish community monitoring

Regional councils are required to identify freshwater objectives and set resource limits that maintain and improve freshwater values (National Policy Statement for Freshwater Management 2017). In addition, there are several region-specific mandates that require councils to monitor and report on freshwater fish (Resource Management Act, unitary plans, watershed plans, consents and cultural indicators). Nationally relevant freshwater values that are monitored include ecosystem health, fishing and mahinga kai, all of which are assessed in part by determining the presence, abundance, and health of freshwater fish communities.

Monitoring of freshwater fish communities is limited in many areas of New Zealand (Joy et al. 2013), primarily due to the costly and labour-intensive nature of such monitoring. Electric fishing is commonly used to survey freshwater communities but it has several limitations: it is ineffective in deeper water bodies such as lakes and non-wadeable streams, it can fail to detect transient or elusive species, it can damage in-stream habitat, and it can injure or kill fish. Electric fishing requires extensive training of personnel and poses a health and safety risk. Additionally, electric fishing can be influenced by environmental factors such as water conductivity, temperature and depth, stream size, substrate type, macrophyte growth, conductive surfaces, rain, fish species, fish size and fish behaviour. Monitoring methods for deeper water such as netting can be lethal to fish and other animals such as waterfowl. As such, there is a need for more widely applicable non-invasive sampling methods to monitor fish communities and environmental DNA is one potential methodology to fill this need.

1.2. Environmental DNA (eDNA) monitoring

The term environmental DNA (eDNA) was first used to describe a DNA-based method of characterising microbial communities from sediments (Ogram et al. 1987). The term has expanded to describe any DNA extracted from environmental samples such as soil and water without isolating target organisms from the sample. The source of the monitored DNA includes faeces, skin cells and mucus shed naturally by the target organisms (Taberlet et al. 2012).

Environmental DNA is particularly attractive to fish biologists as the technique addresses many of the limitations of current monitoring methods. Environmental DNA is increasingly used internationally to determine fish biodiversity, and the distribution of endangered and pest fish. Regions such as North America (e.g. Olds et al. 2016), Europe (e.g. Valentini et al. 2016), and Australia (e.g. Renshaw et al. 2015; Hinlo et al. 2017) have put considerable effort into using eDNA to monitor freshwater fish communities.

The benefits of monitoring fish communities with eDNA compared with current monitoring methods include:

1. Reduced field time associated with eDNA sampling, compared with existing methods. Reduced per sample collection time allows more samples to be collected and thus greater spatial and temporal coverage of fish communities that will contribute to better policy, resource management consent decisions, restoration efforts, threatened species monitoring, and fish distribution models.
2. Because eDNA can be transported a considerable distance, estimates of fish diversity (and potentially relative abundance) include communities from stream reaches upstream of the collection point. Integrated estimates are more relevant to 'State of the Environment' reporting because the spatial scale represented from a single eDNA sample is representative of upstream communities.
3. The need for specialised taxonomic knowledge is reduced as fish are identified from diagnostic DNA sequences rather than sometimes obscure morphological characteristics. Additionally, juveniles of some fish species will be able to be identified more easily.

There is considerable interest from New Zealand local and national government agencies in developing and standardising environmental DNA-based methods to monitor New Zealand freshwater fish communities. Methods of characterising communities from eDNA used internationally have differed in: the volumes of water collected, isolating the DNA from the environmental sample, extracting the eDNA, amplifying the genetic information that identifies individual species, the gene region used to characterise individual species, and the bioinformatics (the computer code) used to filter out poor quality sequences, identify sequence artefacts and assign taxonomic identity to DNA sequences.

1.3. Overview of environmental DNA workflow

Analysing environmental DNA requires a series of steps (Figure 1). The first step is capturing bulk DNA (which may contain DNA of the target species) from the environment. The second step is extracting and purifying the DNA so that it can be analysed, and the third step is detecting (or not) the presence of the target species, or the characterisation of the target communities. These steps are unlikely to change over time; however, the ways in which each step will be done will almost certainly change as new instruments and software are developed. It may be helpful to think of analysing eDNA as a modular process where the details of each module can change, e.g. the method used to capture the eDNA may vary, but there will almost certainly always be a need to capture the DNA from environmental matrix.

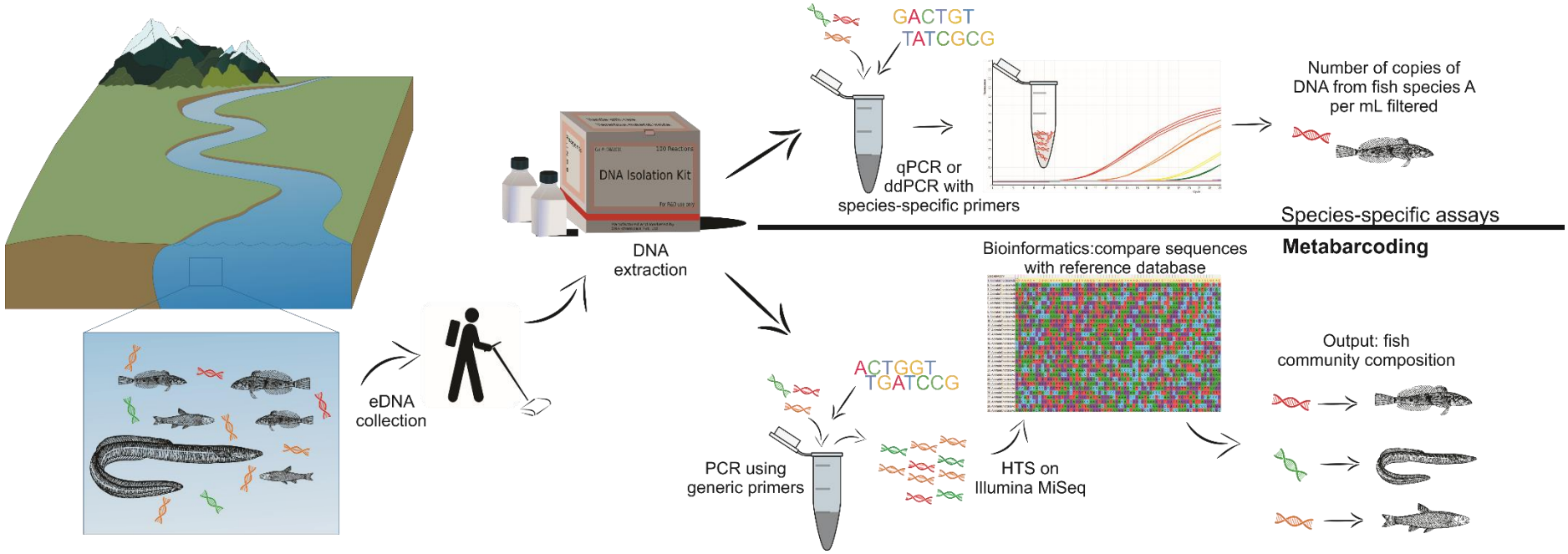


Figure 1. The general process for eDNA workflows from eDNA collection to outputs. The process begins by collecting eDNA from the environment (e.g. by filtering) to concentrate the DNA from the water column. The next step is to extract the DNA from the filters (usually this is done using commercially available extraction kits). At this point, the workflow splits and the DNA can be processed in different ways: (i) assays that target specific species and typically yield either presence/absence or (semi)quantitative results, or (ii) metabarcoding assays that use wider range primers to target a community of organisms (in this case, fish) and yield presence/absence results and relative abundance of DNA sequences. Once the DNA is extracted from samples, it can be used for either approach; the choice of method will depend on the question. The DNA extracts can also be archived and re-used with either method or new methods if needed in future.

1.3.1. DNA collection

Most DNA isolation methods can be categorised as either precipitation-based or filter-based (Ficetola et al. 2008; Turner et al. 2015; Lear et al. 2018). Precipitation-based methods generally precipitate the DNA from a smaller volume of water than the volumes used by filtration-based methods. Typically, the DNA from between 45 mL (e.g. Ficetola et al. 2008) and 100 mL of water (e.g. Muha et al. 2019) is isolated and precipitated using ethanol and sodium acetate followed by centrifugation. Filtration-based methods pass water through filters with pores ranging from 0.22 µm to 10 µm (Turner et al. 2014; Goldberg et al. 2016). Precipitation has been shown to give lower concentrations of DNA from the same water samples compared with filtration (Deiner et al. 2015).

1.3.2. DNA extraction

Once the DNA has been collected on the filters or precipitated, the DNA must be resolubilised, and contaminants such as clay particles, humic acid, and proteins removed to reduce the extent of polymerase chain reaction inhibition these can cause. This process is known as DNA extraction, and there are many methods to extract and purify the DNA. Lear et al. (2018) conducted an extensive review of the eDNA literature and found that Qiagen 'DNeasy Blood & Tissue' kits were the most widely used method to extract DNA from fresh and marine waters.

1.3.3. DNA-based species detection and community characterisation

Once the DNA has been purified, the next step is to attempt to detect the target species or characterise the community of interest. There are two widely used approaches to detect species from eDNA: species-specific detection or community characterisation. The two approaches both use polymerase chain reactions to amplify the DNA but differ in the primers used and the method used to detect the species of interest.

Polymerase chain reaction (PCR) is a method of amplifying the DNA of target species up to detectable amounts. The PCR process copies the DNA present in a sample using exogenously supplied buffers, nucleotides, synthetic DNA primers (a short piece of DNA approximately 24 nucleotides long), and a DNA polymerase enzyme that produces up to 2^{40} copies (approximately 10^{12} copies) of each target sequence in a sample (Figure 2). The reaction process involves approximately 40 cycles of heating the double stranded target DNA to 94 °C to separate the double-stranded DNA into single-stranded DNA, then cooling the reaction to a temperature around 60 °C to allow the primers to bind to the target DNA, and then heating the reaction to 72 °C for the DNA polymerase to copy the region downstream of the primer binding sequence.

The key to directing the PCR is the choice of primers, as the DNA polymerase enzyme cannot start amplifying DNA without a short piece of DNA first binding to the

target DNA. If there is no target the primers cannot bind, and the target species is not detected. Synthetic primers are complementary to short, unique sequences of the genome that are present in all members of the target group, and primers can be designed to amplify the taxonomic level of interest, for example a single species or multiple species. There are further guidelines such as matching the predicted annealing temperatures of the two primers and having the guanine and cytosine (two of the four nucleotides present in DNA) content between 40 to 60%. One very important restriction is keeping the amplicon length (i.e. the number of nucleotides in the sequence amplified) compatible with the capabilities of the instrument being used to detect the DNA. For example, the read length of the Illumina high throughput sequencer is restricted to a maximum of approximately 550 nucleotides (Illumina 2020).

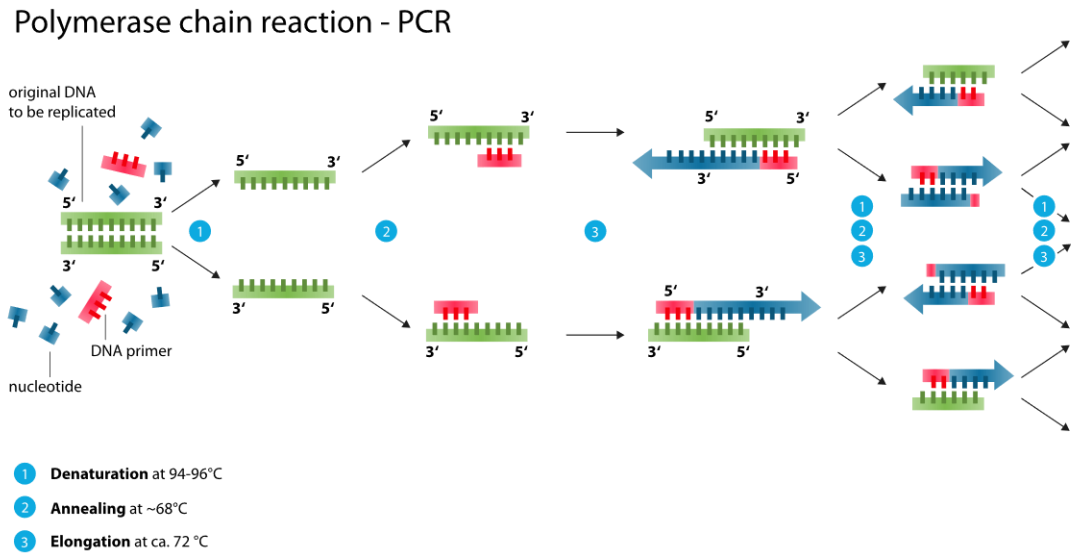


Figure 2. Schematic of the polymerase chain reaction (PCR) process. (https://commons.wikimedia.org/wiki/File:Polymerase_chain_reaction.svg).

Species-specific detection

To detect specific species in an environmental sample, primers are synthesised that bind only to the target species' DNA. Production of PCR products infers that the species has been detected. Amplification can be measured by a range of methods including electrophoresis, the use of fluorescent hydrolysis probes (quantitative PCR), or droplet digital PCR (Quan et al. 2018). Detections can be confirmed by further processing of the PCR and reading the DNA sequence. Sanger sequencing reads the nucleotides in the sequence; the sequence is then matched to sequences that have been generated from morphologically identified specimens that have previously been sequenced for that gene region. An example of a species-specific survey from eDNA

samples was the use of primers specific for brown trout *Salmo trutta,* to monitor the progress of trout removal from Karori Reservoir (Banks et al. 2016).

Community characterisation using metabarcoding

Metabarcoding is characterising a community from a small portion of the genetic material from the environment without the need for isolation and laboratory cultivation of individual species. Metabarcoding characterisation of fish communities requires a different approach to species-specific testing. Primers are designed that will initiate DNA amplification of all species in the target group. The primers chosen flank regions that are unique to each species so that when the sequence of each polymerase chain reaction product is read, the sequence can be matched back to a sequence generated from tissue collected from a physically identified voucher specimen.

Metabarcoding involves an initial PCR using community-specific primers (e.g. New Zealand freshwater fish). The key to high throughput sequences (HTS) is separating each of the DNA strands produced from the initial PCR. In the case of Illumina DNA sequencing this is achieved by adding an oligonucleotide (a short piece of DNA) of known sequence that is complementary to an oligonucleotide bound to a flow cell. Once bound to the flow cell, the pieces of DNA are copied to produce clusters of copies. DNA polymerase, and single nucleotides (adenine, cytosine, guanine, or thymine: A, G, C, T) are washed over the flow cell. Each of the four nucleotides has a unique fluorescent reporter molecule that emits light at a different wavelength when excited by a laser. The nucleotides also have a reversible terminator molecule that prevents extension of the DNA strand by the DNA polymerase enzyme. Unbound nucleotides are washed away, the nucleotides are excited by the laser, and a photo is taken to identify which nucleotide has bound to each of the clusters. The terminator molecule is then removed, and the flow cell is washed with nucleotides again, extending the complementary DNA fragment by a single nucleotide again. This cycle is repeated hundreds of times to read the sequence of each of the clusters.

1.3.4. Bioinformatics

The data obtained from the high throughput sequencer are then converted to text sequences known as FastQ files that contain the sequence and quality scores for each of the base calls. Quality scores assess the accuracy of the base calls made by the software; sequences that do not meet the quality score requirements are removed from the data set. Other processing of the data includes removing the primer sequences from the beginning and ends of the sequence reads, characterising the error profiles and de-noising the sequences (i.e. algorithmic error correction), removal of chimeras and merging of the forward and reverse reads. Once the sequences have been trimmed and assessed for quality control, each sequence is given a taxonomic assignment using an algorithm that matches the sequence to a reference database of sequences constructed using morphologically identified specimens. Sources of voucher sequences include GenBank (https://www.ncbi.nlm.nih.gov/genbank/) and the Barcode of Life databases (https://www.boldsystems.org/).

1.3.5. Key considerations in design and interpretation of eDNA metabarcoding studies

There are caveats that must be considered when using eDNA metabarcoding for the characterisation of freshwater fish communities. The caveats relate to issues with detection probabilities that arise from the different parts of the eDNA metabarcoding workflow, from eDNA sampling to laboratory workflows and data analysis.

The first component that needs to be considered is what Barnes and Turner (2016) described as the ecology of eDNA. This comprises the origin, state, transport, and fate of eDNA, which will impact the eDNA sampling design. The relevance of each of these to detecting freshwater fish in New Zealand is outlined here.

The 'origin' includes understanding how different fish species shed DNA into the environment. For example, diurnal and seasonal changes in fish activity related to their life histories, and species-specific rates of eDNA shedding can impact the likelihood of detection (Barnes & Turner 2016; Coulter et al. 2019). More DNA may be shed when species are spawning or otherwise more active in the water body (Klymus et al. 2015). Several studies have found that species-specific allometric scaling models perform better than using biomass alone as a predictor of eDNA concentrations (Maruyama et al. 2014).

'State' is related to origin but expands to include other environmental parameters. For example, eDNA can persist as free-floating molecules in the water column, or it could be bound to particulates or retained within mucilaginous material. The rate of degradation of these different forms of DNA can influence its persistence and, thus, the detectability of a species.

A consideration for the sampling design of eDNA studies is the 'transport' of eDNA downstream, thus a spot sample taken at a site is actually an integrated assessment of the community upstream of the sampling point (Li et al. 2018). The degree of transportation and the persistence of the eDNA will influence the distance upstream that is captured in a spot sample. Studies using caged trout in otherwise fishless streams showed that eDNA was present 240 m downstream of the fish regardless of flow rates, that high flow rates resulted in lower DNA capture and that the presence of PCR inhibitors (e.g. humic and tannic acids) (Rossen et al. 1992; Rådström et al. 2004) could result in non-detection of the eDNA (Jane et al. 2015).

'Fate' is described as the potential for degradation via chemicals, radiation, or enzymatic activity, or through mechanical fragmentation of the DNA. The influence of environmental variables such as pH, temperature, turbidity, and dissolved oxygen concentrations on the persistence of eDNA have been investigated in both laboratory and field studies (Stoeckle et al. 2017; Seymour et al. 2018). Although some consistent findings have been established, the effects of all parameters are difficult to quantify and model as these effects are generally site-specific (Poté et al. 2009;

Barnes & Turner 2016). Thus, it is important to keep these considerations in mind when interpreting results from eDNA metabarcoding from lotic waterbodies.

The second component that needs to be considered are the laboratory protocols for eDNA isolation, extraction, and processing. It is imperative that the laboratory workflow has controls in place to minimise the risk of contamination, i.e. a sterile working environment. Laboratory workflow also encompass choice of primers to minimise systematic primer bias, the use of PCR replicates to reduce the impact of stochastic primer bias, and the inclusion of sufficient negative controls to enable the detection of contamination. For example, the United States Fish and Wildlife Service has a comprehensive quality plan for species-specific eDNA tests that covers the stages of an eDNA study from the pre-collection planning to the interpretation of results (US Fish and Wildlife Service Midwest Region 2019)

Similarly, a range of aspects in the bioinformatic analysis of the data can impact the reported results. The choice of parameters and stringency during the filtering stages of the analysis can also impact the results and there is often a trade-off between maximising sequence retention and ensuring high-quality sequences (Callahan et al. 2016). The reference database that is used to assign species identifications to sequences is critically important. The Barcode of Life Database (BOLD) has a series of requirements for the recognition of a formal barcode: a species name, voucher data (including the catalogue number and institution storing the specimen), collection record (including the collector, date collected and location with GPS coordinates), the identifier of the specimen, the sequence (for BOLD this is COI, but the same principles apply for other gene regions including 12S), the PCR primers used to generate the sequence and the raw trace files from sequencing (Ratnasingham & Herbert 2007). The database must be as complete as possible and appropriately curated to reduce false positive and false negative results. For example, omitting outgroups (i.e. non-fish taxa) from the reference database means that sequences belonging to other taxonomic groups such as mammals are classified as unassigned fish sequences. Similarly, spurious assignments can occur at genus and species levels when the reference database is not complete (Lecaudey et al. 2019; Schenekar et al. 2020).

1.4. Scope of this project

The key to introducing a new protocol for monitoring New Zealand fish communities is identifying a method that, when compared with the standardised netting and electric fishing methods, is as effective, can be applied to a wide range of waterways, and is more economical. This document provides information on the development of a protocol that uses environmental DNA to monitor fish communities in wadable streams. The recommended guidelines are intended to produce a method to characterise freshwater fish communities from eDNA.

In developing a series of protocols, the following aspects are addressed:

- an in silico analysis of the sequences available for New Zealand freshwater fish to identify (as far as possible from the available sequences) a region of the fish genomes that corresponds to morphologically recognised fish species
- protocols that cover both presence/ absence and relative abundance of freshwater fish species
- protocols are provided for the sampling and extraction of eDNA
- protocols are provided for the generation of DNA amplicon libraries by polymerase chain reactions (PCR)
- protocols are provided for the purification and normalisation of concentrations of the amplicon libraries before sequencing the amplicon libraries on a high throughput DNA sequencing instrument
- we provide a bioinformatic pipeline (computer code) to filter and analyse the DNA sequences obtained from the high throughput DNA sequencer with minimal specialist knowledge.

The environmental DNA collection protocols were tested in a range of environments; however, some aspects of eDNA collection (e.g. the effects of season) were outside the scope of this study and require additional investigation.

It is expected that aspects of the protocols and pipelines will be refined as additional experience is gained in eDNA metabarcoding for fish community monitoring in New Zealand. The reference database should be viewed as a ‘living’ document that will be updated as additional sequencing is undertaken to capture the genetic diversity within New Zealand freshwater fish species.

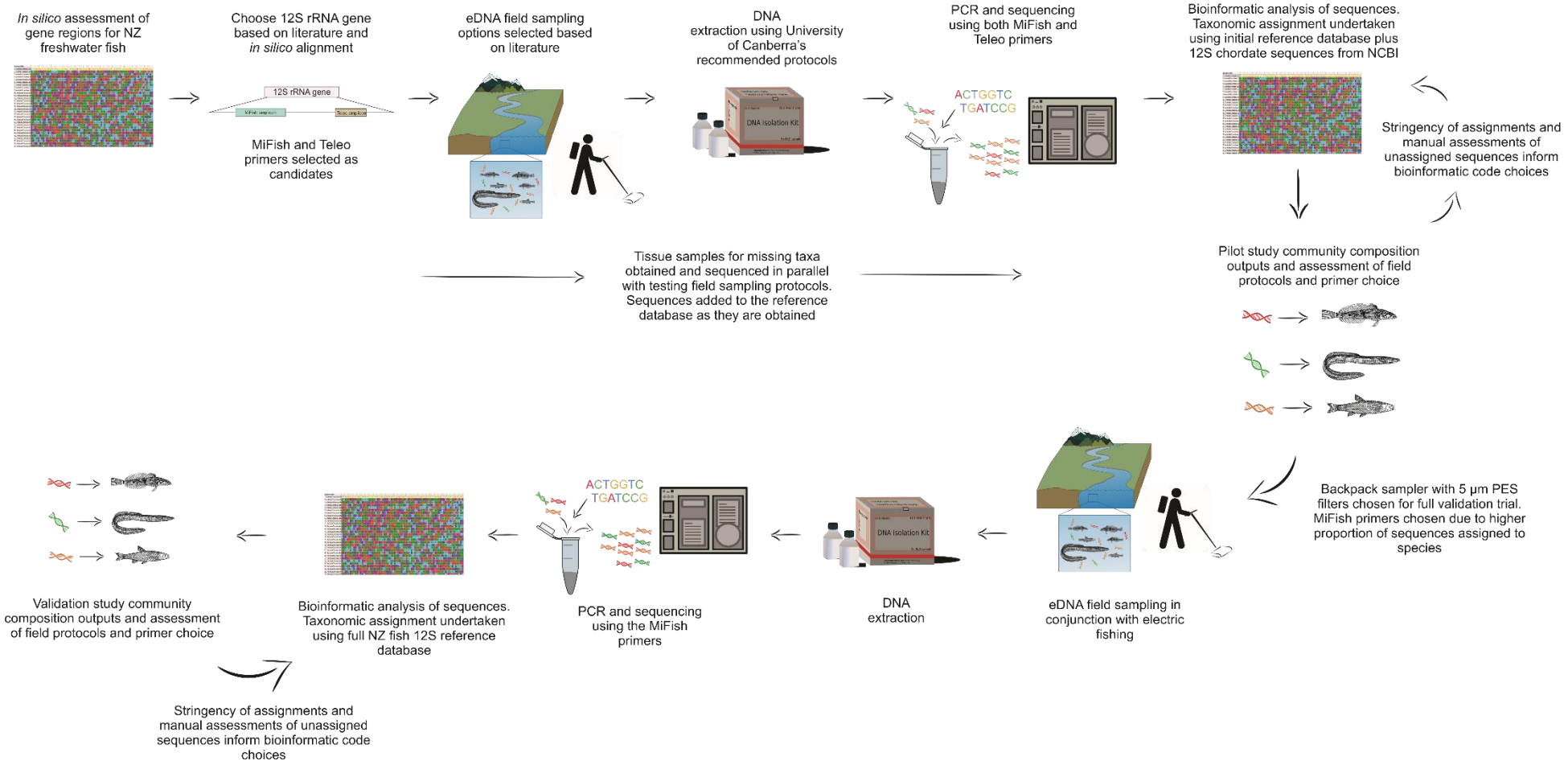


Figure 3. A conceptual overview of the workflow process throughout the project. Note that there are feedbacks among the results obtained from the bioinformatic process that enabled setting choices for the pipelines to be assessed and refined based on the available data. Reference database sequences were added when these were available.

2. ENVIRONMENTAL DNA SAMPLING AND EXTRACTION

The protocols outlined below were developed using data from a pilot study in which 3 filter types, 2 filtration systems, and 2 sample collection protocols were compared. Water samples were collected from 6 sites in the Tasman and Waikato regions that had a variety of water chemistry characteristics and were selected based on prior knowledge of diverse fish communities. The tests of replicate samples and sample volumes were conducted in a subsequent field trial with paired electric fishing.

2.1. Sample collection

Successfully characterising a fish community is dependent on the concentration of the DNA in the sample and the sensitivity of the method of detecting that DNA. The distribution of eDNA appears to be heterogeneous both in space and in time as its distribution is affected by a number of factors such as water flow, behaviour of the target species, and seasonality (Furlan et al. 2015). These variables are further compounded when characterising communities as these factors are likely to affect each species within the community differently, at different times.

The degree to which inter species variation is compensated for during eDNA collection will be dependent on the importance of minimising false negatives (species present but not detected) and false positives (species absent but detected). Based on a literature review (Lear et al. 2018), and previous experience, precipitation methods were not considered as an option for isolating DNA from water samples as they are limited to small water samples and typically have poorer DNA yields than other eDNA isolation methods (Peixoto et al. 2020). Thus, we chose to focus on filtration methods and filters that allowed us to filter a larger volume of water as logic suggested that the detection probability would increase with an increase in volume of water filtered.

To isolate eDNA via filtration we first evaluated three filter types (PES filters with 1.2 or 5 µm pores, Smith Root, Vancouver, WA, USA) and glass fibre microfilters grade F (GF) 0.7 µm pores, Cytiva, UK) and two pump systems (Geopump™ Peristaltic Pump Series II, Geotech Environmental Equipment, In, Denver, CO, USA, and the eDNA back pack sampler, Smith-Root, Vancouver, WA, USA). These systems were chosen as a literature review found they have been used successfully by other researchers for monitoring fish communities.

2.1.1. Geotech pump

The Geotech pump could draw water through filters in the field using battery power but the time required to filter 1 L samples was extensive, i.e. greater than 60 min. There was also the potential for contamination with handling filters in the field. Thus, water was collected and returned to the laboratory for filtering.

Triplicate 1 L samples of water were collected in bottles that had previously been soaked in 10% sodium hypochlorite solution for at least 10 minutes and then rinsed with UV-treated, reverse osmosis filtered water. Samples were filtered through either a 1.5 µm pore glass fibre filter or a 1.2 µm pore polyethersulfone (PES) filter paper using the Geotech peristaltic pump within 12 h of collecting the sample. Each sample was filtered until flow stopped, the filter removed, and a new filter installed; this was continued until the entire 1 L was filtered. Filter papers were stored at -20 °C in 5 mL gamma sterilised, screw cap polypropylene tubes (Techno Plas, Australia, catalogue number P5016SL). Initial trials revealed the GF filters were prone to ripping and difficult to extract DNA from, and they were abandoned prior to further trials.

During subsequent field trials, triplicate 1 L samples were collected at the bottom, middle and top of a 20 m reach at each stream site. Samples were collected from the downstream end of the reach first, then the middle and then upper parts to minimise the impacts of sampling on subsequent samples.

2.1.2. Smith-Root eDNA sampler—previously ANDe

The Smith-Root eDNA backpack sampler (https://www.smith-root.com/edna/edna-sampler) is a purpose-built eDNA filtration system method which is used to draw stream water across a filter on site (Figure 3). We used pre-packed 1.2 µm or 5 µm pore (47 mm diameter) polyethersulfone (PES) filters provided in a single-use proprietary housing and tubing from Smith Root (Thomas et al. 2018). We filtered up to 10 L of stream water at a minimum rate of 1 L/min with a maximum negative pressure of 12 PSI. Filtration was stopped once the flow rate could not be maintained at 1 L/min with the maximum pressure. Once filtration was finished, the intake hose was inverted, and air drawn through the filter to dry the filter. The filters were then removed from the housings using the single-use forceps supplied with each filter, placed in the 5 mL screw cap container described in Section 2.1.1, and stored at -20 °C. Self-preserving filters are also available (and were used in later field trials), whereby there is no need to remove the filter from the housing (minimising risk of contamination) and samples can be stored at room temperature prior to analysis (Thomas et al. 2019).

Figure 4. Smith Root eDNA sampler backpack (patent pending) with boom retracted.

The Smith-Root sampler could efficiently filter water (Table 1) using 12V battery power. Two sealed lead-acid batteries (9ah) were sufficient for a full day of sampling. The extendable boom allowing a range of in-stream habitats to be accessed. At field trials, three replicate samples were collected from the lower, middle, and upper points of a 20-m reach at each stream site. Samples were collected from the bottom end of the reach first, then the middle and then the top to minimise the impacts of sampling on subsequent samples. Additionally, a transect sample was collected whereby the reach was traversed and water was continuously pumped across two replicate filters from the range of hydraulic habitats present (Figure 4).

Table 1. Time required to filter 1, 2, 3 and 5 L of water (four replicates of each volume) through 5 µm PES filters using the Smith Root sampler backpack.

Mean volume filtered (L)	Mean time required for filtration (min)
1.0	3.6
2.0	4.5
3.0	6.2
5.0	8.8

Figure 5. Smith Root sampler backpack collecting a) replicate sample from the bankside, and b) a transect sample along the stream reach.

2.1.3. Filter and pump comparison

We tested various sample collection protocols at six stream sites – four in the Nelson area and two in Waikato (Table 2). The field testing in March 2019 included the following comparisons:

- Sample collection methods: a) contemporary spot water sample collection and laboratory filtering, b) bankside spot water collection using Smith Root sampler, c) transect water collection using Smith Root sampler
- Filter sizes: a) 1.2 µm PES filter, b) 5 µm PES filter, c) 1.2 µm GF
- Range of environmental conditions: six wadeable stream sites with various water quality and expected fish diversity.

At each site, a 50-m reach was established across a run-riffle-pool sequence. Within the sample reach, five methods of surface water sample collection were applied as described below for a total of 15 replicate samples. Additionally, two field controls were collected per site, whereby DNA-free water was transferred to and from the site in a water bottle or processed through 1.2 µm PES filter using the ANDe system.

- Three replicate 1 L samples collected following the protocol of the University of Canberra (Jack Rojahn, University of Canberra)
- Three replicate 1-L samples following the protocols of Thomas et al. (2018), using a 1.2 µm PES filter with a set flow rate of 1 L/min and a maximum pump pressure of 12PSI
- Three replicate 1-L samples following protocols of Thomas et al. (2018), using a 1.2 µm GF filter with a set flow rate of 1 L/min and a maximum pump pressure of 12PSI
- Three replicate 3-L samples following the protocols of Thomas et al. (2018), using a 5 µm PES filter with a set flow rate of 1 L/min and a maximum pump pressure of 12PSI
- Two replicate 1-L surface water samples following the protocols of Thomas et al. (2018), collected simultaneously during a transect of the sample reach and using a 1.2 µm PES filter with a set flow rate of 0.5 L/min and a maximum pump pressure of 12PSI.

Table 2. Initial field trial sites.

Site name	Region	Turbidity (NTU)	Specific conductivity (µS cm^{-1})	Dominant aquatic plants	Width/Depth (m)
Brook Stream	Nelson	0.38	114.8	Filamentous	2.0/0.15
Maitai River	Nelson	0.42	155.4	Diatoms	7.0/0.25
Lud Valley Stream	Nelson	0.61	141.9	Filamentous, mats	1.0/0.30
Orphanage Stream	Nelson	0.86	243.3	Diatoms, mats	3.0/0.45
Mystery Creek	Waikato		252.0	*Egeria densa*	1.85/0.42
Komakorau Stream	Waikato		240.0	none	2.2/0.32

The initial field trial to investigate the use of different filter types and filtering systems revealed differences in the number of species detected among the sampling protocols, although these differences were not consistent among all sample types (Figure 5). As the PES 1.2 µm and 5 µm pore sizes exhibited similar performance based on the number of species detected, the 5 µm size was chosen for the remaining trials as it enabled a larger volume of water to be filtered across the membrane. Also, it was a lot easier to use the Smith-Root filtration system than to process samples in the laboratory using a benchtop filter station.

The preliminary transect sampling took more than twice as long as point sampling to collect in the field because it required samplers to enter the stream and wade safely. As sample collection took more time and there were no clear advantages in the number of taxa detected, point samples were chosen as the collection method for future sample collection.

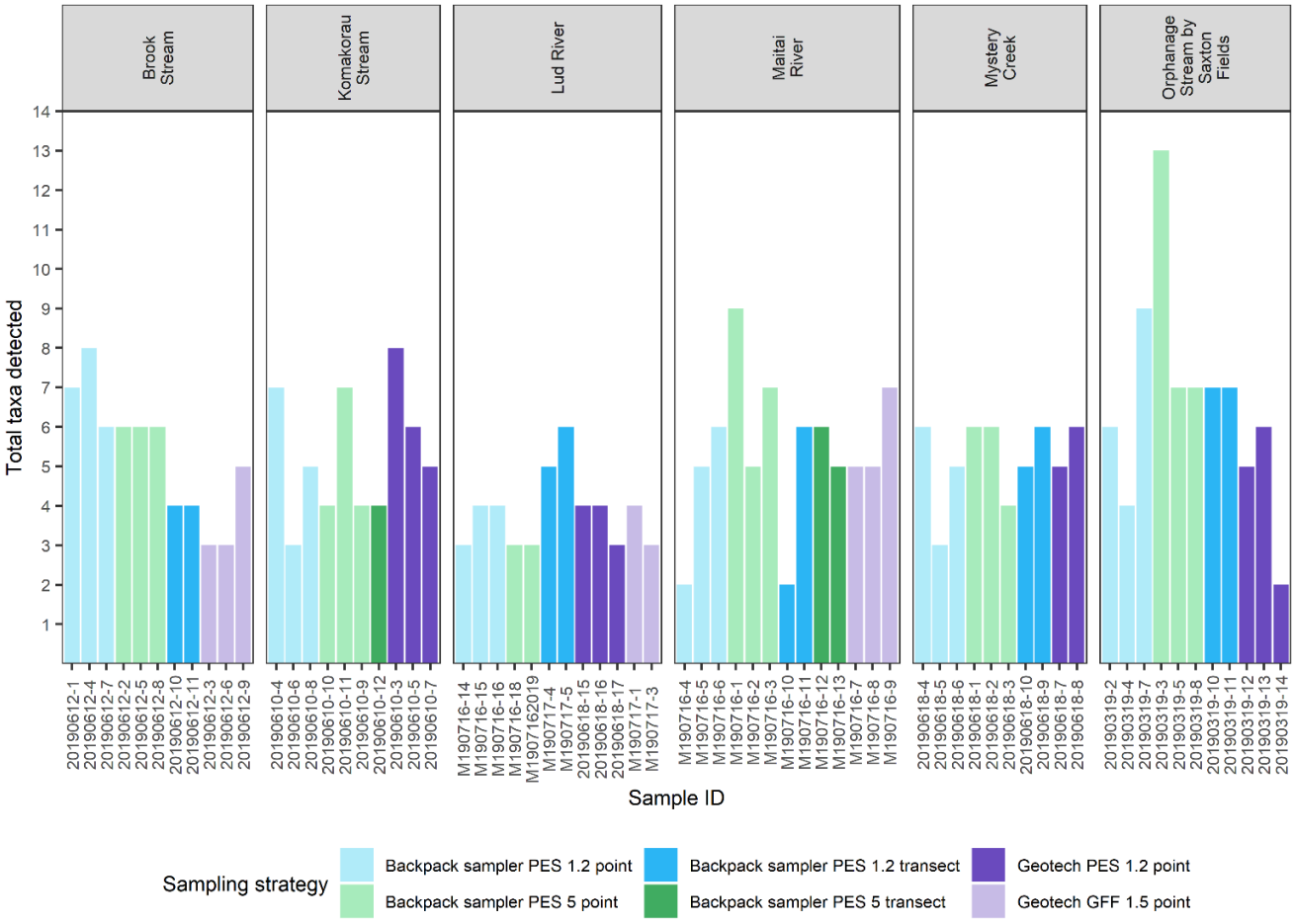


Figure 6. The number of taxa (species or genus where sequences could not be assigned to species) detected from the sequences using different field sampling methodologies and the teleo primer set.

2.1.4. Testing different water volumes

Large volumes of water can be filtered across the 1.2 µm and 5 µm pore filters with the Smith-Root sampler backpack, however, this volume is dependent on fine suspended sediment and other particulate matter in the water column, and the time available for filtering. To understand how different volumes impacted community detection, a trial was undertaken to filter four different volumes of water across the filter: 1, 2, 3 or 5 litres using the backpack sampler. The experiment was conducted in February 2020 at a single site (Poorman Valley Stream) in conjunction with single-pass electric fishing over a 150-m reach (Joy et al. 2013) to enable comparison of the communities detected. Samples for eDNA were collected first at the bottom of the 150-m electric fishing reach to minimise disturbance of the fish prior to sample collection. Electric fishing was undertaken immediately following the eDNA sample collection starting at the upstream end of the reach. Five replicates were taken of each water volume onto non-self-preserving 5-µm pore size filters, which were

transferred using sterile, single-use forceps into sterile tubes in the field and placed on ice immediately. Samples were transferred to the laboratory within 3 h and stored at -20 °C until the DNA was extracted.

The number of species detected by eDNA methods at the Poorman Valley Stream site in smaller volumes of water (1 L and 2 L) varied across the five replicates, and the maximum number of species detected was six (Figure 6). By contrast, six or seven species were detected consistently across all five replicates from the 3-L and 5-L volumes. No particular taxa were undetected at lower volumes; rather it appeared non-detection of taxa occurred randomly.

The volume trial, based on results from this single site, also revealed useful information regarding sample replication. Results suggest that to detect 100% of species present based on electric fishing records, more than three 3-L or 5-L samples are required (but see also Appendix 9). To detect 85% (6 out of 7) species present, three or more samples are required of any volume. More research from a range of sites and for a suite of fish communities is required to identify the confidence intervals around species' detection probabilities.

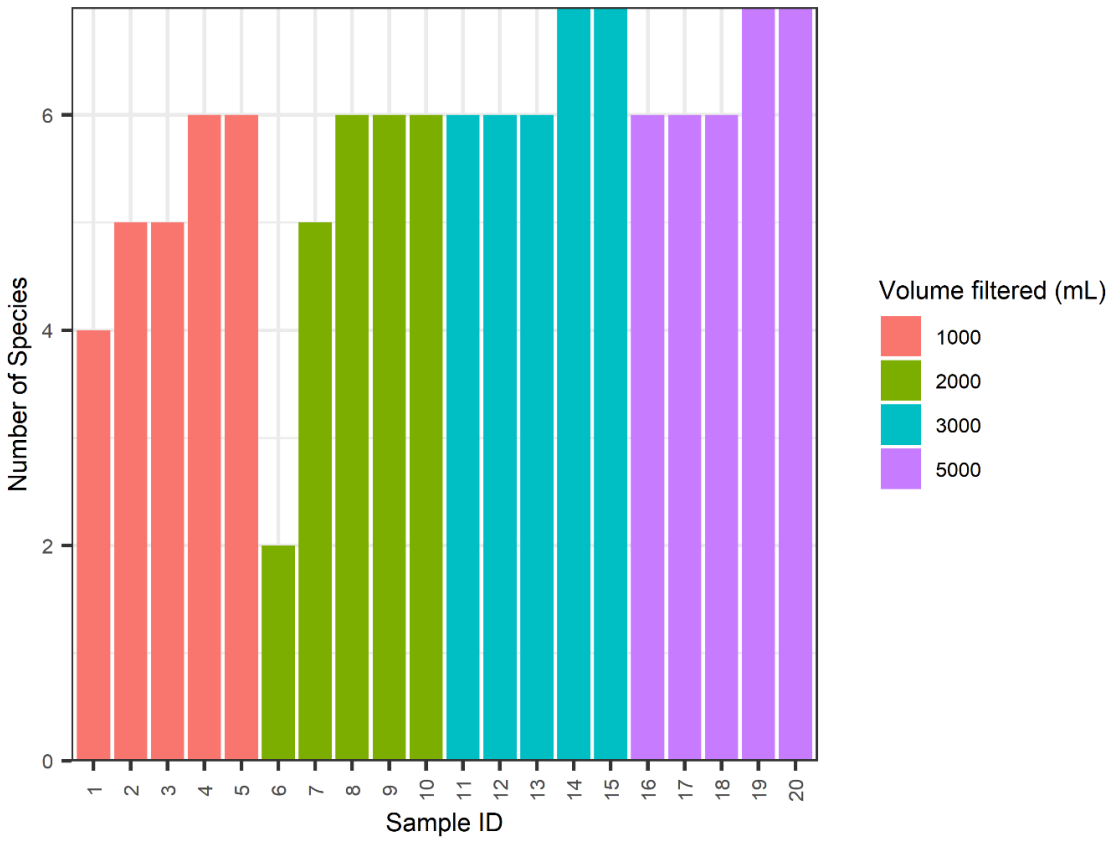


Figure 7. Number of species detected using eDNA from replicate samples for five volumes of water filtered at the Poorman Valley Stream site.

Standard operating procedures for the collection of fish eDNA are provided in Appendix 1.

2.2. eDNA extraction

There are many DNA extraction methods and in a recent review Lear et al. (2018) found that Qiagen DNeasy Blood & Tissue kits were more commonly used than DNeasy PowerSoil kits to extract DNA from filters for fish studies. Eichmiller et al. (2016) found that DNeasy Blood & Tissue kit obtained higher total DNA concentrations from filters than the PowerSoil kit (a method that uses beads to release the DNA) but that the PowerSoil kit was less likely to extract PCR inhibitors. Our experience with the PowerSoil kit is that the DNA can be damaged, and thus not amplifiable, by excessive time and/or speed during the bead beating step.

A modified protocol to extract DNA from filters using DNeasy Blood & Tissue kits has been shown to yield significantly more copies of DNA than three other methods (Renshaw et al. 2015). The modified protocol has also been used successfully by the EcoDNA lab, University of Canberra, (Jack Rojahn, University of Canberra, pers. comm.), one of the most experienced eDNA labs in Australia. Thus, we used DNeasy Blood & Tissue kits using the University of Canberra protocol detailed in Appendix 2 to extract DNA from filters.

Filters from all three filtration methods were removed from their 5-mL containers and cut into quarters using single-use scalpel blades and forceps that had been sterilised in 10% bleach. DNA was extracted from the filters using the method outlined in Appendix 2.

3. ENVIRONMENTAL DNA DETECTION

Next generation sequencing for metabarcoding requires the amplification of a target region of DNA using two short pieces of DNA called primers and a process called polymerase chain reaction (PCR). The two primers bind to complementary sequences and flank the target gene or region of DNA that is to be amplified. Gene targets and primer choices in eDNA analyses are dependent on the target organism/s. In general, primers should target regions of DNA that are conserved across all the organisms of interest (e.g. they retain the same or a very similar sequence so that the primers are able to attach). However, the region between the primers should include enough variation in the DNA sequence that the different species can be distinguished—they need to have a unique 'barcode'—hence the term metabarcoding.

Various gene targets have been used for fish eDNA studies. For example, mitochondrial genes cytochrome b (*cytb*), cytochrome c oxidase subunit 1 (COI), and the mitochondrial d-loop region have all been used in analyses of fish communities. More recently, the mitochondrial 12S rRNA gene has been used widely, with some studies also making use of the 16S rRNA gene (Shaw et al. 2016). The secondary structure of the 12S rRNA gene makes it amenable to use in metabarcoding studies as the secondary structure results in highly conserved stem regions that flank loops in which the DNA sequences are not as constrained, thus there are conserved primer binding regions flanking highly variable sections that enable sequences to be assigned to a fish taxon.

We chose to focus on mitochondrial gene regions, rather than nuclear gene regions as mitochondrial genes are more numerous in individual cells compared with nuclear genes (Rees et al. 2015), and their circular format increases their persistence in the environment (Turner et al. 2014). Thus, the probability of detecting rare species is increased by targeting regions of the mitochondrial genome, a feature which is particularly important given that the mitochondrial DNA from one species can make up as little as 10^{-8} percent of the total eDNA in freshwater (Turner et al. 2014).

3.1. In-silico analysis of available sequences to identify a suitable region of the genome to assign species identities

We searched GenBank (Sayers et al. 2019) for each species of freshwater fish listed in Dunn et al. (2018) for sequences from four gene regions and for complete mitochondria sequences to conduct an in silico assessment of potential primer sites and regions that unequivocally correspond to morpho-species. Four mitochondrial gene regions (12s rRNA, cytochrome b, cytochrome C oxidase subunit 1 (COI), and the d-loop of the mitochondrial control region) as well as complete mitochondrial genomes were downloaded from GenBank, aligned in Geneious 9.1.8 (https://www.geneious.com) and examined for: the number of species for which

sequences were available, variation in the gene regions among species, the availability of invariant sites of suitable length for primers that would amplify all species of New Zealand freshwater fish, the suitability of the length of the variable region flanked by the primers chosen for high throughput sequencing, and compatibility of potential regions with international studies.

We were unable to find a gene region with a complete list of sequences for all the 85 freshwater fish species in New Zealand. We found sequences for 12S for 42 species, cytochrome b—60 species, COI—42 species, control region—51 species (Table 4). Despite the lower number of sequences for the 12S rRNA, an examination of the sequences found that 12S had regions with invariant sites across all taxa that were sufficiently long (approximately 20 nucleotides) for primers to bind to. Additionally, a review of the literature found several publications using the 12S rRNA gene for fish community characterisation (Miya et al. 2015; Valentini et al. 2016) and published primers that were complementary with New Zealand fish species, i.e. MiFish and teleo (Figure 8). Our use of the 12S region has also been supported by an extensive comparison of the 12S, 16S rRNA gene, cytb and COI gene regions that found primers targeting 12S detected more fish diversity than 16S or COI, and considerably more diversity than primers targeting COI (Zhang et al. 2020)

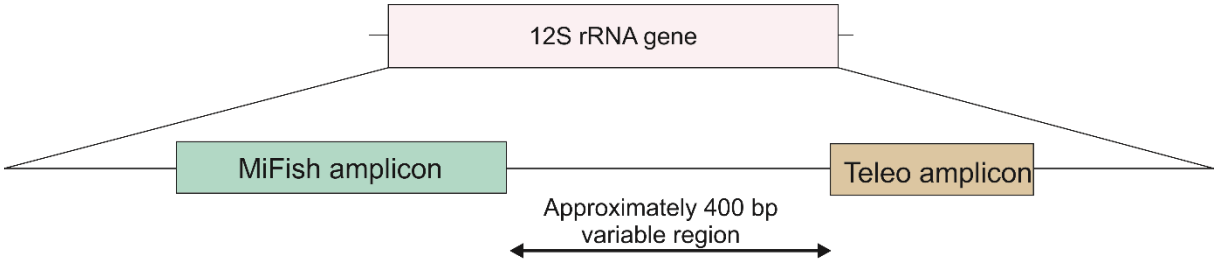


Figure 8. A schematic layout of the 12S rRNA gene regions with the locations of the MiFish and teleo amplicons. The *in silico* assessment of the two primer pairs consisted of aligning all available DNA sequences from different NZ fish taxa and determining the presence of informative variations among sequences.

To check that sequences for each species would differ sufficiently to allow detection, we built phylogenies for the taxa for which there were sequences using the Neighbor Joining algorithm (Saitou & Nei 1987). The availability of suitable primer sites that were invariant among species, and flanked variable regions, was assessed by visually examining the aligned sequences. To aid in visualising the pairwise differences, we produced a phylogeny for the freshwater fish using the Neighbor Joining method (Saitou & Nei 1987) for both gene regions after two rounds of sequencing DNA from specimens collected by other researchers (Figure 9, Figure 10, Figure 11, Figure 12). A summary of taxa that did not differ significantly based on pairwise differences between primer pair gene regions is given in Table 3.

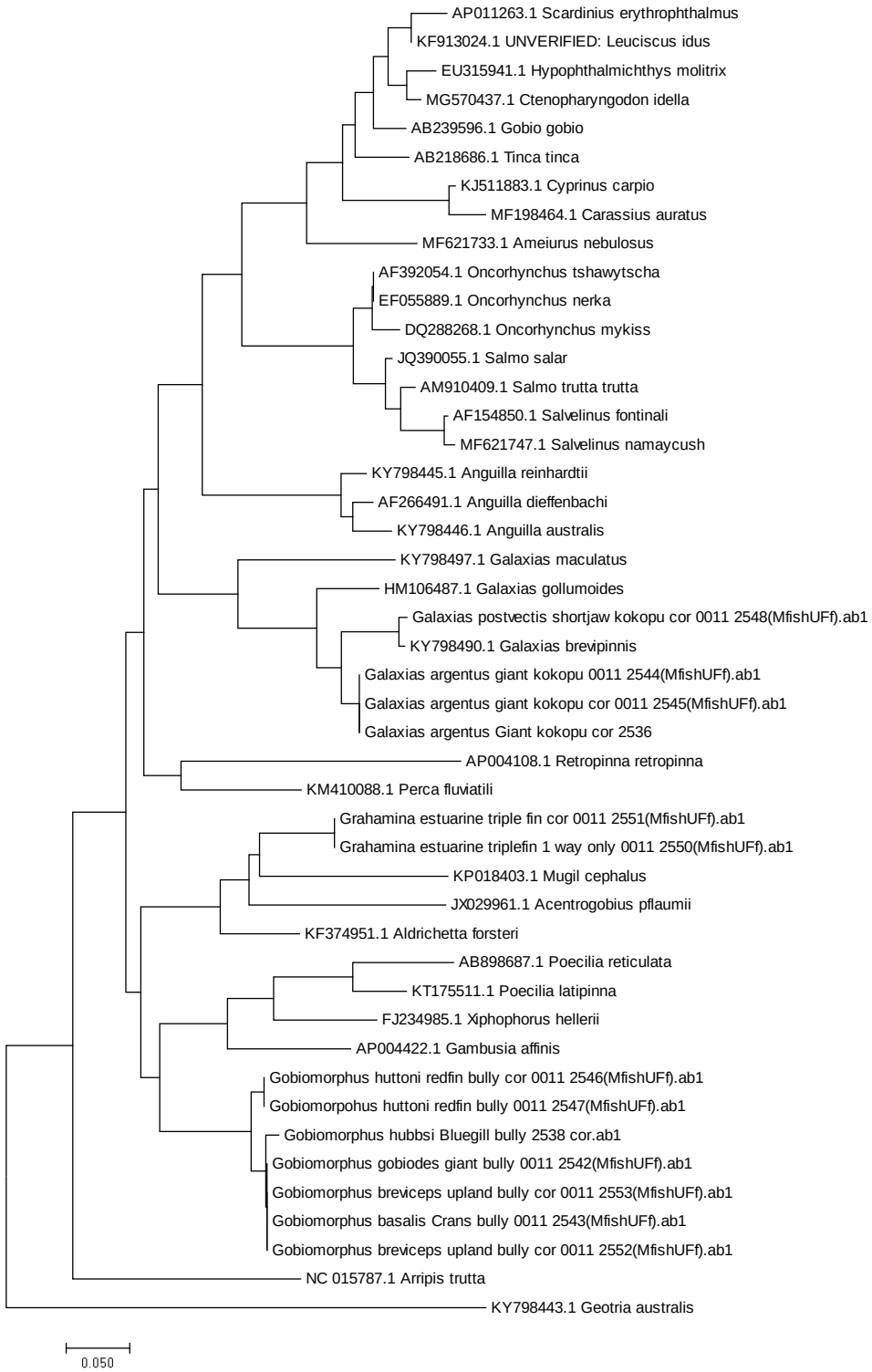


Figure 9. Phylogeny produced using the Neighbor Joining method (Saitou & Nei 1987) to visualise pairwise sequence differences between New Zealand freshwater fish species for the region of the 12S rRNA gene amplified by the MiFish primers (approximately 230 nucleotides) after the first round of additional sequences were added to the database. The horizontal length of each branch is proportional to the number of differences between taxa. Note the lack of differences between the upland (*Gobiomorphus breviceps*), Cran's

(*G.* basalis) and giant (*G. gobioides*) bullies.

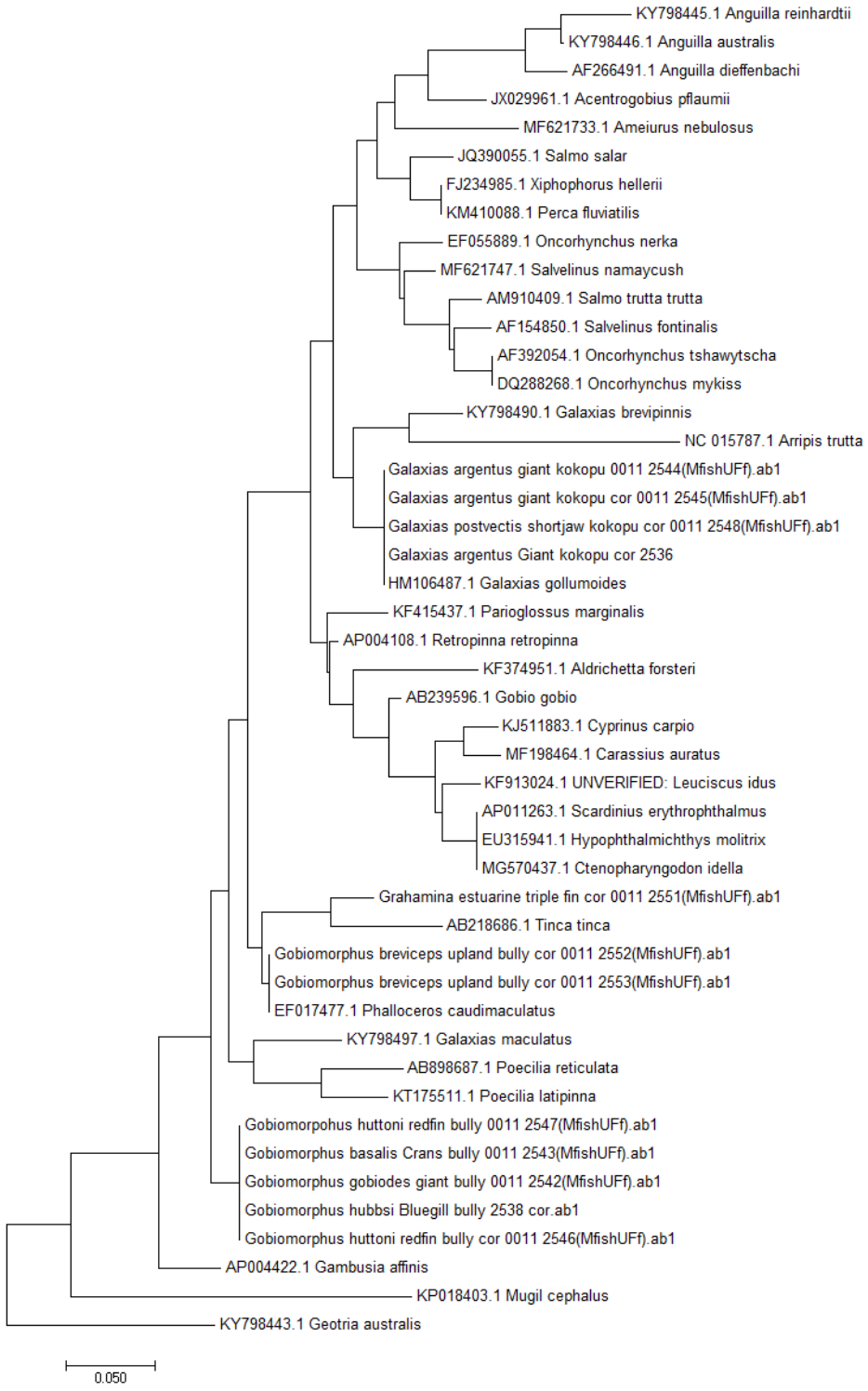


Figure 10. Phylogeny produced using the Neighbor Joining method (Saitou & Nei 1987) to visualise pairwise sequence differences between New Zealand freshwater fish species for the region of the 12S rRNA gene amplified by the Teleo primers (approximately 80 nucleotides) after the first round of additional sequences were added to the database. The horizontal length of each branch is proportional to the number of differences between taxa. Note the lack of differences for several clades and particularly between the redfin

(*Gobiomorphus huttoni*), bluegill (*G. hubbsi*), Cran's (*G.* basalis) and giant (*G. gobioides*) bullies.

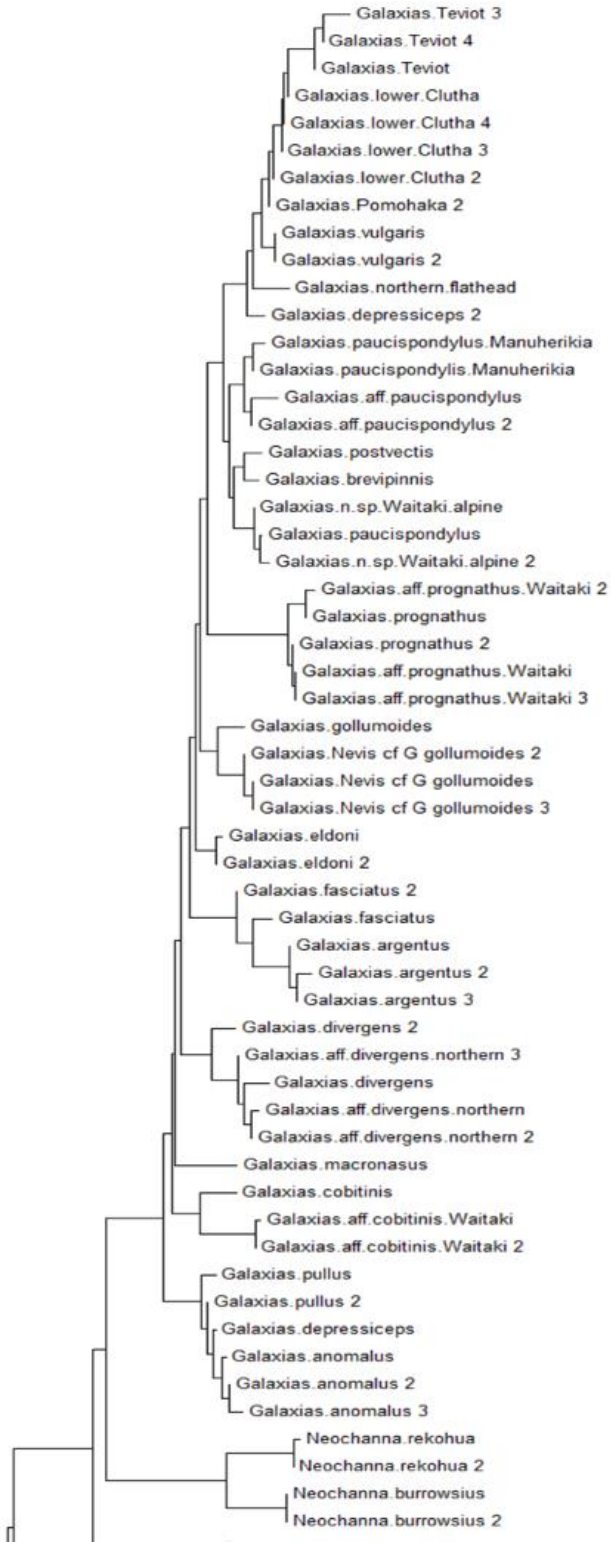


Figure 11. Phylogeny produced using the Neighbor Joining method (Saitou & Nei 1987) to visualise pairwise sequence differences between New Zealand freshwater fish species for the region of the 12S rRNA gene amplified by the MiFish primers (approximately 230 nucleotides) after a second round of sequencing of additional sequences were added to

the database. The horizontal length of each branch is proportional to the number of differences between taxa. Continued over page.

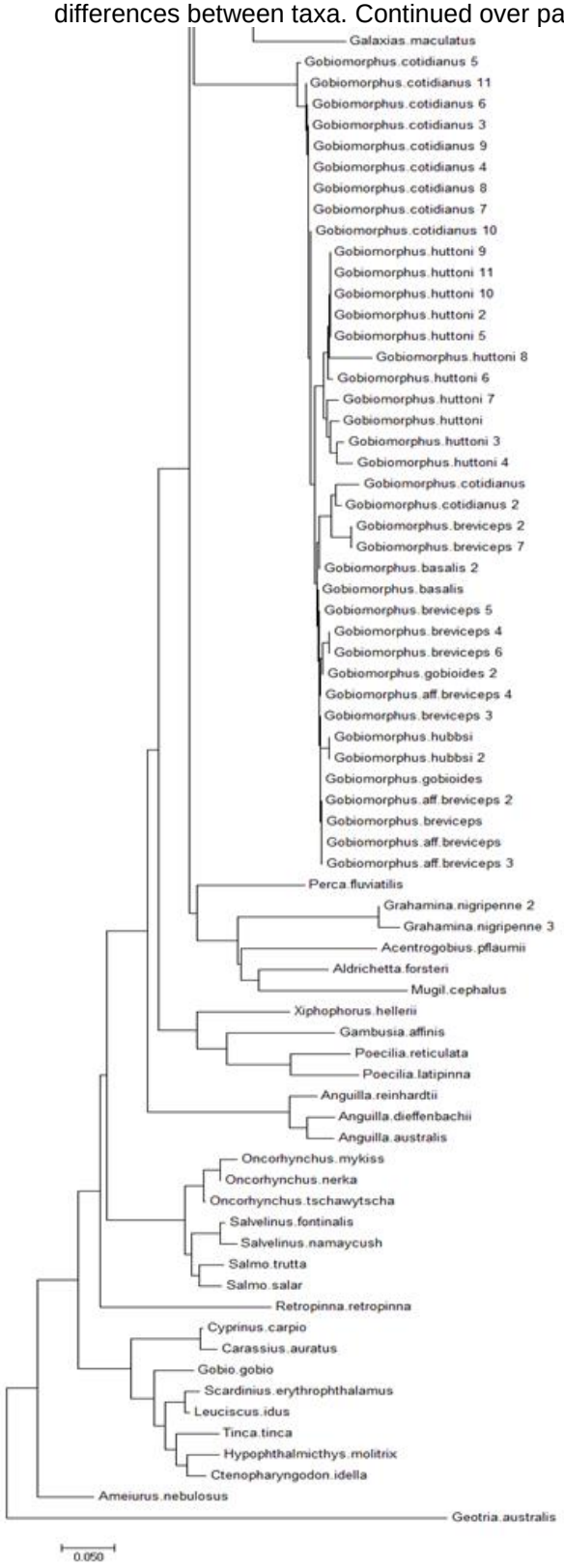


Figure 11 continued.

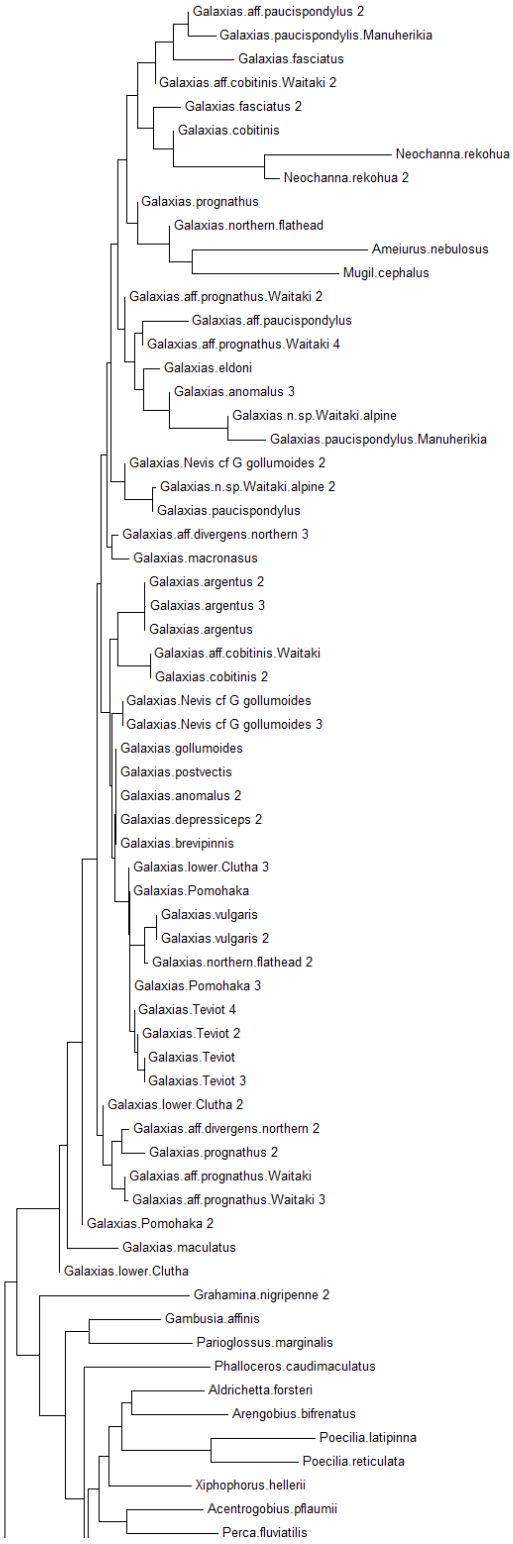


Figure 12. Phylogeny produced using the Neighbor Joining method (Saitou & Nei 1987) to visualise pairwise sequence differences between New Zealand freshwater fish species for the region of the 12S rRNA gene amplified by the Teleo primers (approximately 120 nucleotides after a second round of sequencing of additional sequences were added to the database. The horizontal length of each branch is proportional to the number of differences between taxa.

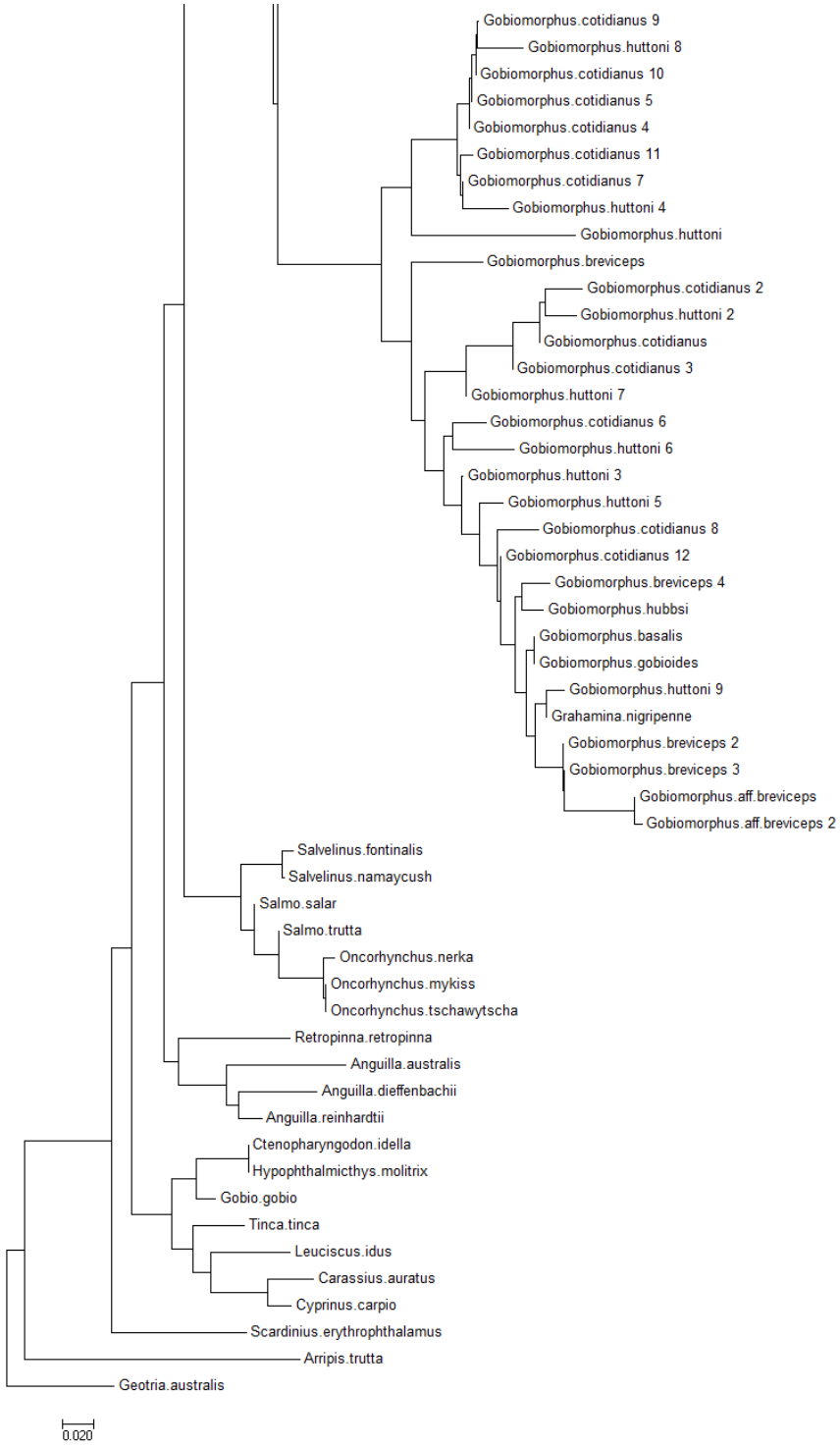


Figure 12 continued.

Table 3. Groups with zero pairwise differences in the portions of the 12S rRNA gene used in the database.

MiFish primer pair	Teleo primer pair
Galaxias "Pomohaka"	*Gobiomorphus basalis*
Galaxias "Lower Clutha"	*Gobiomorphus cotidianus*
	Gobiomorphus gobioides
Galaxias anomalus	
Galaxias depressiceps	*Galaxias cobitinis*
Galaxias pullus	*Galaxias* aff. *cobitinis* Waitaki
	Galaxias aff. *prognathus* Waitaki
Galaxias prognathus	
Galaxias aff. *prognathus* Waitaki	*Galaxias* aff. *paucispondylus*
	Galaxias paucispondylus Manuherikia
Gobiomorphus basalis	
Gobiomorphus breviceps	*Galaxias anomalus*
Gobiomorphus gobioides	*Galaxias depressiceps*
Gobiomorphus aff. *breviceps*	*Galaxias brevipinnis*
	Galaxias Lower Clutha
	Galaxias Pomohaka
	Ctenopharyngodon idella
	Hypophthalmichthys molitrix
	Oncorhynchus tshawytscha
	Oncorhynchus mykiss

Table 4. List of New Zealand freshwater fish species with sequences available for the mitochondrial 12S rRNA gene (12S), cytochrome b gene (cytb), cytochrome c oxidase subunit gene (COI), d-loop of the control region (d-loop), and the complete mitochondrial genome.

		Genome region				
Native species	Scientific name	12S rRNA	cyt b	COI	control region, d-loop	complete mitochondrion
Yellow-eyed mullet	*Aldrichetta forsteri*	y	y	y		y
Shortfin eel	*Anguilla australis*	y	y	y	y	y
Longfin eel	*Anguilla dieffenbachii*	y	y	y		y
Spotted/ Australian longfin eel	*Anguilla reinhardtii*	y	y	y	y	y
Kahawai	*Arripis trutta*	y	y	y		y
Torrentfish	*Cheimarrichthys fosteri*	y	y	y		
Estuarine triplefin aka *Grahamina* sp.	*Forsterigyion nigripenne*			y		
Roundhead galaxias	*Galaxias anomalus*		y		y	
Giant kokopu	*Galaxias argenteus*		y (300bp)		y	
Kōaro	*Galaxias brevipinnis*	y	y	y	y	
Lowland longjaw	*Galaxias cobitinis*		y		y	
Lowland longjaw galaxias (Waitaki River)	*Galaxias* aff*. cobitinis* “Waitaki”				y	
Taieri flathead galaxias	*Galaxias depressiceps*		y		y	
Dwarf galaxias	*Galaxias divergens*		y		y	
Dwarf galaxias (Nelson, Marlborough, and North Island)	*Galaxias* aff*. divergens* “northern”					
Dune Lakes galaxias (Kai Iwi Lakes)	*Galaxias “*dune lakes”					
Eldon’s galaxias	*Galaxias eldoni*		y		y	
Banded kokopu	*Galaxias fasciatus*	y	y		y	
Gollum galaxias	*Galaxias gollumoides*	y	y	y	y	y
Dune lake galaxias (= inanga)	*Galaxias gracilis (=maculatus)*				y	
Inanga	*Galaxias maculatus*	y	y	y	y	y

Native species	Scientific name	Genome region: 12S rRNA	cyt b	COI	control region, d-loop	complete mitochondrion
Bignose galaxias	*Galaxias macronasus*		y		y	
Nevis galaxias	*Galaxias* "nevis", Poss is *G. gollumoides*					
Waitaki alpine galaxias	*Galaxias n. sp.*					
Alpine galaxias	*Galaxias paucispondylus*		y		y	
Alpine galaxias (Southland)	*Galaxias* aff *paucispondylus* "Southland"					
Alpine galaxias (Manuherikia River)	*Galaxias* aff. *paucispondylus* "Manuherikia"					
Shortjawed kokopu	*Galaxias postvectis*		y		y	
Longjawed galaxias	*Galaxias prognathus*		y		y	
Upland longjaw galaxias	*Galaxias* aff. *prognathus* "Waitaki River"					
Dusky galaxias	*Galaxias pullus*		y		y	
Clutha flathead galaxias	*Galaxias* "species D"				y	
Lower Clutha galaxias	*Galaxias* "lower Clutha"					
Northern flathead galaxias	*Galaxias* "northern"		y			
Pomohaka galaxias (Pomohaka River)	*Galaxias* "Pomohaka"					
Southern flathead galaxias (Southland, Otago)	*Galaxias* "southern"	y	y	y	y	y
Teviot flathead galaxias	*Galaxias* "Teviot"		y		y	
Canterbury galaxias	*Galaxias vulgaris*		y		y	
Lamprey	*Geotria australis*	y	y	y	y	y
Tarndale bully	*Gobiomorphus alpinus*		y (360bp)			
Cran's bully	*Gobiomorphus basalis*		y (360bp)			
Upland bully (east coast South Island)	*Gobiomorphus breviceps*		y		y	
Upland bully (West Coast South Island, North Island)	*Gobiomorphus* aff. *breviceps*					

Native species	Scientific name	Genome region 12S rRNA	cyt b	COI	control region, d-loop	complete mitochondrion
Common bully	*Gobiomorphus cotidianus*	y	y			
Giant bully	*Gobiomorphus gobioides*		y (360bp)			
Bluegill bully	*Gobiomorphus hubbsi*		y	y		
Redfin bully	*Gobiomorphus huttoni*		y (360bp)			
Stargazer	*Leptoscopus macropygus*					
Grey mullet	*Mugil cephalus*	y	y	y	y	y
Brown mudfish	*Neochanna apoda*		y			
Canterbury mudfish	*Neochanna burrowsius*		y		y	
Black mudfish	*Neochanna diversus*		y		y	
Burgundy (Northland) mudfish	*Neochanna heleios*		y			
Chatham Island mudfish	*Neochanna rekohua*		y		y	
Grayling	*Prototroctes oxyrhynchus*					
Common smelt	*Retropinna retropinna*	y	y	y	y	y
Yellowbelly flounder	*Rhombosolea leporina*			y		
Black flounder	*Rhombosolea retiaria*					
Stokell's smelt	*Stokellia anisodon*	y	y	y		
Asian goby, striped sand goby	*Acentrogobius pflaumii*	y	y	y	y	
Bridled goby	*Arenigobius bifrenatus*	y	y	y		
Brown bullhead catfish	*Ameiurus nebulosus*	y	y	y	y	y
Goldfish	*Carassius auratus*	y	y	y	y	y
Grass carp	*Ctenopharyngodon idella*	y	y	y	y	y
European (koi) carp	*Cyprinus carpio*	y	y	y	y	y
Mosquitofish	*Gambusia affinis*	y	y	y		y
European gudgeon	*Gobio gobio*	y	y	y	y	y

Native species	Scientific name	Genome region				
		12S rRNA	cyt b	COI	control region, d-loop	complete mitochondrion
Glass goby	*Gobiopterus semivestitus*	y	y	y		
silver carp	*Hypophthalmichthys molitrix*	y	y	y	y	y
Golden orfe	*Leuciscus idus*	y	y	y	y	y
Rainbow trout	*Oncorhynchus mykiss*	y	y	y	y	y
Sockeye salmon	*Oncorhynchus nerka*	y	y	y	y	y
Chinook salmon	*Oncorhynchus tshawytscha*	y	y	y	y	y
Dart goby	*Parioglossus marginalis*	y				
European perch	*Perca fluviatilis*	y	y	y	y	y
Caudo	*Phalloceros caudimaculatus*	y		y		
Sailfin molly	*Poecilia latipinna*		y	y	y	y
Guppy	*Poecilia reticulata*	y	y	y	y	y
Atlantic salmon	*Salmo salar*	y	y	y	y	y
Brown trout	*Salmo trutta*	y	y	y	y	y
Brook char	*Salvelinus fontinalis*	y	y	y	y	y
Mackinaw	*Salvelinus namaycush*	y	y	y	y	y
Rudd	*Scardinius erythrophthalmus*	y	y	y	y	y
Tench	*Tinca tinca*	y	y	y	y	y
Swordtail	*Xiphophorus helleri*	y	y	y	y	y

3.2. 12S rRNA gene database extension

A search of GenBank revealed 41 species of New Zealand freshwater fish for which 12S rRNA sequences were not available. Tissue samples for 42 individuals (22 species) were provided by staff from the Universities of Auckland and Otago (Table 5). DNA for sequencing was extracted from approximately 50 mg of tissue obtained from morphologically identified specimens using a Qiagen DNeasy kit following the manufacturer's protocol. To amplify sequences that included both the MiFish and teleo portions of the 12S gene, the forward MiFish primer was paired with the reverse teleo primer for the following amplification step. A portion of the 12S rRNA gene was amplified using 10 µL of Go Taq (Promega, Madison, WI, USA, catalogue no. M1723), 1 µL each of MiFish-U-F (5'-GTC GGT AAA ACT CGT GCC AGC-3', (Miya et al. 2015) and teleoR (5'-CTT CCG GTA CAC TTA CCA TG-3', (Thomsen et al. 2016) (IDT Singapore), 6 µL of Ultrapure™ Dnase Rnase free water (Thermo Fisher, Waltham, MI, USA, catalogue number 10977023), and 2 µL of template DNA. For PCR amplification, the mixture was heated to 94 °C for two minutes, followed by 40 cycles of 94 °C for 30 seconds, 55 °C for 30 seconds and 72 °C for 45 seconds, with a final extension step at 72 °C for five minutes. Products from the PCRs were cleaned using a NucleoSpin® Gel and PCR clean-up kits (Macherey Nagel, Duren, Germany, catalogue no. 740609.50) and sent to the University of Waikato DNA Sequencing Facility for bi-directional sequencing. Sequences were edited manually in Geneious 9.1.8 (https://www.geneious.com) and submitted to GenBank (see Table 5 for accession numbers). Sequences were added to the alignment in Geneious and the file exported as a fastA file with no gaps in the sequences for use in the reference database (included in Appendix 7).

Table 5. Species sequenced in this study for the 12S rRNA gene and their GenBank accession numbers. TK = Tania King, University of Otago, GL = Gavin Lear, University of Auckland, AH = Andy Hicks, Hawkes Bay Regional Council, AP = Alton Perrie, Greater Wellington Regional Council, DK = David Kelly, Cawthron Institute.

Species	Provider	Collection accession number	Reaction number	GenBank accession number	Collection location
Galaxias aff. *paucispondylus*	TK	924-1	3412	MT952795	Waiau - Mararoa - Main Channel - Station Bridge Map GPS/ref D43 178 108
Galaxias aff. *paucispondylus*	TK	924-2	3413	MT952796	Waiau - Mararoa - Main Channel - Station Bridge D43 178 108
Galaxias divergens	TK	1071	3437	MT952797	Tutaki River Map GPS/ref E2469795 N5917771
Galaxias divergens	TK	1076	3436	MT952798	Wye River Map GPS/ref E1626077 N5384045
Galaxias eldoni	TK	1050-1	3439	MT952799	Deep Stream - Unnamed Tributary Map GPS/ref E2267297 N5495543
Galaxias eldoni	TK	1051-1	3438	MT952800	Deep Stream - Unnamed Tributary Map GPS/ref E2266649 N5498311
Galaxias fasciatus	TK	744-1	3408	MT952801	Pitt Island - Southern Tributary of Tupuangi Lagoon - Site 6 Map GPS/ref 44º14'44.9,S; 176º 12'10.5,W
Galaxias fasciatus	TK	745-1	3409	MT952802	Pitt Island - Third Water Creek - Near mouth - Site 12
Galaxias macronasus	TK	1056-1	3440	MT952803	Hakataramea Spring Map GPS/ref E2315832 N5610413
Galaxias n sp Waitaki Alpine	TK	572-1	3406	MT952804	Ahuriri - Braid at Main Road Bridge Map GPS/ref H29 713 333
Galaxias n sp Waitaki alpine	TK	572-2	3407	MT952805	Ahuriri - Braid at Main Road Bridge Map GPS/ref H29 713 333
Galaxias northern flathead	TK	1023-2	3443	MT952806	Clarence - True right tributary of Clarence below Bobs Stream Map GPS/ref E2573376 N5907916

Species	Provider	Collection accession number	Reaction number	GenBank accession number	Collection location
Galaxias northern flathead	TK	1023-1	3442	MT952807	Clarence - True right tributary of Clarence below Bobs Stream Map GPS/ref E2573376 N5907916
Galaxias paucispondylus	TK	309-1	3402	MT952808	Manuherikia - Below bridge - Site 3
Galaxias paucispondylus	TK	995-2	3444	MT952809	Ashburton River - Upper South Burn Map GPS/ref E2351349 N5751302
Galaxias paucispondylus	TK	309-2	3403	MT952810	Manuherikia - Below bridge - Site 3
Galaxias prognathus	TK	905-1	3445	MT952811	Wilberforce River Map GPS/ref E2379815 N5781581
Galaxias prognathus	TK	905-2	3446	MT952812	Wilberforce River Map GPS/ref E2379815 N5781581
Galaxias pullus	TK	1059-1	3448	MT952813	Bullocks Creek Map GPS/ref E1342868 N4946719
Galaxias pullus	TK	1238-1	3447	MT952814	Clutha River - Waitahuna River Tributary - Berwick Forest
Galaxias vulgaris	TK	1107-1	3451	MT952815	Waianakarua Map GPS/ref E2316589 N5551618
Galaxias vulgaris	TK	1107-2	3452	MT952816	Waianakarua Map GPS/ref E2316589 N5551618
Gobiomorphus aff. *breviceps*	TK	486-2	3405	MT952817	Waianakarua Lower Wairau - Goult 5 Map GPS/ref N28 254 519
Gobiomorphus aff. *breviceps*	TK	486-1	3404	MT952818	Waianakarua Lower Wairau - Goult 5 Map GPS/ref N28 254 519
Gobiomorphus breviceps	TK	1011-4	3453	MT952819	Mount Arrowsmith - Tarn? Map GPS/ref E2359502 N5747199
Neochanna burrowsius	TK	1046-1	3454	MT952820	Pareora River - Unnamed Tributary Map GPS/ref E2361596 N5636960

Species	Provider	Collection accession number	Reaction number	GenBank accession number	Collection location
Neochanna burrowsius	TK	1046-2	3455	MT952821	Pareora River - Unnamed Tributary Map GPS/ref E2361596 N5636960
Neochanna rekohua	TK	825-1	3410	MT952822	Chatham Islands - Lake Rekohua
Neochanna rekohua	TK	828-1	3411	MT952823	Chatham Islands - Tuku a Tamatea Stream
Galaxias argenteus	GL	GK	2544	MT952824	Unknown location
Galaxias argenteus	GL	GK	2545	MT952825	Unknown location
Galaxias postvectis	GL	SK	2548	MT952826	Unknown location
Galaxias argenteus	GL	GK	2536	MT952827	Unknown location
Gobiomorphus basalis	GL	CB	2543	MT952828	Unknown location
Gobiomorphus breviceps	GL	UB	2552	MT952829	Unknown location
Gobiomorphus breviceps	GL	UB	2553	MT952830	Unknown location
Gobiomorphus gobioides	GL	CB	2542	MT952831	Unknown location
Gobiomorphus hubbsi	GL	BB	2538	MT952832	Unknown location
Gobiomorphus huttoni	GL	RB	2546	MT952833	Unknown location
Gobiomorphus huttoni	GL	RB	2547	MT952834	Unknown location
Forsterygion nigripenne	GL	T2	2551	MT952835	Unknown location
Forsterygion nigripenne	GL	T2	2550	MT952836	Unknown location
Rhombosolea retiaria	AH	20201006-1	3660	MW187727	Unknown location
Neochanna apoda	AP	20201006-2	3661	MW187728	Unknown location
Scardinius erythrophthalmus	JB	20201006-3	3662	MW187726	Unknown location
Cheimarrichthys fosteri	TK	20201006-5	3665	MW187731	Unknown location
Cheimarrichthys fosteri	TK	20201006-6	3666	MW187730	Unknown location

3.3. Polymerase chain reactions

Two primer sets that had been developed internationally were chosen for comparison. These both target sections within the 12S rRNA gene. Mifish-U-F and Mifish-U-R (Miya et al. 2015) targets a region of approximately 220 base pairs, while Teleo-F and Teleo-R (Valentini et al. 2016) target a different region of approximately 100 base pairs.

An in vitro assessment of the performance of two primer sets was undertaken to determine how well each primer set amplified and discriminated New Zealand fish species using eDNA samples from a range of sites in the field pilot study. The fish-specific PCR primer pairs (MiFish-U-F/R (Miya et al. 2015) and Teleo-F/R (Valentini et al. 2016) were used to amplify fragments of the 12S rRNA gene. Reactions were run in triplicate to minimise the impact of stochastic PCR biases. The PCR reactions consisted of 10 µL MyFi Taq Mastermix (Meridian Bioscience, London), 1 µL of each of the forward and reverse primers, 6 µL sterile DNAse free water, and 2 µL of template DNA. Cycling conditions consisted of: an initial denaturation step at 95 °C for 2 min, followed by 40 cycles of denaturation at 95 °C for 30 s, annealing at 55 °C for 30 s and extension at 72 °C for 45 s, with a final extension of 72 °C for 5 min. Following PCR, the triplicate reactions were combined into a single well and PCR products were visualised on a 1% agarose gel. Library preparation steps were undertaken upon confirmation of successful amplification in samples and the absence of PCR product in negative controls. High throughput sequencing was undertaken at Auckland Genomics using an Illumina MiSeq instrument and the bioinformatic pipeline outlined in Appendices 5 and 6 was used to process the samples. Standard operating procedures and protocols for the PCRs are included in Appendix 3.

Samples from the initial pilot study (at six stream sites) were supplemented by opportunistic samples collected from 10 other sites in the Tasman region collected at the same time of year and processed in the same way as the initial pilot site data (Banks et al. in review). Results highlighted differences in the performance of the MiFish and teleo primer pairs in their ability to distinguish New Zealand freshwater fish species (Table 6). The MiFish primer pair unambiguously assigned sequences to species level more often than the teleo primer set. The teleo primer pair distinguished more bully species than the MiFish primer pair, but produced a large number of sequences that could not be assigned beyond genus (for example, both longfin and shortfin eel were detected in Orphanage Stream using electric fishing, whereas eDNA sample 20190319-4 contained sequences from both eel species using the teleo primer set, but most of the sequences generated were unassigned by the software (Table 7).

Table 6. Species lists of sequences assigned to species with each of the MiFish and teleo primers. The total pool of species at each site from the New Zealand Freshwater Fish Database (NZFFD) is included as an indication of the potential species that could be expected, noting, however, that the NZFFD includes records over time so may overestimate diversity at a site.

Site	Total pool of species from NZFFD	Sequences assigned to species with MiFish primers	Sequences assigned to species with teleo primers
All combined		Yellow-eyed mullet, Brown bullhead catfish, Shortfin eel, Longfin eel, Goldfish, Common carp, Banded kokopu, Inanga, Gambusia, Lamprey, Upland bully, Common bully, Redfin bully, Estuarine triplefin, Chinook salmon, Perch, Smelt, Brown trout, Tench	Yellow eyed mullet, Brown bullhead catfish, Shortfin eel, Longfin eel, Goldfish, Inanga, Gambusia, Crans Bully, Upland bully, Estuarine triplefin, Grey mullet, Chinook salmon, Perch, Smelt, Brown trout, Rudd, Tench
Brook Stream	Longfin eel, Shortfin eel, Brown trout, Upland bully, Redfin bully, Inanga, Torrentfish, Kōaro	Longfin eel, Shortfin eel, Brown trout	Longfin eel, Shortfin eel, Brown trout, Smelt
Maitai River	Longfin eel, Shortfin eel, Brown trout, Smelt, Upland bully, Common bully, Giant bully, Redfin bully, Inanga, Kōaro, Estuarine triplefin, Torrentfish, Common bully	Longfin eel, Shortfin eel, Brown trout, Smelt, Lamprey, Goldfish, Inanga, Estuarine triplefin	Longfin eel, Shortfin eel, Brown trout, Smelt
Lud River	Longfin eel, Inanga, Kōaro, Common bully, Giant bully, Redfin bully, Brown trout	Longfin eel, Shortfin eel	Longfin eel, Shortfin eel, Inanga
Orphanage Stream	Longfin eel, Shortfin eel, Inanga, Yellow-eyed mullet, Grey mullet, Upland bully, Common bully, Banded kokopu, Redfin bully, Smelt, Black flounder, Giant bully, Torrentfish, Giant kokopu, Tench, Kōaro	Longfin eel, Shortfin eel, Inanga, Common bully, Redfin bully, Estuarine triplefin	Longfin eel, Shortfin eel, Inanga, Gambusia, Cran's bully
Mystery Creek	Longfin eel, Shortfin eel, Gambusia, Common bully	Longfin eel, Shortfin eel, Gambusia, Common bully, Brown bullhead catfish	Longfin eel, Shortfin eel, Gambusia, Brown bullhead catfish
Komakorau Stream	Longfin eel, Shortfin eel, Brown bullhead catfish, Torrent fish, Koi carp, Grass carp, Giant kokopu, Banded kokopu, Inanga, Gambusia aff., Common bully, Black mudfish	Longfin eel Shortfin eel, Inanga, Gambusia aff., Smelt	Longfin eel, Shortfin eel, Inanga, Smelt, Gambusia, Rudd

Site	Total pool of species from NZFFD	Sequences assigned to species with MiFish primers	Sequences assigned to species with teleo primers
Riwaka at Tennis Courts		Shortfin eel, Gambusia, Estuarine triplefin, Common carp, Inanga	Shortfin eel, Gambusia
Riwaka at Staples Rd		Longfin eel, Shortfin eel, Gambusia Perch, Inanga, Common bully	Longfin eel, Shortfin eel, Gambusia, Inanga
Motueka Golf course		Longfin eel, Shortfin eel, Gambusia, Chinook salmon	Longfin eel, Shortfin eel, Gambusia Inanga
Riwaka Drain		Goldfish, Gambusia, Inanga, Shortfin eel	Inanga, Chinook salmon, Gambusia, Shortfin eel
Batchelor Rd		Longfin eel, Shortfin eel, Gambusia, Common bully, Inanga, Yellow-eyed mullet, Estuarine triplefin, Banded kokopu	Longfin eel, Shortfin eel, Grey mullet, Estuarine triplefin, Inanga, Gambusia, Yellow-eyed mullet, Upland bully,
Hursthouse St Drain		Longfin eel, Shortfin eel, Common bully, Inanga, Gambusia, Brown trout	Longfin eel, Shortfin eel, Inanga, Gambusia
Moutere Arm Creek		Longfin eel, Shortfin eel, Tench, Goldfish, Gambusia, Inanga,	Longfin eel, Shortfin eel, Inanga, Gambusia, Tench, Goldfish
Jubilee Bridge drain		Gambusia, Tench, Shortfin eel, Common bully, Inanga	Inanga, Gambusia, Shortfin eel
Moutere River at riverside community		Longfin eel, Shortfin eel, Common bully Redfin bully, Smelt, Inanga, Banded kokopu, Gambusia	Longfin eel, Shortfin eel, Smelt, Gambusia
Moutere River	Longfin eel, Shortfin eel, Torrent fish Kōaro, Banded kokopu, Inanga, Common bully, Giant bully, Bluegill bully, Redfin bully, Smelt, Tench	Longfin eel, Shortfin eel, Tench, Goldfish, Smelt, Inanga, Gambusia	Shortfin eel, Inanga, Gambusia, Tench, Goldfish

Table 7. The number of sequences from the field pilot study that were assigned to various taxonomic levels are indicated. Note that the sequences assigned to fish species includes the assignment of lamprey, which is taxonomically distinct from bony fish.

	MiFish primers		Teleo primers	
Taxonomic level sequences assigned to	Number of sequences	Percentage of sequences	Number of sequences	Percentage of sequences
Fish species	1,984,637	44.9	2,386,577	36.1
Fish genera	811,722	18.4	682,133	10.3
Fish class to order	13,556	0.3	667,565	10.1
Non-fish	1,608,768	36.4	2,867,631	43.4
Total	4,418,683		6,603,906	

The in silico investigation of the 12S rRNA gene indicated that the teleo primer pair would amplify a region better suited to the discrimination of bullies (*Gobiomorphus* sp.) than MiFish primers, while the MiFish primers would amplify a region better suited to discriminate other fish taxa. Sequencing the entire 12S region between the MiFish-F and teleo-R primers, including both the MiFish and teleo amplicon regions may enable the discrimination of both sets of taxa, however, this region is approximately 700 base pairs long. Illumina sequencing technology has limitations on the length of DNA fragment that can be sequenced (the maximum sequence length is approximately 550 bases when using 2 × 300 bp chemistry) (Illumina 2020). In contrast, the Oxford Nanopore long read sequencing platform MinION can sequence reads that are hundreds of kilobases long (Jain et al. 2016). Trials were undertaken with the MinION using the MiFish-U-F and teleoR primers to amplify the entire 12S region of interest. PCRs using the MiFish-U-F and teleoR primers were successful in isolation, however, PCRs using each primer with MinION adaptor sequences (which enable library preparation and sequencing on the MinION device) were unsuccessful despite changing the PCR amplification conditions. As a result, further attempts to use this sequencing platform were abandoned in favour of the Illumina sequencing platform.

3.4. Library preparation and high throughput sequencing

The PCR product was purified to remove unincorporated nucleotides and reagents that can interfere with the sequencing process and normalised using SequalPrep™ Normalisation plates (ThermoFisher, Waltham, MA, USA). Normalisation of the sample concentrations allows them to be pooled for sequencing.

Purified and normalised PCR product was transferred to a new sterile DNA/RNA free plate and sealed using a plate sealer and sent for sequencing at Auckland Genomics,

The University of Auckland, on an Illumina MiSeq platform. On arrival at Auckland Genomics, Nextera indices were added, quality control undertaken, and the samples pooled for sequencing. The PCR product for the MiFish primer set was sequenced using 2 × 250 base paired-end sequencing chemistry, while the Teleo primer set PCR product was sequenced using 2 × 150 base paired-end sequencing chemistry. This difference is due to the length of the PCR product being sequenced.

4. BIOINFORMATICS

Early approaches in eDNA metabarcoding used clustering approaches to group sequences together prior to assigning taxonomy (a process called OTU clustering). Recent evidence suggests that this clustering approach yields artificially high diversity and is not easily scalable or comparable among studies (Callahan et al. 2017). Instead, amplicon sequence variants (ASVs) are recommended as they encode the exact sequences, allowing comparison among studies and this approach is more easily scaled with larger datasets without significantly increased computing requirements (Callahan et al. 2017; Edgar 2018). In particular, the dada2 package makes use of ASVs from the data, has higher resolution than OTU clustering methods, is scalable and the package is open source, enabling the pipeline to be run by any provider who has access to the reference database (Callahan et al. 2016). Pipeline accessibility was a key consideration in the development of the bioinformatic process for this study to ensure a tool that could be used by any provider using openly available software and packages.

4.1. Bioinformatic analysis

Bioinformatic analysis of the sequences was undertaken using packages within R (version 4.0.2 ;Team 2020). The general pipeline consisted of importing sequences from BaseSpace (Illumina's results sharing platform) into R, removing the primer sequences from the ends of the sequences, filtering out poor quality reads, undertaking error profiling and de-noising steps, merging the forward and reverse reads, and removing chimeras (Appendix 4). Following the initial filtering and data preparation steps, taxonomic assignment was undertaken using the dada2 package (Callahan et al. 2016) and the reference database constructed earlier.

Taxonomic assignments were undertaken using the assignTaxonomy function in the dada2 package, with a minimum bootstrap value of 80 (Wang et al. 2007). The sequences were then combined with the metadata from fieldwork using the phyloseq package and the dataset subset to exclude all non-fish taxa. Data were conglomerated by species using the tax_glom function in phyloseq to simplify downstream presence-absence tables. The remaining sequences were used to generate presence-absence tables for each species and site. Non-conglomerated tables were also generated to enable manual assessment of the sequences that were unassigned to species level.

To compare the relative abundance of the reads from each species, the data were rarefied to an even sampling depth of 4000 reads per sample. Presence-absence tables and relative read numbers were calculated from the rarefied dataset. All code relating to the bioinformatic steps is included in Appendices 5 and 6 (for the MiFish and teleo primer sets, respectively).

Species presence-absence results for the eDNA samples were compared with the electric fishing results for the same site. Relative reads of each taxon were plotted against the approximated relative biomass of fish at each site.

5. METHODS VALIDATION

5.1. Community detection

Fish communities detected using the eDNA metabarcoding protocol (using the MiFish primer pair) were compared with electric fishing results from validation sites for which electric fishing and eDNA sample collection were undertaken concurrently (n = 9), within one month of eDNA sampling (n = 6), or more than one month apart (n = 1).

The eDNA collection protocol consisted of filtering up to 5.5 L of water through a 5 µm pore size self-preserving PES filter using the Smith Root backpack sampler. Filters were then stored at room temperature until DNA extraction. Samples were processed using the DNA extraction, PCR and bioinformatic protocols outlined in Appendices 2 to 4. The electric fishing method consisted of single pass fishing of a stream reach measuring 35 m to 150 m long and recording fish presence and the length or relative size of fish (e.g. small, medium, large) following the protocols of Joy et al. (2013).

The eDNA metabarcoding approach detected species that were not detected using electric fishing at seven sites and electric fishing detected species not identified with eDNA at 12 sites (Table 8). At most sites where eDNA detected fewer species than electric fishing, the difference was due to unassigned sequences belonging to either bullies or galaxiids.

The most detected taxa were longfin and shortfin eel for both sampling methods. Both species were discriminated using eDNA where they were present at a site, although there were three sites where one of the species was not detected despite them being caught using electric fishing (shortfin eel at Upper Wainui, shortfin eel at Roding River upstream of weir, and longfin eel at Kaupokonui upstream of culvert). However, these three sites did not have electric fishing undertaken concurrently with eDNA sample detection, which may explain the discrepancy.

Torrentfish were detected in one eDNA sample from one of the two sites where electric fishing and eDNA were both undertaken. A further eDNA sample detected torrentfish sequences from one site in the Waikato, which has previously recorded torrentfish presence.

Kōaro were caught at four sites and sequences (using the MiFish primer pair) were not detected in the eDNA samples, however, at three of the four sites there were unassigned galaxias sequences. These sequences were most similar to kōaro and *Galaxias* sp. “southern” when using an NCBI Blast search, although the differences were more than 3% in all cases. Intraspecific genetic differences of more than 3% can indicate genetically isolated populations and may indicate cryptic species within a morphologically similar group (see for example, Hardy et al. 2011).

Redfin bully sequences were assigned to species at all sites where they were found with electric fishing. Common bully sequences were assignable to species level at all but one site, and upland bully sequences were assigned at all but two sites where they were found by electric fishing. Alignment of the unassigned bully sequences against the reference database showed that they most closely matched either redfin bully or upland bully; however, there were a number of sequence differences which resulted in an inability to unambiguously assign the sequences to either taxon using the bioinformatic pipeline alone.

Table 8. A comparison of fish communities detected using electric fishing compared with eDNA metabarcoding.

Site	Region	Species detected with both methods	Species detected only by electric fishing	Species detected only by eDNA
Upper Wainui[a]	Waikato	Longfin eel Redfin bully Banded kokopu	Shortfin eel	
Waingongoro upstream of culvert	Hawke's Bay	Longfin eel Redfin bully	Kōaro (broad-finned galaxias)	Shortfin eel Unassigned galaxias[#]
Waingongoro downstream of culvert	Hawke's Bay	Redfin bully Longfin eel	Common bully Kōaro (broad-finned galaxias) Torrentfish	Shortfin eel
Waingongoro at Cabbage Tree Flat	Hawke's Bay	Longfin eel		
Lower Moutere	Tasman	Longfin eel Shortfin eel Redfin bully Unassigned bully Inanga (common galaxias) Common smelt Common bully	Upland bully	
Reservoir Creek upstream of Hill St	Tasman	Longfin eel Shortfin eel Inanga (common galaxias) Common bully	Kōaro (broad-finned galaxias) Banded kokopu	
Roding River upstream of weir[b]	Nelson	Longfin eel	Shortfin eel Upland bully	Unassigned bully[#] Unassigned galaxias[#]
Poorman Valley Stream at Whakatu Drive	Nelson	Longfin eel Shortfin eel Redfin bully Unassigned bully Inanga (common galaxias) Common smelt	Upland bully	Unassigned bully[#] Chinook salmon^

Site	Region	Species detected with both methods	Species detected only by electric fishing	Species detected only by eDNA
		Common bully		
Kaupokonui above weir[a]	Taranaki	Shortfin eel Unassigned trout	Longfin eel	Redfin bully
Kaupokonui below weir[a]	Taranaki	Longfin eel Shortfin eel Redfin bully Inanga (common galaxias) Torrentfish Unassigned trout	Unassigned bully	
Kaupokonui upstream of Fonterra[a]	Taranaki	Shortfin eel Brown trout	Unassigned eel Unassigned trout	Longfin eel Redfin bully
Kaupokonui downstream of Mangawhero-iti[a]	Taranaki	Longfin eel Shortfin eel	Unassigned trout Unassigned eel	
Kaupokonui upstream of Mangawhero-iti[a]	Taranaki	Longfin eel	Brown trout	
Grays River Tributary	Canterbury	Longfin eel Upland bully Bignose galaxias Brown trout	Kōaro (broad-finned galaxias)	Unassigned galaxias[#]
Upper Irishmans Creek	Canterbury	Upland bully Brown trout	Canterbury galaxias	Unassigned bully[#] Bignose galaxias
Little Joseph Creek	Canterbury	Upland bully	Rainbow trout Kōaro (broad-finned galaxias)	Longfin eel Unassigned trout[#] Unassigned bully[#] Unassigned galaxias[#]

[#] denotes detection of DNA sequences that could not be unambiguously assigned to species level.
[a] indicates electric fishing samples for the site were not collected on the same day but within one month of the eDNA sample.
[b] indicates electric fishing samples for the site were collected more than a month before eDNA sampling. ^ denotes that the species detection is unlikely due to the presence of individuals from this species at this site, but rather may indicate environmental contamination with DNA from this species (for example from food-borne contamination or the relatively close proximity of chinook salmon processing facilities to the sampling site). Sites with no superscript letter had eDNA samples collected concurrently with electric fishing.

5.2. Relative read abundance and relative biomass

Fish biomass was calculated using a quadratic power function at sites where regional data were available to inform predictive relationships (Waikato and Hawke's Bay sites) and using a linear power function from Jellyman et al. (2013) at all other sites where fish length data were available. Relative biomass was calculated as the proportion of total biomass of any given species present and compared to the relative abundance of reads at 6 sites where electric fishing was conducted on the same day as eDNA sampling.

The normality of the data was tested using a Shapiro-Wilk test in R. As the data were non-normally distributed, a log-transformation was undertaken; however, the data distribution remained non-normally distributed. As a result, non-parametric tests were used for subsequent analysis. Spearman's rank correlation was used to test the correlation between the relative biomass of the fish species and the relative read numbers. Overall, there was a weak relationship between the relative abundance of reads and the relative biomass of species collected during electric fishing if all taxa that were not identified to species from both electric fishing and eDNA were excluded (Figure 13). This relationship was driven by longfin eel.

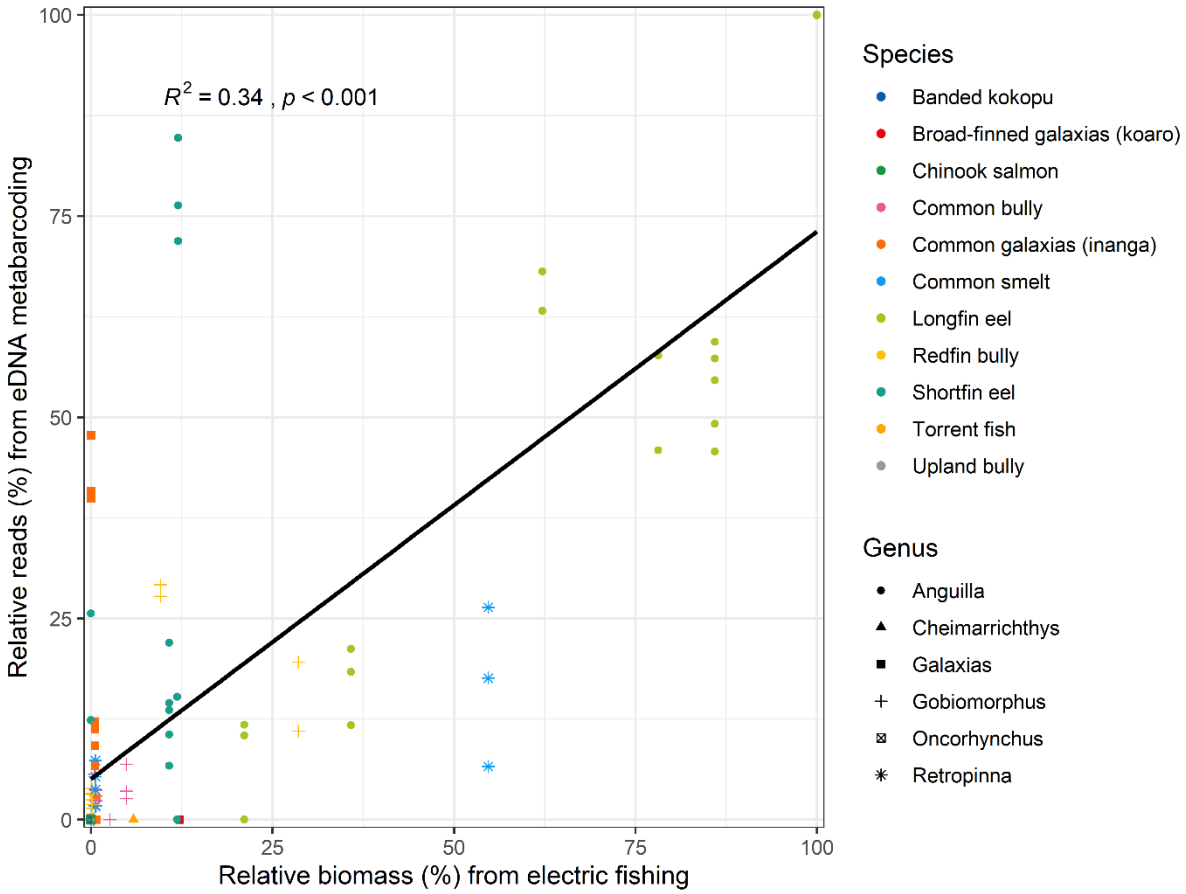


Figure 13. Combined data from 6 sites of the relative reads from eDNA metabarcoding and the relative biomass of different fish taxa from electric fishing.

Because biomass was calculated using different methods (e.g. linear vs quadratic, fish length measured vs fish size estimated, 35 m vs 150 m stream length fished) we explored site-specific relationships. Potential site-specific relationships between relative biomass and relative reads were illustrated (Figure 14), although the data are skewed by a single species (longfin eel) that is highly abundant at these sites. Correlation coefficients were calculated using Spearman's rank correlation.

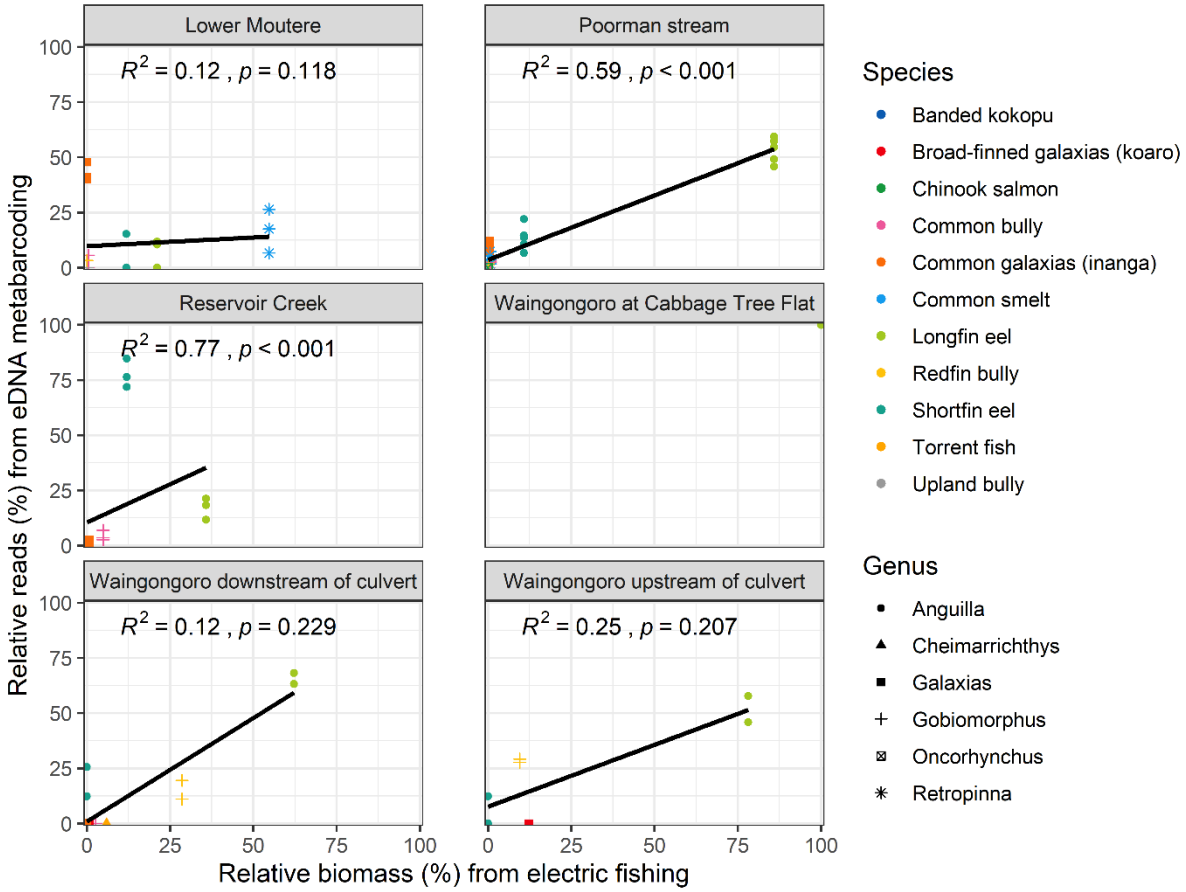


Figure 14. Site-specific relationships between relative reads from eDNA metabarcoding and the relative biomass of different fish taxa from electric fishing. Correlation coefficients were calculated using Spearman's rank correlation.

6. DISCUSSION

6.1. eDNA isolation and sample collection

The volume of water filtered determined the consistency of the number of fish species detected. The detection of species was the same for both 3-L and 5-L volumes filtered, suggesting both volumes are sufficient to enable detection of the community present. We recommend that water is collected until the filter clogs or the flow rate across the filter decreases below 1 L/min at 12 PSI pressure. At most sites, this method resulted in more than 3 L of water passing over the filter, which based on the results obtained in the trial will result in maximum species detection.

Conclusion: Maximising the volume of water using a minimum of three replicate 5 µm pore filters (four or more filters are required if filtrate volumes are < 3 L) will improve the probability of detecting all species present.

6.2. Community detection

The eDNA metabarcoding workflow using the MiFish primers presented in this study performed similarly to electric fishing at characterising fish communities, although more work is required to reach equivalence with electric fishing. As with any method, including electric fishing there are instances of non-detections using eDNA metabarcoding. Although the workflow is able to characterise many species, there were several species that could not be assigned unambiguously with the method used. Of primary concern is the lack of detection of kōaro and of some species of bullies. Blast searches of the unassigned galaxias sequences at sites where kōaro were confirmed as present by electric fishing indicated that these sequences are most likely from kōaro (although the sequence similarity is less than 97%). That these sequences were unassigned by the taxonomy assignment algorithm may be indicative of genetic variability in kōaro populations that has not been captured by the reference database.

Bully species could also not be resolved to species level for many sequences. Sequences at two Canterbury sites were assigned to upland bully; however, we were unable to assign sequences to upland bully for other sites at which they were identified using electric fishing. It is possible that there is intraspecific variability within bullies that has not yet been captured by the reference database, resulting in the inability to assign the sequences to species. There is evidence from amplified fragment length polymorphism (AFLP) analysis in common bully (*G. cotidianus*) that genetic variation can form in bully species even in populations with close geographic proximity (Michel et al. 2008). Few other studies examining intraspecific variability in New Zealand freshwater fish could be found. The development of sequencing

technologies enabling both amplicon and whole-genome sequencing presents opportunities to explore these intraspecific differences in more detail.

A potential route for further development may come through technological advancements in long-read sequencing. As a whole, the 12S rRNA gene is very informative for all New Zealand native fish, although the region amplified by the MiFish primer pair is less informative for bullies than for other species, and the region amplified by the teleo primer pair is less informative for the non-bully taxa. A recent paper by Shelley et al. (2020) used 8094 single nucleotide polymorphisms (SNPs) to distinguish *Gobiomorphus basalis, G. cotidianus*, and *G. alpinus.* Shelley et al. found evidence for consistent genetic differences within *G. basalis* and within *G. breviceps.* However, generating these large datasets using standard metabarcoding approaches is not possible currently due to instrument limitations, nor affordable, for routine population monitoring. Analysis of these large data sets also present an added cost to routine monitoring.

Ideally, the ability to metabarcode communities using the whole 12S rRNA gene may overcome some of the issues of distinguishing *G. basalis, G. breviceps* and *G. gobioides*. Illumina MiSeq is not able to sequence such long amplicons; however, there are two long-read sequencing platforms for high-throughput sequencing that are both rapidly advancing: MinION Nanopore Sequencing and PacBio sequencing. Currently the costs (in the case of PacBio) and the sequencing accuracy (MinION) are barriers to the implementation of these long-read technologies in routine monitoring. This is an area that is rapidly developing and there is considerable potential for improvements to contribute to eDNA monitoring of fish communities in future.

The importance of a robust and complete reference database was recently demonstrated by Schenekar et al. (2020), who re-analysed a dataset using an expanded reference database and found several sequences assigned incorrectly, and species that were not detected in the initial study. Many of the bully sequences in our study were not assignable to species level. At some sites where only a single species is known to be present, it could be possible to use known distributions to manually assign these sequences to the appropriate taxon using a weight of evidence approach (Bylemans et al. 2018), and labelling those sequences assigned using a weight of evidence approach, *sensu lato,* (i.e., in the broad sense). Although the tool workflow does not explicitly state which of the species within a genus are most likely, it is possible to manually compare the unassigned sequences to reference sequences (for example in NCBI Blast) and return lists of the possible matches and their percent similarity. This combined with knowledge of a site or river may enable the likely identity of the unassigned species to be inferred if the sequences are similar enough to those being compared with NCBI Blast. It would be preferable, however, to sequence tissue samples of morphologically identified specimens to increase the representation of these taxa in the reference database. This is because if using a weight of evidence approach, any assignments that were incorrect would be hard-

coded into the reference database, and this would then impact taxonomic assignments in future. However, using the *sensu lato* epithet would distinguish sequences assigned an identity derived from a voucher specimen from those sequences assigned an identity by inference.

Conclusion: Expansion of the 'living' reference database to include sequences from voucher specimens from across the geographic range identified by experts will ensure a robust bioinformatics procedure can be adopted that does not rely on BLAST searches or the manual assignment of taxa based on weight of evidence.

6.3. Biomass assessment

There were indications of a relationship between relative proportion of reads and relative biomass at some sites, but not at others. The sites for which there were apparent relationships had a high biomass and proportion of reads from eel species (particularly longfin eel; *Anguilla dieffenbachii*). We found no general relationship between the relative biomass of different species and the relative abundance of their sequences using eDNA metabarcoding. Given the difference between the individual site patterns and the general pattern, further sampling including multiple samples over time at a single site would be needed to determine whether this relationship holds true at a site or was generated by chance.

The lack of a general relationship between relative biomass and reads is unsurprising as the eDNA sample integrates the community from upstream, meaning the catchment area is potentially larger than the electric fishing reach of 150 m. Environmental DNA can act like fine particulate organic matter and has been shown to be transported over kilometres in small streams and more than 100 kilometres in large rivers (Pont et al. 2018). There is considerable debate in the literature around the ability to use read numbers from eDNA metabarcoding to infer abundance of fish in the environment (Goldberg et al. 2016; Snyder & Stepien 2020). Positive correlations among read numbers and abundance or biomass of organisms have been identified in some studies (Hänfling et al. 2016; Thomsen et al. 2016), whilst correlations were not observed in others (Shaw et al. 2016; Gillet et al. 2018). Chambert et al. (2018) used an experimental dataset in which the abundance of animals (common carp) was known without error, combined with eDNA metabarcoding results to model animal density. While the models performed well on the experimental dataset, their performance declined when applied to an environmental dataset. The authors highlighted the need to include the uncertainty in traditional sampling methods in models, otherwise there is a risk of bias and erroneous confidence in eDNA-based estimates of the abundance of target taxa.

Although relationships between biomass and read numbers have been found using eDNA metabarcoding for single species (Elbrecht & Leese 2015), the abundance of

the organism cannot be inferred because one large organism can shed the same amount of DNA as many small ones and the number of copies of mitochondrial DNA varies among species. Elbrecht and Leese (2015) argue that PCR biases among different species mean that estimates of biomass from eDNA metabarcoding read numbers from diverse environmental samples is not possible and that metabarcoding should be used for presence-absence information only.

Conclusion: Metabarcoding is not the preferred approach to determine the biomass of fish species. Much more research is required to determine the ecology of eDNA (origin, state, transport, and fate) for different species.

6.4. Primer performance

We found differences in the discrimination of species among the two primer sets tested. The MiFish primer set performed better at species level discrimination across most taxonomic groups, although it performed poorly for taxonomic assignments within bullies. In contrast, the teleo primer set performed well at enabling the assignment of bully sequences to species level, but it performed poorly and there were many unassigned sequences among other taxonomic groups. Our results are similar to those of Bylemans et al. (2018), who found lower taxonomic resolution of the teleo primer set than the MiFish primer set for species in the Murray-Darling Basin. These differences reflect the trade-offs of different primer sets and highlight that there is no truly universal metabarcoding primer set. McElroy et al. (2020) suggested sequences from several genome regions (i.e., the use of more than one primer pair) improve the accuracy of estimates of fish biodiversity, and our findings suggest this is also the case in New Zealand.

There are four key considerations for primer choice in eDNA metabarcoding: primer length, taxonomic resolution, specificity to the target organisms and amplification efficiency (or potential for primer bias). Primers should aim to amplify a short fragment of DNA (< 200 base pairs long) to maximise recovery from eDNA samples that might be degraded (Coissac et al. 2012). They must have sufficient resolution to discriminate the taxa of interest. They should be specific to the taxa of interest to minimise non-specific amplification and the subsequent sequencing of non-target taxa. They should have similar efficiency across the taxa of interest (i.e. they should aim to minimise primer biases) to ensure consistent detection probabilities among species (Polz & Cavanaugh 1998; Elbrecht & Leese 2015). However, there are trade-offs among these four requirements for primer design: shorter amplicon regions reduce the taxonomic resolution of the primer set because there are fewer differences among species. Similarly, reducing primer bias by minimizing mismatches among the taxa of interest can result in decreased specificity of the primer set, thereby increasing the amplification of non-target taxa. Validation of the MiFish primer set both in silico for this study and among other studies has revealed that the primer binding region is

highly conserved among fish taxa, which minimises the impact of primer bias on the data. Even among metabarcoding primer sets developed specifically for the identification of fish taxa within an eco-region, there were a proportion of sequences (up to 35%) that could not be assigned to species level and up to 22% of sequences that could not be assigned to genus (Bylemans et al. 2018).

Conclusion: The MiFish primer set can characterise many New Zealand fish communities with Illumina sequencing. Additional primer sets should be investigated for distinguishing the bullies.

6.5. Bioinformatic analysis

The bioinformatic pipeline includes filtering steps at multiple stages of the pipeline. Initial raw sequences are filtered to remove low quality reads, primers are removed from the sequences (and any sequences not containing the primers also discarded), sequences outside the expected amplicon length are discarded and non-target taxa filtered from the dataset (Schloss et al. 2011). There is a trade-off between certainty of detected species being present and the stringency of the bioinformatic pipeline. The higher the stringency, the greater the certainty, but that is concomitant with a higher potential for false negatives (where a species is not defined as present), whereas a lower stringency leads to higher uncertainty and a greater likelihood of false positives. The defined stringency is usually context and study dependent (Evans et al. 2017). For example, an assessment should be made regarding the impact of false positives and false negatives on the outcome of the monitoring programme. A low stringency approach was used to define presence in this study; positive results in one of the three field-replicates were characterised as positive for that species to maximise detection rates. As this is a post hoc assessment, it is possible to change the chosen stringency during or following analysis without impacting any of the earlier steps, although it is recommended that the stringency is kept consistent among repeated samplings to ensure results are comparable.

As the MiFish primer set has not been validated for the detection of non-fish taxa and the study aim was to characterise fish community, all non-fish taxa were excluded from further analysis. This does not preclude the potential investigation of non-fish taxa in future to gain an understanding of catchment-level processes that contribute to the eDNA pool in streams and rivers. Significant development work would be required to understand the contribution of terrestrial DNA sources into the aquatic environment and how to interpret the results, nevertheless, this could be an interesting avenue for on-going investigation.

Conclusion: Open source pipelines are used for bioinformatic analysis and code is available to enable analyses to be updated if new pipelines or database sequences are added in future.

6.6. Summary and recommendations for future development

The tool developed in this study provides a robust workflow for the use of eDNA metabarcoding to characterise freshwater fish communities. The tool encompasses:

- protocols for the collection of fish eDNA samples in wadeable New Zealand streams with lowered risk of contamination
- validated primers to identify New Zealand freshwater fish species, which discriminate most taxa but noting issues with some bullies
- protocols for laboratory workflows and sequencing parameters
- a reference database of 12S rRNA sequences for fish taxa in New Zealand, which is a living library that should be added to
- code for the bioinformatic analysis of the raw sequence data, which is openly available (see Appendices).

Following the protocols and using the database and code provided in this tool will enable the characterisation of freshwater fish communities in New Zealand, with the caveat that some sequences currently can be assigned to genus but not to species. The approximate cost of undertaking these analyses in given in Appendix 8.

Each step of the workflow can be considered somewhat modular (Figure 15); that is, the protocols can be updated or exchanged as technology changes and matures, although, as with sampling protocol changes using other technologies, these should be validated by running them alongside the existing protocols to ensure the outcomes are comparable.

The development of this tool has led to the identification of priorities for future research, development, and extension of the use of molecular technologies in fish community monitoring. Firstly, there is a need for a clear framework to guide the choice of eDNA analysis depending on the question or required outcome of the analysis. For example, if quantitative information is required then a species-specific assay would be preferred over metabarcoding, whereas if community composition information is the objective then the metabarcoding approach is preferred. Where both quantitative and community composition information is desired, then a combination approach is recommended: metabarcoding first to identify which species are present, then targeted species-specific assays for only those taxa identified using metabarcoding. Gleeson and Hinlo (2020) summarise the factors around eDNA studies that should be considered before starting an eDNA study, including i) the selection of assays that only provide reproducible and repeatable results, ii) improved eDNA detection through improved eDNA survey design, iii) establishment of minimum reporting requirements, and iv) establishment of management decision frameworks.

A useful next step would be to conduct robust comparisons of species detection rates obtained using electric fishing and eDNA. Occupancy modelling is one option; the

technique provides a statistical framework for evaluating species distributions and site occurrences, and can estimate the false negative detection rates of the sampling procedure (failure to collect DNA from a species) or during the PCR steps (failure to amplify DNA of a species) (Schmelzle & Kinziger 2016; Doi et al 2019). Packages in Rare available to implement occupancy models and have been applied to fish eDNA studies (for example, Dorazio & Price 2018).

Second, ongoing development of the reference database is recommended. This development could be included within existing monitoring programmes in which fish are caught and identified. Swabs can be taken of these individuals and their DNA sequenced to add to the database. This would achieve the outcome of a 'living' database that continually expands to capture more of the genetic diversity of New Zealand's native fish. For now, reference sequences have been uploaded and are available via GenBank (https://www.ncbi.nlm.nih.gov/genbank/).

Given the degree of debate, and the weak relationship between read numbers and biomass that we found, we recommend that the relationship between biomass or abundance is investigated further for New Zealand fish species. An alternative to more research into the relationship between read numbers in community studies is to develop a series of species-specific quantitative PCR (qPCR) or droplet digital PCR (ddPCR) tests. Species-specific qPCR/ddPCR assays are typically more sensitive than metabarcoding. A potential workstream is to extract the eDNA from the water sample and use metabarcoding for community characterisation, followed by species-specific assays for species of interest. Once the results from the community characterisation by metabarcoding have been obtained, species specific qPCR/ddPCR can be used on taxa of interest or conservation concern (for example, key indicator species such as torrentfish, or enigmatic species such as lamprey). It should be noted that neither metabarcoding or species-specific assays directly quantify the number of individuals, nor biomass, but rather the number of copies of the DNA sequence of interest. This means that if the primary aim is to determine biomass or abundance, species-specific models will be needed that can relate the number of DNA copies to the variable of interest. Allometric scaling models have been used with some success for determining these relationships using targeted species-specific assays (Yates et al. 2020; Shelton et al. 2019), although the size structure of the population must be known in advance.

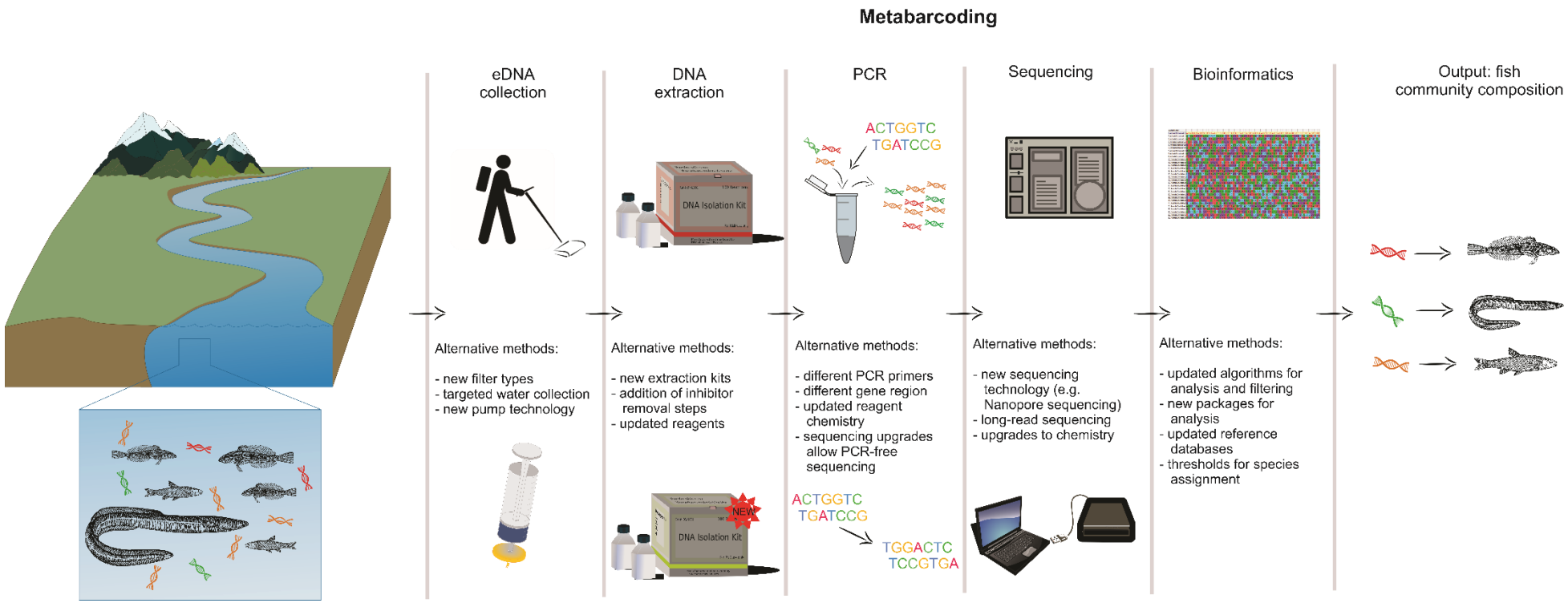


Figure 15. The modularity of the eDNA analysis workflow for metabarcoding. Each part of the process between the grey bars can be exchanged as molecular technologies continue improving, with examples of potential changes or updates that may occur in future included for reference.

7. ACKNOWLEDGEMENTS

This work was supported by an Envirolink Tools grant from the Ministry of Business, Innovation & Employment (CAWX1802). We thank the project advisory panel for direction and insight into the development of these molecular methods and for their review of draft reports, including Andy Hicks, Bruno David, Dave West, Dianne Gleeson, Gavin Lear and Austen Thomas. Our thanks to Jack Rojahn and Austen Thomas for technical support and Katie Collins for further report review. Further we thank Andy Hicks (Hawkes Bay Regional Council), Tania King (University of Otago), Gavin Lear (University of Auckland) and Alton Perrie (Greater Regional Wellington Council) for fish tissue samples and swabs from fish skin for sequencing. Finally, our thanks to the field teams involved in developing and testing the method including Robin Holmes, Rasmus Gabrielsson, Paul Fisher, Tom Kroos, Andy Hicks, Daniel Fake, Bruno David, and Paddy Deegan.

8. REFERENCES

Banks JC, Hogg ID 2014. Development and validation of hydrolysis assays for seven species of exotic fish. Environmental Research Institute Report No. 44. Client report prepared for Lake Ecosystem Restoration New Zealand (LERNZ). 17 p.

Banks JC, Demetras NJ, Hogg ID, Knox MA, West DW 2016. Monitoring brown trout (*Salmo trutta*) eradication in a wildlife sanctuary using environmental DNA. New Zealand Natural Sciences 41: 1-13.

Banks JC, Kelly LT, Falleiros R, Rojahn J, Gabrielsson R, Clapcott J In review. Detecting the pest fish, *Gambusia affinis* from environmental DNA: a comparison of methods. New Zealand Journal of Zoology.

Barnes MA, Turner CR 2016. The ecology of environmental DNA and implications for conservation genetics. Conservation Genetics 17(1): 1-17.

Bylemans J, Gleeson DM, Hardy CM, Furlan E 2018. Toward an ecoregion scale evaluation of eDNA metabarcoding primers: A case study for the freshwater fish biodiversity of the Murray–Darling Basin (Australia). Ecology and Evolution 8(17): 8697-8712.

Callahan BJ, McMurdie PJ, Holmes SP 2017. Exact sequence variants should replace operational taxonomic units in marker-gene data analysis. The ISME Journal 11(12): 2639-2643.

Callahan BJ, McMurdie PJ, Rosen MJ, Han AW, Johnson AJA, Holmes SP 2016. DADA2: High-resolution sample inference from Illumina amplicon data. Nature Methods 13(7): 581-583.

Chambert T, Pilliod DS, Goldberg CS, Doi H, Takahara T 2018. An analytical framework for estimating aquatic species density from environmental DNA. Ecology and Evolution 8(6): 3468-3477.

Coissac E, Riaz T, Puillandre N 2012. Bioinformatic challenges for DNA metabarcoding of plants and animals. Molecular Ecology 21(8): 1834-1847.

Coulter DP, Wang P, Coulter AA, Van Susteren GE, Eichmiller JJ, Garvey JE, Sorensen PW 2019. Nonlinear relationship between Silver Carp density and their eDNA concentration in a large river. PLOS ONE 14(6): e0218823..

Deiner K, Walser J-C, Mächler E, Altermatt F 2015. Choice of capture and extraction methods affect detection of freshwater biodiversity from environmental DNA. Biological Conservation 183: 53-63.

Doi H, Fukaya K, Oka S, Sato K, Kondoh M, Miya M 2019. Evaluation of detection probabilities at the water-filtering and initial PCR steps in environmental DNA metabarcoding using a multispecies site occupancy model. Scientific Reports 9: 3581.

Dorazio RM, Price M 2018. State-space models to infer movements and behavior of fish detected in a spatial array of acoustic receivers. Canadian Journal of Fisheries and Aquatic Sciences. 76(4): 543-550.

Dunn NR, Allibone RM, Closs GP, Crow SK, David BO, Goodman JM, Griffiths M, Jack DC, Ling N, Waters JM, Rolfe JR 2018. Conservation status of New Zealand freshwater fishes, 2017. 11 p.

Edgar RC 2018. Updating the 97% identity threshold for 16S ribosomal RNA OTUs. Bioinformatics 34(14): 2371-2375. doi:10.1093/bioinformatics/bty113.

Eichmiller JJ, Miller LM, Sorensen PW 2016. Optimizing techniques to capture and extract environmental DNA for detection and quantification of fish. Molecular Ecology Resources 16(1): 56-68.

Elbrecht V, Leese F 2015. Can DNA-based ecosystem assessments quantify species abundance? Testing primer bias and biomass—sequence relationships with an innovative metabarcoding protocol. PLOS ONE 10(7): e0130324.

Evans NT, Li Y, Renshaw MA, Olds BP, Deiner K, Turner CR, Jerde CL, Lodge DM, Lamberti GA, Pfrender ME 2017. Fish community assessment with eDNA metabarcoding: effects of sampling design and bioinformatic filtering. Canadian Journal of Fisheries and Aquatic Sciences 74(9): 1362-1374.

Ficetola GF, Miaud C, Pompanon F, Taberlet P 2008. Species detection using environmental DNA from water samples. Biology Letters 4(4): 423-425.

Furlan EM, Gleeson D, Hardy CM, Duncan RP 2015. A framework for estimating the sensitivity of eDNA surveys. Molecular Ecology Resources: n/a-n/a. doi:10.1111/1755-0998.12483.

Garnier S 2019. viridis: Default Color Maps from ,matplotlib.,. R package version 0.5.1 1.

Gillet B, Cottet M, Destanque T, Kue K, Descloux S, Chanudet V, Hughes S 2018. Direct fishing and eDNA metabarcoding for biomonitoring during a 3-year survey significantly improves number of fish detected around a South East Asian reservoir. PLOS ONE 13(12): e0208592..

Gleeson D, Hinlo R 2020. Targeting the science–management interface: a critique on the use of eDNA monitoring for the management of freshwater fish in New Zealand. Prepared for Department of Conservation. University of Canberra. 23 p.

Goldberg CS, Turner CR, Deiner K, Klymus KE, Thomsen PF, Murphy MA, Spear SF, McKee A, Oyler-McCance SJ, Cornman RS 2016. Critical considerations for the application of environmental DNA methods to detect aquatic species. Methods in Ecology and Evolution 7(11): 1299-1307.

Hänfling B, Lawson Handley L, Read DS, Hahn C, Li J, Nichols P, Blackman RC, Oliver A, Winfield IJ 2016. Environmental DNA metabarcoding of lake fish communities reflects long-term data from established survey methods. Molecular Ecology 25(13): 3101-3119.

Hardy CM, M Adams, DR Jerry, LN Court, MJ Morgan, DM Hartley 2011. DNA barcoding to support conservation: species identification, genetic structure and biogeography of fishes in the Murray—Darling River Basin, Australia. Journal of Marine and Freshwater Research 62(8): 887-901.

Hinlo R, Gleeson D, Lintermans M, Furlan E 2017. Methods to maximise recovery of environmental DNA from water samples. PLoS ONE 12(6): e0179251

Illumina 2020. Illumina sequencing platforms https://www.illumina.com/systems/sequencing-platforms.html

Jain M, Olsen HE, Paten B, Akeson M 2016. The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community. Genome Biol 17, 239.

Jane SF, Wilcox TM, McKelvey KS, Young MK, Schwartz MK, Lowe WH, Letcher BH, Whiteley AR 2015. Distance, flow and PCR inhibition: e DNA dynamics in two headwater streams. Molecular ecology resources 15(1): 216-227.

Jellyman PG, Booker DJ, Crow SK, Bonnett ML, Jellyman DJ 2013. Does one size fit all? An evaluation of length–weight relationships for New Zealand's freshwater fish species. New Zealand Journal of Marine and Freshwater Research 47(4): 450-468.

Joy M, David B, Lake M 2013. New Zealand freshwater fish sampling protocols. Massey University, Palmerston North, New Zealand.

Kandlikar G 2020. ranacapa: Utility functions and 'shiny' app for simple environmental DNA visualizations and analyses. R package version 0.1.0. https://github.com/gauravsk/ranacapa.

Klymus KE, Richter CA, Chapman DC, Paukert C 2015. Quantification of eDNA shedding rates from invasive bighead carp *Hypophthalmichthys nobilis* and silver carp *Hypophthalmichthys molitrix*. Biological Conservation 183: 77-84.

Lear G, Dickie I, Banks J, Boyer S, Buckley HL, Buckley TR, Cruickshank R, Dopheide A, Handley KM, Hermans S, Kamke J, Lee CK, MacDiarmid R, Morales SE, Orlovich DA, Smissen R, Wood J, Holdaway R 2018. Methods for the extraction, storage, amplification and sequencing of DNA from environmental samples. New Zealand Journal of Ecology 42(1).

Lecaudey LA, Schletterer M, Kuzovlev VV, Hahn C, Weiss SJ 2019. Fish diversity assessment in the headwaters of the Volga River using environmental DNA metabarcoding. Aquatic Conservation: Marine and Freshwater Ecosystems 29(10): 1785-1800.

Li Y, Evans NT, Renshaw MA, Jerde CL, Olds BP, Shogren AJ, Deiner K, Lodge DM, Lamberti GA, Pfrender ME 2018. Estimating fish alpha-and beta-diversity along a small stream with environmental DNA metabarcoding. Metabarcoding and Metagenomics 2: e24262.

Martin M 2011. Cutadapt removes adapter sequences from high-throughput sequencing reads. EMBnet. journal 17(1): 10-12.

Maruyama A, Nakamura K, Yamanaka H, Kondoh M, Minamoto T 2014. The release rate of environmental DNA from juvenile and adult fish. PLoS One 9(12): e114639.

McElroy ME, Dressler TL, Titcomb GC, Wilson EA, Deiner K, Dudley TL, Eliason EJ, Evans NT, Gaines SD, Lafferty KD, Lamberti GA, Li Y, Lodge DM, Love MS, Mahon AR, Pfrender ME, Renshaw MA, Selkoe KA, Jerde CL 2020. Calibrating environmental DNA metabarcoding to conventional surveys for measuring fish species richness. Frontiers in Ecology and Evolution 8(276): xx-xx. doi:10.3389/fevo.2020.00276.

McMurdie PJ, Holmes S 2013. phyloseq: an R package for reproducible interactive analysis and graphics of microbiome census data. PloS One 8(4): e61217.

Michel C, Hicks BJ, Stölting KN, Clarke AC, Stevens MI, Tana R, Meyer A, Van den Heuvel MR 2008. Distinct migratory and non-migratory ecotypes of an endemic New Zealand eleotrid (*Gobiomorphus cotidianus*)–implications for incipient speciation in island freshwater fish species. BMC Evolutionary Biology 8(1): 49.

Miya M, Sato Y, Fukunaga T, Sado T, Poulsen JY, Sato K, Minamoto T, Yamamoto S, Yamanaka H, Araki H, Kondoh M, Iwasaki W 2015. MiFish, a set of universal PCR primers for metabarcoding environmental DNA from fishes: detection of

more than 230 subtropical marine species. Royal Society Open Science 2(7): 150088. doi:10.1098/rsos.150088.

Morgan M, Anders S, Lawrence M, Aboyoun P, Pages H, Gentleman R 2009. ShortRead: a bioconductor package for input, quality assessment and exploration of high-throughput sequence data. Bioinformatics 25(19): 2607-2608.

Muha TP, Robinson CV, Garcia de Leaniz C, Consuegra S 2019. An optimised eDNA protocol for detecting fish in lentic and lotic freshwaters using a small water volume. PLOS One 14(7): e0219218.

Ogram A, Sayler GS, Barkay T 1987. The extraction and purification of microbial DNA from sediments. Journal of Microbiological Methods 7(2–3): 57-66.

Oksanen J, Blanchet FG, Kindt R, Legendre P, Minchin P, O'hara R, Simpson G, Solymos P, Stevens M, Wagner H 2019. vegan: Community ecology package. R package version 2.5-6. https://CRAN.R-project.org/package=vegan.

Olds BP, Jerde CL, Renshaw MA, Li Y, Evans NT, Turner CR, Deiner K, Mahon AR, Brueseke MA, Shirey PD, Pfrender ME, Lodge DM, Lamberti GA 2016. Estimating species richness using environmental DNA. Ecology and Evolution 6: 4214-4226.

Pagès H, Aboyoun P, Gentleman R, DebRoy S 2020. Biostrings: efficient manipulation of biological strings. R package version 2.56.0.

Peixoto S, Chaves C, Velo-Antón G, Beja P, Egeter B 2020. Species detection from aquatic eDNA: Assessing the importance of capture methods. Environmental DNA.

Polz MF, Cavanaugh CM 1998. Bias in template-to-product ratios in multitemplate PCR. Applied and Environmental Microbiology 64(10): 3724-3730.

Pont D, Rocle M, Valentini A, Civade R, Jean P, Maire A, Roset N, Schabuss M, Zornig H, Dejean T 2018. Environmental DNA reveals quantitative patterns of fish biodiversity in large rivers despite its downstream transportation. Scientific Reports 8(1): 1-13.

Poté J, Ackermann R, Wildi W 2009. Plant leaf mass loss and DNA release in freshwater sediments. Ecotoxicology and Environmental Safety 72(5): 1378-1383.

Quan P, Sauzade M, Brouzes E 2018. dPCR: A technology review. Sensors 18: 1271.

Rådström P, Knutsson R, Wolffs P, Lövenklev M, Löfström C 2004. Pre-PCR processing. Molecular Biotechnology 26(2): 133-146.

Ratnasingham S, Hebert PD 2007. BOLD: The Barcode of Life data system (http://www. barcodinglife. org). Molecular ecology notes 7(3): 355-364.

Rees HC, Gough KC, Middleditch DJ, Patmore JRM, Maddison BC 2015. Applications and limitations of measuring environmental DNA as indicators of the presence of aquatic animals. Journal of Applied Ecology 52(4): 827-831.

Renshaw MA, Olds BP, Jerde CL, McVeigh MM, Lodge DM 2015. The room temperature preservation of filtered environmental DNA samples and assimilation into a phenol–chloroform–isoamyl alcohol DNA extraction. Molecular Ecology Resources 15: 168-176.

Rossen L, Nørskov P, Holmstrøm K, Rasmussen OF 1992. Inhibition of PCR by components of food samples, microbial diagnostic assays and DNA-extraction solutions. International Journal of Food Microbiology 17(1): 37-45.

Saitou N, Nei M 1987. The neighbor-joining method: a new method for reconstructing phylogenetic trees. Molecular Biology and Evolution 4(4): 406-425.

Sayers EW, Cavanaugh M, Clark K, Ostell J, Pruitt KD, Karsch-Mizrachi I 2019. GenBank. Nucleic Acids Research 47(D1): D94-D99. doi:10.1093/nar/gky989.

Schenekar T, Schletterer M, Lecaudey LA, Weiss SJ 2020. Reference databases, primer choice, and assay sensitivity for environmental metabarcoding: Lessons learnt from a re-evaluation of an eDNA fish assessment in the Volga headwaters. River Research and Applications 36(7): 1004-1013.

Schloss PD, Gevers D, Westcott SL 2011. Reducing the effects of PCR amplification and sequencing artifacts on 16S rRNA-based studies. PLOS One 6(12): e27310.

Schmelzle MC, Kinziger AP 2016. Using occupancy modelling to compare environmental DNA to traditional field methods for regional-scale monitoring of an endangered aquatic species. Molecular Ecology Resources 16: 895-908

Seymour M, Durance I, Cosby BJ, Ransom-Jones E, Deiner K, Ormerod SJ, Colbourne JK, Wilgar G, Carvalho GR, de Bruyn M, Edwards F, Emmett BA, Bik HM, Creer S 2018. Acidity promotes degradation of multi-species environmental DNA in lotic mesocosms. Communications Biology 1(1): 4..

Shaw JL, Clarke LJ, Wedderburn SD, Barnes TC, Weyrich LS, Cooper A 2016. Comparison of environmental DNA metabarcoding and conventional fish survey methods in a river system. Biological Conservation 197: 131-138.

Shelley JJ, David BO, Thacker CE, Hicks AS, Jarvis MG, Unmack PJ 2020. Phylogeography of the Cran's bully *Gobiomorphus basalis* (Gobiiformes: Eleotridae) and an analysis of species boundaries within the New Zealand radiation of Gobiomorphus. Biological Journal of the Linnean Society 130(2): 365-381.

Shelton AO, Kelly RP, O'Donnell JL, Park L, Schwenke P, Greene C, Henderson RA, Beamer EM 2019. Environmental DNA provides quantitative estimates of a threatened salmon species. Biological Conservation 237: 383-391.

Snyder MR, Stepien CA 2020. Increasing confidence for discerning species and population compositions from metabarcoding assays of environmental samples: case studies of fishes in the Laurentian Great Lakes and Wabash River. Metabarcoding and Metagenomics 4: e53455.

Stoeckle BC, Beggel S, Cerwenka AF, Motivans E, Kuehn R, Geist J 2017. A systematic approach to evaluate the influence of environmental conditions on eDNA detection success in aquatic ecosystems. PLOS One 12(12): e0189119.

Taberlet P, Coissac E, Hajibabaei M, Rieseberg LH 2012. Environmental DNA. Molecular Ecology 21: 1789-1793.

Team RC 2020. R: A language and environment for statistical computing. Vienna, Austria, R Foundation for Statistical Computing.

Thomas AC, Howard J, Nguyen PL, Seimon TA, Goldberg CS 2018. ANDe™: A fully integrated environmental DNA sampling system. Methods in Ecology and Evolution 9(6): 1379-1385.

Thomas AC, Nguyen PL, Howard J, Goldberg CS 2019. A self-preserving, partially biodegradable eDNA filter. Methods in Ecology and Evolution 10: 1136– 1141.

Thomsen PF, Møller PR, Sigsgaard EE, Knudsen SW, Jørgensen OA, Willerslev E 2016. Environmental DNA from Seawater Samples Correlate with Trawl Catches of Subarctic, Deepwater Fishes. PLOS ONE 11(11): e0165252.

Turner CR, Uy KL, Everhart RC 2015. Fish environmental DNA is more concentrated in aquatic sediments than surface water. Biological Conservation 183: 93-102.

Turner CR, Barnes MA, Xu CCY, Jones SE, Jerde CL, Lodge DM 2014. Particle size distribution and optimal capture of aqueous macrobial eDNA. Methods in Ecology and Evolution 5(7): 676-684.

US Fish and Wildlife Service Midwest Region 2019. Quality assurance project plan eDNA monitoring of bighead and silver carps. Bloomington, MN, USA. Pp. 161.

Valentini A, Taberlet P, Miaud C, Civade R, Herder J, Thomsen PF, Bellemain E, Besnard A, Coissac E, Boyer F, Gaboriaud C, Jean P, Poulet N, Roset N, Copp GH, Geniez P, Pont D, Argillier C, Baudoin J-M, Peroux T, Crivelli AJ, Olivier A, Acqueberge M, Le Brun M, Møller PR, Willerslev E, Dejean T 2016. Next-generation monitoring of aquatic biodiversity using environmental DNA metabarcoding. Molecular Ecology 25(4): 929-942.

Wang Q, Garrity GM, Tiedje JM, Cole JR 2007. Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. Applied and Environmental Microbiology 73(16): 5261-5267.

Wickham H 2011. The split-apply-combine strategy for data analysis. Journal of Statistical Software 40(1): 1-29.

Wickham H 2016. ggplot2: Elegant graphics for data analysis. New York, Springer-Verlag.

Wickham H, François R, Henry L, Müller K 2020. dplyr: A grammar of data manipulation. R package version 1.0.0. https://CRAN.R-project.org/package=dplyr.

Wickham H, Averick M, Bryan J, Chang W, McGowan LDA, François R, Grolemund G, Hayes A, Henry L, Hester J 2019. Welcome to the Tidyverse. Journal of Open Source Software 4(43): 1686.

Xie Y 2014. knitr: a comprehensive tool for reproducible research in R. Implement Reprod Res 1: 20.

Xie Y 2015. Dynamic Documents with R and knitr, CRC Press.

Xie Y 2020. Knitr: A general-purpose package for dynamic report generation in R (Version 1.29).

Yates MC, Glaser DM, Post JR, Cristescu ME, Fraser DJ, Derry AM 2020. The relationship between eDNA particle concentration and organism abundance in nature is strengthened by allometric scaling. Molecular Ecology. doi:10.1111/mec.15543.

Zhang S, Zhao J, Yao M 2020. A comprehensive and comparative evaluation of primers for metabarcoding eDNA from fish. Methods in Ecology and Evolution. doi.org/10.1111/2041-210X.13485

9. APPENDICES

Appendix 1. Standard operating procedure for fish eDNA collection

Site description data sheet

Table A1.1. Example datasheet highlighting key information that should be collected in conjunction with the eDNA sample.

Site name:		
Date		
Time		
Water temperature		
Turbidity		
DO %		
Sp. Conductivity		
Aquatic plants		
Substrate		
Reach average shade		
Average stream width		
Average stream depth		
N.o. fish taxa		
5-yr MCI score		

Using the Smith Root backpack to collect spot water samples (5 µm PES filter, not self-preserving)

The Smith Root backpack DNA sampler uses a pump that produces a pre-set maximum negative pressure. This feature prevents rupturing of the filter membranes that can occur if excess pressure is exerted by the pump.

1. Connect battery to electrodes

2. Switch on
3. Check settings = 1L/min, total 10 L, automatic, max 12 PSI
4. Connect tubing to instrument (tubing with red tape to ,IN, port)

5. Connect waste tubing (tubing with white tape to ,OUT, port)

6. Put on gloves; a new pair for each site
7. Connect tubing to filter housing (end with blue tape)

8. Lower nozzle into water without hitting the streambed
9. Filter until flow rate is less than 1 L/ min or 10 L are filtered
    - AVOID lifting the nozzle out of the water as air bubbles will prevent pump re-priming
    - AVOID poking the nozzle into the stream bed; sediment will block the filter
10. To stop filtering, lift nozzle up from the stream and invert, keep pump running for a several seconds to empty the line and to dry the filter
11. Leaving filter in the holder, crack the filter housing
12. Using forceps and nozzle to help, roll up filter and place filter in 5 mL yellow top eDNA tube
13. Label tube, store on ice in a poly bin

ANDe spot water samples (5 µm PES filter self-preserving)

Steps 1 through to 9 are the same as previous pages.

10. To stop filtering, lift nozzle up from the stream and invert, keep pump running for a several seconds to empty the line and to dry the filter (**IMPORTANT** it is necessary to dry the filter completely or the DNA may degrade)
11. Remove complete filter housing from boom, label filter housing, store in a poly bin

Trouble shooting

Issue I touch the screen, and nothing happens.
Resolution The screen is not a touch screen; use the buttons on the right-hand side.

Issue I switch the instrument on, and nothing happens.
Resolution Be patient; the screen takes a surprisingly long time to light up. Check the battery connections and fuses.

Issue The instrument was working but the screen has gone blank.
Resolution Occasionally the screen goes blank if it gets too hot in direct sunshine. Put the instrument in the shade.

Issue The instrument does not recognise the remote controller on the boom.
Resolution If the remote controller on the boom is first switched on too far away from the instrument, the connection will not be made. Move closer to the instrument and switch the remote control off and on again. Check the controller batteries.

There is a handy You Tube video https://www.youtube.com/watch?v=FdmuCkUU4cc.

Appendix 2. Extraction of fish eDNA from filters.

Equipment and consumables required

- 10% bleach solution prepared within four weeks.
- Plastic forceps.
- Scalpel blades.
- Heating block capable of heating up to 65^{o}C.
- Ethanol 100% ANALAR grade.
- Qiagen Blood and Tissue kit.
- Additional Qiagen AL and ATL buffers.
- Microcentrifuge tubes (1.5 mL).

Methods

- Remove filters from containers.
- Cut filters into quarters using sterile forceps and scalpel blades.
- Extract DNA from filters using the DNeasy Blood and Tissue (Qiagen, Hilden).
- Follow the manufacturer's instructions except increase the incubation temperature of the proteinase K step, and the volumes of Proteinase K, buffers ATL and AL, and ethanol.
- Incubate filters at 65 °C for 1 hour (Catalogue no. 19133) in Proteinase K (60 µL) and buffer ATL (560 µL). Vortex every 15 minutes.
- Follow the manufacturer's protocol except add 630 µL of buffer AL (Catalogue no. 19075) and 630 µL of ethanol (100%, Analar grade).
- Elute the DNA from the filter in 100 µL of buffer AE.
- Prepare 1:10 dilutions of the extracted DNA in AE buffer immediately after elution from the spin columns
- Store the DNA at -20oC until needed.

Appendix 3. Standard operating procedure for PCR amplification and library preparation

Equipment and consumables required

- Gloves.
- Bleach 10% solution. Diluted within four weeks.
- Calibrated pipettes for PCR set-up only (2-20 µL, 20-200 µL, 100-1000 µL).
- Calibrated pipettes for DNA template addition only (8-channel 0.5-10 µL).
- Calibrated pipettes for post-PCR only (8-channel 10-100 µL, 8-channel 0.5-10 µL).
- Filter tips (Eppendorf DNAse/RNAse free filter tips).
- Fine tip marker pen.
- 96-well PCR plate (Labcon XX).
- Plate sealing foil (XX).
- Plate sealer.
- 2 mL Eppendorf microcentrifuge tubes.
- MiFi 2X Mastermix (Bioline XX).
- DNA/RNA free water (XX).
- Mifish-UF-Illumina and Mifish-UR-Illumina primers (IDT, 10 µM working stock).
- DNA template (diluted to 1:10 in DNA/RNA free water into a 96-well PCR plate).
- Eppendorf PCR thermocycler.
- 6× loading buffer (XX).
- DNA ladder (XX).
- Agarose.
- RedSafe.
- TAE buffer.
- Gel electrophoresis machine.
- UV gel visualiser.
- SequalPrep Normalisation Plate kit (ThermoFisher XX).
- Vortex mixer.
- Mini-centrifuge (PCR set-up only).
- Centrifuge with adaptor for plates.

TAE buffer:

Tris base	4.844 g/l
Acetic acid	1.21 ml/l
EDTA disodium salt dihydrate	0.372 g/l

Methods

Pre-PCR setup

The reaction layout is pre-planned from the DNA dilution step into a 96-well plate. Clean and wipe the clean benches with bleach solution, followed by 70 % ethanol. Set up the reaction plates, tips, and Eppendorf tubes for the mastermix in the clean bench then turn on the UV sterilisation for 30 min in both the PCR set-up room and DNA template room.

Reagent volumes are calculated per plate (96 wells plus four reactions). Plates are run in triplicate.

Each reaction contains:
a. 10 µL of 2× MiFi Taq Mastermix
b. 1 µL of the Mifish-UF-illumina primer
c. 1 µL of the Mifish-UR-illumina primer
d. 6 µL of DNA/RNA free water
e. 2 µL of either template DNA or DNA/RNA free water

In the PCR set-up room:
Defrost the reagents on ice. Gently mix the mastermix by inverting the tube as it is defrosting. Briefly vortex and centrifuge primers.

In the DNA template room:
Defrost the DNA template on ice. Briefly vortex and centrifuge the DNA template.

In the PCR set-up room:
The mastermix for each plate is set up as follows:
Add the following to a 2 mL Eppendorf microcentrifuge tube (1 per plate, 3 total)
1000 µL of 2× MiFi mastermix
100 µL of the Mifish-UF-illumina primer
100 µL of the Mifish-UR-illumina primer
600 µL of DNA/RNA free water
Close the lid, mix by inverting several times and briefly centrifuge.

Pipette 18 µL of the mastermix into each well of a 96-well plate and cover with a plate-sealing foil.

In the DNA template room:
Bring the plates containing the reaction mastermix into the DNA template room. Add 2 µL of DNA template to the relevant well in each 96-well plate. Note: when the DNA template is diluted, include at least 2 wells distributed throughout the plate with only water. Cover with the plate-sealing foil.
Use a plate-sealer to seal the foil to the top of the plates. Ensure the plates are well sealed before continuing.

Centrifuge the plates briefly

Place each plate in the Eppendorf thermocycler and run a three-step PCR with the following parameters:
a. 5 min initial denaturation at 95 °C
b. 40 cycles of 30 sec at 95 °C, 30 sec at 55 °C and 45 sec at 72 °C
c. 5 min final elongation at 72 °C
d. Hold at 4 °C

Post-PCR processing

In the gel-room:
Make a 1% (weight/volume) agarose gel to visualise the PCR product. Add 1 µL of RedSafe per 10 mL of gel prior to pouring the gel into the gel-former. Add the gel comb and allow to set at room temperature for a minimum of 1 hour.

In the PCR-clean-up room:
Pool the reactions from the three replicate plates into a single plate.
Note, each plate had a 20 µL reaction volume, so to combine, transfer 20 µL of PCR product from the second and third plate into the corresponding well of the first plate. Approximately 60 µL should now be in each well.

In the gel room:
Remove the comb from the gel and rotate the gel-former within the electrophoresis tank. Add TAE buffer until it reaches the fill line indicated on the tank.
Using a pipette combine 5 µL of PCR product with 1 µL of loading dye by gently pipetting on a piece of parafilm. **Important:** Do not add loading dye to the PCR plate.
Load the first well in the gel with the DNA ladder then progressively load the remaining wells with the PCR product mixed with loading dye, recording which well on the gel corresponds with which well in the 96-well plate.
Run the gel to separate PCR product sizes. Visualise these using a UV transilluminator and check that all negative controls show no PCR product.

In the PCR clean-up room:
Transfer 20 µL of PCR product into the corresponding well of a SequalPrep Normalisation Plate and follow the kit instructions as follows

Add 20 µL of binding buffer and incubate at room temperature for 1 hour

Aspirate the liquid using a pipette and discard

Add 50 µL of wash buffer, agitate by pipetting then aspirate the buffer and discard

Remove any residual wash buffer using a 10 µL pipette

Add 20 µL of elution buffer to each well and incubate at room temperature for 5 min

Transfer 10 µL to a clean 96-well plate, add plate sealing foil and seal in the plate sealer.

Store at 4 °C until plate is couriered to Auckland Genomics for sequencing

Sequencing is conducted at Auckland Genomics with the following parameters:
1 x Amplicon QC
96 x Nextera Indexing
1 x Library QC
1 x 500 cycle MiSeq (2 x 250 bp paired end)

Appendix 4. Bioinformatic analysis.

Overview of the bioinformatic process

Raw sequence data from Illumina MiSeq runs must undergo bioinformatic processing to extract data for subsequent analysis. The appended R-markdown document is comprised of distinct code chunks comprising different stages of the data processing and analysis. Where user input is required, this is indicated in the text above that code chunk. User input will be required to define the file-paths. User input is also required to investigate read quality and make decisions on the value used during rarefaction.

- Load packages in R
- Set-up file paths and read in data
- Remove primers from the sequencing reads
- Filter the sequences to remove poor quality reads and trim the sequences
- Determine expected error profiles, dereplicate and de-noise the sequences
- Merge forward and reverse reads
- Size-select remaining reads and remove chimeras
- Assign the resulting amplicon sequence variants (ASVs) to a species by comparing the ASV to the reference taxonomy file
- Check the negative controls and subtract any reads present in them from the samples
- Filter samples to exclude non-fish taxonomic assignments
- Rarefy to an even sampling depth per sample
- Export both presence/absence tables and read numbers per species for rarefied and unrarefied data

Requirements for data processing

Raw sequence datafiles from BaseSpace (Illumina's data delivery platform)
Metadata spreadsheet for the files
The reference database as a *.fastA* file

A computer with the following

- RStudio
- dada2 (Callahan et al. 2016)
- ggplot2 (Wickham 2016)
- knitr (Xie 2014, 2015; Xie 2020)
- kableExtra (Zhu, 2020)
- phyloseq (McMurdie & Holmes 2013)
- tidyverse (Wickham et al. 2019)

- plyr (Wickham 2011)
- dplyr (Wickham et al. 2020)
- viridis (Garnier 2019)
- ranacapa (Kandlikar 2020)
- ShortRead (Morgan et al. 2009)
- Biostrings (Pagès et al. 2020)
- Vegan (Oksanen et al. 2019)
- cutadapt (note, cutadapt is installed outside of R, but is called through R using the code provided) (Martin 2011)

Appendix 5. Bioinformatic pipeline for MiFish primer set

Load libraries

Libraries are the packages that contain the functions used for the bioinformatic analysis. The following code loads these into the R environment.

```
library(dada2)
library(ggplot2)
library(knitr)
library(kableExtra)
library(phyloseq)
library(tidyverse)
library(plyr)
library(dplyr)
library(viridis)
library(ranacapa)
library(ShortRead)
library(Biostrings)
library(vegan)
library(remotes)
```

Pre-processing

File-paths should be defined at the start of the session, which reduces the need for hard-coding of these throughout the code. The advantage of this is that if the code is run on a different computer or file-paths need to be changed, it minimises the amount of code requiring modification.

```
path1 <- Sys.glob("C:/Users/laura.kelly/Desktop/Fish_eDNA/AG0246-42-Mifish/*")  ## CHANGE ME to the directory containing the fastq files.

path <- "C:/Users/laura.kelly/Desktop/Fish_eDNA/Mifish_validation"

list.files(path1)

fnFs <- sort(list.files(path1, pattern = "L001_R1_001.fastq.gz", full.names = TRUE))
fnRs <- sort(list.files(path1, pattern = "L001_R2_001.fastq.gz", full.names = TRUE))
```

Primer sequences are defined at the start of the session so they can be found and trimmed from the sequences using cutadapt.

```
FWD <- "GTCGGTAAAACTCGTGCCAG"
REV <- "CATAGTGGGGTATCTAATCCCA"
```

Vectors are made with all the possible orientations of both the forward and reverse primers.

```
allOrients <- function(primer) {
    # Create all orientations of the input sequence
    require(Biostrings)
    dna <- DNAString(primer)  # The Biostrings works w/ DNAString obj
ects rather than character vectors
    orients <- c(Forward = dna, Complement = complement(dna), Reverse
= reverse(dna),
        RevComp = reverseComplement(dna))
    return(sapply(orients, toString))  # Convert back to character ve
ctor
}
FWD.orients <- allOrients(FWD)
REV.orients <- allOrients(REV)
FWD.orients
```

The following step is a check step. This calculates the number of reads where the forward and reverse primers are found from one sample. If the number of hits is very low, this may indicate a sequencing problem or an issue with the user-defined primer sequences.

```
primerHits <- function(primer, fn) {
    # Counts number of reads in which the primer is found
    nhits <- vcountPattern(primer, sread(readFastq(fn)), fixed = FALS
E)
    return(sum(nhits > 0))
}
rbind(FWD.ForwardReads = sapply(FWD.orients, primerHits, fn = fnFs[[1
]]),
    REV.ReverseReads = sapply(REV.orients, primerHits, fn = fnRs[[1]]
))
```

Use cutadapt to trim primers

Cutadapt removes the primer sequences from the ends of the sequences. Sequences where the primers are not found are discarded from the output.

```
cutadapt <- "C:/Users/laura.kelly/AppData/Local/Continuum/miniconda3/
envs/cutadapt/Scripts/cutadapt" # CHANGE ME to the cutadapt path on y
our machine
system2(cutadapt, args = "--version") # Run shell commands from R
```

```
path.cut <- file.path(path, "cutadapt")

if(!dir.exists(path.cut)) dir.create(path.cut)

fnFs.cut <- file.path(path.cut, basename(fnFs))
fnRs.cut <- file.path(path.cut, basename(fnRs))
```

```
# Trim FWD off of R1 (forward reads) -
R1.flags <- paste0("-g", " ^", FWD)
# Trim REV off of R2 (reverse reads)
R2.flags <- paste0("-G", " ^", REV)


for(i in seq_along(fnFs)) {
  system2(cutadapt, args = c(R1.flags, R2.flags, "-e", 0.05,
                             "--discard-untrimmed",
                             "-o", fnFs.cut[i], "-p", fnRs.cut[i],
                             fnFs[i], fnRs[i]))
}
```

The code below is a check step to ensure that no primers are left on the reads. Occasionally some primers are present due to internal primer hits (primer sequences in the middle of the sequences) this is okay as these are removed later in the process. This step also checks that the same number of forward and reverse reads remain after cutadapt has processed them. If they are different an error message will be generated for the user to check the processing steps.

```
path.cut <- file.path(path, "cutadapt")

if(!dir.exists(path.cut)) dir.create(path.cut)

fnFs.cut <- file.path(path.cut, basename(fnFs))
fnRs.cut <- file.path(path.cut, basename(fnRs))

cutFs <- sort(list.files(path.cut, pattern = "R1_001.fastq.gz", full.
names = TRUE))
cutRs <- sort(list.files(path.cut, pattern = "R2_001.fastq.gz", full.
names = TRUE))

if(length(cutFs) == length(cutRs)) print("Forward and reverse files m
atch. Go forth and find nemo")
if (length(cutFs) != length(cutRs)) stop("Forward and reverse files d
o not match. Go back and have a check")
```

Extract sample names

Extract the sample names and split the sequences into forward and reverse files, which makes downstream processing easier.

```
# Extract sample names, assuming filenames have format:
get.sample.name <- function(fname) strsplit(basename(fname), "_")[[1]
][1]
sample.names <- unname(sapply(cutFs, get.sample.name))
head(sample.names)
```

```
# Split files into forward and reverse to make it easier later
path.cut.F <- file.path(path, "cutadapt", "forward")
path.cut.R <- file.path(path, "cutadapt", "reverse")

if(!dir.exists(path.cut.F)) dir.create(path.cut.F)
if(!dir.exists(path.cut.R)) dir.create(path.cut.R)

file.copy(list.files(path1, pattern = "L001_R1_001.fastq.gz", full.na
mes = TRUE), path.cut.F)
file.copy(list.files(path1, pattern = "L001_R2_001.fastq.gz", full.na
mes = TRUE), path.cut.R)
```

Quality plots

This step is critical. The read quality profiles are plotted (initially for a subset of the reads). If there are fewer than 20 samples plot all of the samples. If there are more than 20, then plot a random subset of 20. This code requires user input to change which option is used at the end (by modifying which lines have # in the front). The quality plots should show high quality across the read, with a reduction at the end. If the quality plot shows the quality score dropping early in the read or the quality being low throughout, this indicates poor quality sequences. This can occur in some samples, but if it occurs in more than one of the 20 plots, it is worth investigating the other samples as this may indicate a poor sequencing run.

```
if(length(cutFs) <= 20) {
  forplots <- plotQualityProfile(cutFs) +
  scale_x_continuous(breaks=seq(0,250,10)) +
  scale_y_continuous(breaks=seq(0,40,2)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
  revplots <- plotQualityProfile(cutRs) +
  scale_x_continuous(breaks=seq(0,250,10)) +
  scale_y_continuous(breaks=seq(0,40,2)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
} else {
  rand_samples <- sample(size = 20, 1:length(cutFs)) # grab 20 random
samples to plot
  fwd_qual_plots <- plotQualityProfile(cutFs[rand_samples]) +
  scale_x_continuous(breaks=seq(0,250,10)) +
  scale_y_continuous(breaks=seq(0,40,2)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
  rev_qual_plots <- plotQualityProfile(cutRs[rand_samples]) +
  scale_x_continuous(breaks=seq(0,250,10)) +
  scale_y_continuous(breaks=seq(0,40,2)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
}

#forplots   # use this if there are fewer than 20 samples
#revplots   # use this if there are fewer than 20 samples
```

```
fwd_qual_plots # use this if there are more than 20 samples
rev_qual_plots # use this if there are more than 20 samples
```

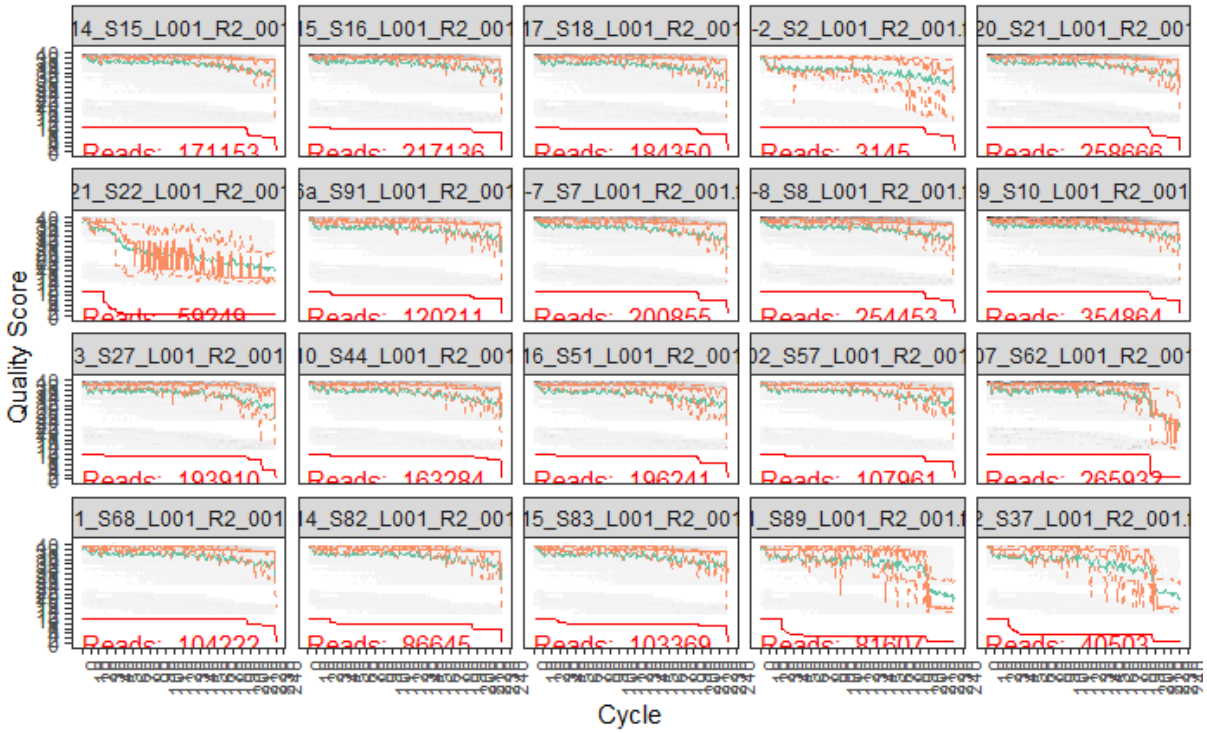


Figure A5.1. Example quality plots. Notice that sample 6 (first column, second row) shows reduced quality early in the cycling profile indicating poor sequence quality.

Filter and trim the sequences

There are a number of options in the below code that may need to be changed.

Commands

fwd	file path of where to find the forward reads
filt	file path of where to put the filtered forward reads
rev	file path of where to find the reverse reads
filt.rev	file path of where to put the filtered reverse reads
truncLen	length of which to truncate the reads. The first number refers to the forward read and the second to the reverse. This is user and dataset variable. There is a trade-off between quality and how much you trim. The more you trim the better quality the final read will have. However,

the forward and reverse reads still need to overlap so you can not't trim too much.

maxEE — after truncation this is the maximum number of ,expected errors, allowed before a read is discarded. Expected errors are calculated as EE = sum(10^(-Q/10)) - based on the quality scores of the reads. In general the reverse reads generally have a lower quality and should be allowed more expected errors. If, however, your plots indicate that over the read length the quality remains high, this can be left the same for both forward and reverse reads.

truncQ — 2 is a special quality score from Illumina denoting the start of a bad read. There is little effect of using 2 if reads are truncated. You can set it to another score but will truncate the read on the first appearance of the score which will likely end up in reads not overlapping

maxN — How many ambiguous bases are allowed. Best to keep this at 0

rm.phix — Do you want to remove phiX from the samples. Phi X is a bacteriophage genome that is spiked into samples run on Illumina machines as a quality control and to aid in mitigating issues from low diversity (amplicon) libraries. It is normally removed when samples are processed from the machine and also shouldn't get through the cutadapt stage but this option can be kept at TRUE to remove any that have somehow made it through the earlier processing stages.

```
pathF <- path.cut.F
pathR <- path.cut.R

filtpathF <- file.path(pathF, "filtered")
filtpathR <- file.path(pathR, "filtered")


fastqFs <- sort(list.files(pathF, pattern="fastq"))
fastqRs <- sort(list.files(pathR, pattern="fastq"))
if(length(fastqFs) != length(fastqRs)) stop("There's a problem. Go ba
ck and check files")

out <- filterAndTrim(fwd=file.path(pathF, fastqFs), filt=file.path(fi
ltpathF, fastqFs),
                     rev=file.path(pathR, fastqRs), filt.rev=file.
path(filtpathR, fastqRs),
                      truncLen=c(150,150), maxEE=c(2,4), truncQ=2,
maxN=0, rm.phix=TRUE,
                     compress=TRUE, verbose=TRUE, multithread=TRUE)
kable(out) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

The output table below summarises the reads in and reads out of the filter and trim process. Check to make sure not too many reads are being lost at this stage as it could indicate a problem with the data.

	reads.in	reads.out
200127-1_S1_L001_R1_001.fastq.gz	1280	947
200127-10_S11_L001_R1_001.fastq.gz	176946	159584
200127-10a_S95_L001_R1_001.fastq.gz	174904	152449
200127-11_S12_L001_R1_001.fastq.gz	240674	204282
200127-11a_S96_L001_R1_001.fastq.gz	21115	15374
200127-12_S13_L001_R1_001.fastq.gz	129319	117272
200127-13_S14_L001_R1_001.fastq.gz	182726	162385
200127-14_S15_L001_R1_001.fastq.gz	175367	161956
200127-15_S16_L001_R1_001.fastq.gz	223404	197156
200127-16_S17_L001_R1_001.fastq.gz	132148	121001
200127-17_S18_L001_R1_001.fastq.gz	188823	168926
200127-18_S19_L001_R1_001.fastq.gz	147102	134875
200127-19_S20_L001_R1_001.fastq.gz	28365	25942
200127-1a_S85_L001_R1_001.fastq.gz	103215	92296

Figure A5.2. Example output table that is generated to show the number of reads before and after quality filtering and trimming of the sequences.

Check outputs

This is another data processing step, both to check outputs and to manipulate the files for easier processing downstream. The code lists the filtered fastq files, removes the fastq.gz part of the filenames (this is a compressing option) and stores it in samples names, checks the names of the forward and reverse files match and renames the filtered output objects.

```
filtFs <- list.files(filtpathF, pattern="fastq", full.names = TRUE)
filtRs <- list.files(filtpathR, pattern="fastq", full.names = TRUE)
sample.names <- sapply(strsplit(basename(filtFs), "_"), `[`, 1) # Ass
umes filename = samplename_XXX.fastq.gz
sample.namesR <- sapply(strsplit(basename(filtRs), "_"), `[`, 1) # As
sumes filename = samplename_XXX.fastq.gz
if(!identical(sample.names, sample.namesR)) stop("Forward and reverse
```

```
files do not match - go back and check files")
names(filtFs) <- sample.names
names(filtRs) <- sample.namesR
```

Error profiles

This code lets dada2 learn the error profiles of the sequences. To correct sequences, dada2 uses a parametric error model which is specific to each data set. It achieves this model by using a machine learning technique which alternates between the estimation of error rates and inferring the sample composition until these two values converge.

We use 1×10^8 bases to calculate the error profile. The higher the number of bases used the more accurate the error profile will be, but it is a trade-off with speed.

```
errF <- learnErrors(filtFs, nbases = 1e8, MAX_CONSIST = 15, multithread=TRUE, verbose = TRUE)
errR <- learnErrors(filtRs, nbases = 1e8, MAX_CONSIST = 15, multithread=TRUE, verbose = TRUE)
```

Check that the error profile is sensible. The red line shows the error under the nominal definition of the Q-score. Observed points (dots) should not deviate far from the black line (estimated error rates).

There should be a general negative trend between error frequency and quality.

```
errors_f <- plotErrors(errF, nominalQ=TRUE)
errors_f

errors_r <- plotErrors(errR, nominalQ=TRUE)
errors_r
```

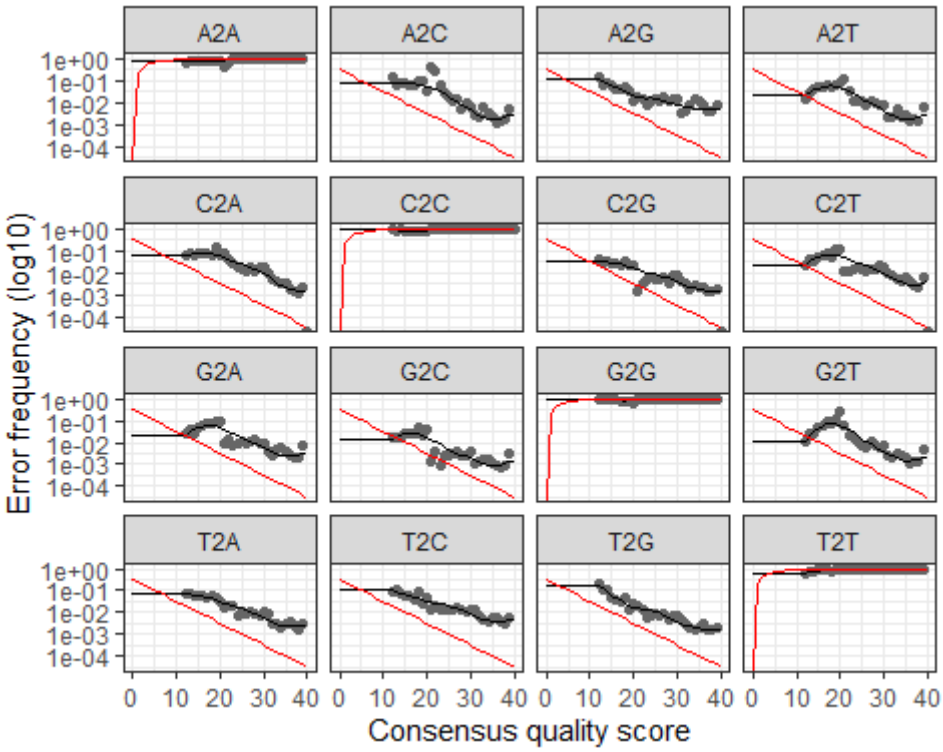


Figure A5.3. An example of the error profile plots generated by dada2. Note the red line is the nominal error based on the quality score, the black line is the dada2 error profile and the black dots are the observed points.

Dereplicate the sequences

Dereplicating combines the identical sequencing reads into “unique sequences”.

This process gives an abundance value to the unique sequences equal to that of the number of reads with that sequence. Quality information for the unique sequences is kept as the average of that of all reads combined to form the unique sequence.

This step reduces the computational power required for future steps.

```
derepF <- derepFastq(filtFs, verbose=TRUE)
derepR <- derepFastq(filtRs, verbose=TRUE)

dadaF.pseudo <- dada(derepF, err=errF, multithread=TRUE)
dadaR.pseudo <- dada(derepR, err=errR, multithread=TRUE)
```

Merge reads and make a sequence table

Merge the forward and reverse reads and produce a sequence table.

Inputs are the results from the inference step (*dadaF.pseudo* and *dadaR.pseudo*) as well as the dereplicated sequences (*derepF* and *derepR*).

maxMismatch number of mismatches allowed in your overlap region.

minOverlap the minimum length you want the sequences to overlap by. This is a factor of how long your amplicon is and how much trimming you did earlier. The longer the overlap the higher confidence you have of the sequence being of good quality, however setting this value too high will result in a failure of sequences to merge.

```
mergers <- mergePairs(dadaF.pseudo, derepF, dadaR.pseudo, derepR, max
Mismatch = 1, minOverlap = 35, verbose=TRUE)
seqtab <- makeSequenceTable(mergers)
saveRDS(seqtab, "C:/Users/laura.kelly/Desktop/Fish_eDNA/Mifish_valida
tion/mifish_eDNA_trunc.rds")
```

Chimera removal and output checks

Check sequence lengths and run a chimera check.

The approximate size of the sequences that should be produced with a primer set are known. The filtering step removes sequences that are considerably longer or shorter than the expected size range as these are likely to be artefacts of the PCR or sequencing processes.

Following the size-filtering step, chimeras need to be removed. Chimeras are sequences that are made up of the DNA of two (or more) species. Sometimes during PCR, the enzymes end up copying the sequence of half of one DNA fragment and half of another. These need to be removed for the following analysis steps.

```
seqtab<-readRDS("C:/Users/laura.kelly/Desktop/Fish_eDNA/Mifish_valida
tion/mifish_eDNA_trunc.rds")

trimtable <- as.data.frame(table(nchar(getSequences(seqtab))))
colnames(trimtable) <- c("Length (bp)", "Frequency")
kable(trimtable)

seqtab2 <- seqtab[,nchar(colnames(seqtab)) %in% seq(150,350)]
seqtab.nochim <- removeBimeraDenovo(seqtab2, multithread=TRUE, verbos
e=TRUE)
saveRDS(seqtab.nochim, "C:/Users/laura.kelly/Desktop/Fish_eDNA/Mifish
_validation/mifish_nochim_eDNA_trunc.rds")
```

Check losses

Produce a table which shows the number of reads at each stage. Most reads are likely to be lost at the trimming stage. Check to see that the merging step does not result in very few reads (this could be a sign of trimming the sequences too short).

To identify the potential losses throughout the filtering steps, the number of sequences at each step is reported in a table.

input	is the number of raw sequence reads from Illumina after the primers have been trimmed from the sequences. Note that any reads lost during the primer trimming step are not included.
filtered	is the number of sequences that remain following read-quality filtering.
denoisedF	is the number of sequences remaining in the forward reads following error profiling.
denoisedR	is the number of sequences remaining in the reverse reads following error profiling.
merged	is the number of sequences that successfully merged. Note a large loss of sequence numbers at this step indicates that truncation of sequences in the initial quality filtering steps is too harsh, meaning the sequences are not long enough to merge.
nochim	is the number of sequences remaining after chimera removal. It is normal to have a reasonable number of sequences removed during this step.

```
getN <- function(x) sum(getUniques(x))
track <- cbind(out, sapply(dadaF.pseudo, getN), sapply(dadaR.pseudo,
getN), sapply(mergers, getN), rowSums(seqtab.nochim))
colnames(track) <- c("input", "filtered", "denoisedF", "denoisedR", "
merged", "nonchim")
rownames(track) <- sample.names

kable(track) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

Assign taxonomy

The following code chunks allow the assignment of taxonomy to the sequences from the steps above. The metadata file, reference database file and sequences are required for the following steps. Note that the formatting of the metadata file is very important, and the sample names of the metadata file and the sequence file must match in order for the following code to successfully run.

Define the location of the reference database.

```
# database location
fishDB <- "C:/Users/laura.kelly/Desktop/Fish_eDNA/Fish_12S_20200629_n
ogaps.fasta"
```

Set up input and output folders, load the datafile and set-up the output prefixes.

```
username <- "Mifish_validation"
input <- paste0("C:/Users/laura.kelly/Desktop/Fish_eDNA/", username,
"/input/")
output <- paste0("C:/Users/laura.kelly/Desktop/Fish_eDNA/", username,
"/output/")

#Load Datafile
sq <- getSequences(readRDS("C:/Users/laura.kelly/Desktop/Fish_eDNA/Mi
fish_validation/mifish_nochim_eDNA_trunc.rds"))
seqtab.nochim<-readRDS("C:/Users/laura.kelly/Desktop/Fish_eDNA/Mifish
_validation/mifish_nochim_eDNA_trunc.rds")

# Setup Output prefix for slice files (rds)
file_name_prefix <- paste0(output, 'mifish.nochimera.tax.slice_')
```

Taxonomy assignment can be a memory-intensive process. To reduce the chance of running out of memory, especially with larger datasets, it is possible to assign the taxonomy for the sequences in "chunks". These are saved as separate files and then merged at the end to produce a single file containing the taxonomy assignments for all of the sequences.

Note: There is a parameter in this code called *minBoot* which impacts the stringency of taxonomic assignments. The taxonomy is assigned by determining the kmer profile of the sequence and then matching this with the kmer profiles of the sequences in the reference database. The *minBoot* designates the number of times out of 100 the same taxonomic assignment must be made for that taxonomic assignment to be designated in the output. The default setting is 50, however, for sequences over 250 bases the recommended threshold is 80.

The following code runs as chunks. Set the chunk sizes, break dataset down into chunks and assign taxonomy for each chunk.
IMPORTANT: change the stringency of taxonomic assignment and other settings below *before* running chunk. Note that there are two parts of the loop that need to be changed.

```
# Set chunksize
CHUNKSIZE = 10000

# Calculate number of slices and remainder
NUM_SQ = length(sq)
NUM_SLICES <- as.integer(NUM_SQ/CHUNKSIZE)
LEN_REMAINDER <- as.integer(NUM_SQ%%CHUNKSIZE)

# Compute full slices
idx <- 0
```

```
while (idx < NUM_SLICES)
{
  start_idx = (idx*CHUNKSIZE)+1
  end_idx =   (idx+1)*CHUNKSIZE
  fname = paste0(file_name_prefix, start_idx, '_', end_idx, '.rds')

  result_slice <- assignTaxonomy(sq[start_idx:end_idx], fishDB, minBo
ot = 80,multithread=TRUE)
  saveRDS(result_slice, fname)
  idx <- idx+1
}
# Compute remainder if present
if (LEN_REMAINDER!=0){
  start_idx = (NUM_SLICES*CHUNKSIZE)+1
  end_idx =   NUM_SQ
  fname = paste0(file_name_prefix, start_idx, '_', end_idx, '.rds')

  result_slice <- assignTaxonomy(sq[start_idx:end_idx], fishDB, minBo
ot = 80, multithread=TRUE)
  saveRDS(result_slice, fname)
}
```

If there are multiple taxonomy slices the files need to be combined. Transfer all *.rds files to local machine if run on HPC or other computer cluster.
Run combine on desktop Computer e.g.:
IMPORTANT: change the file locations below *before* running chunk

```
setwd(output)

mifish_taxa<-readRDS("C:/Users/laura.kelly/Desktop/Fish_eDNA/Mifish_v
alidation/output/mifish.nochimera.tax.slice_1_636.rds")


colnames(mifish_taxa)
```

Create a phyloseq object

Phyloseq is a package that enables the manipulation of next generation sequencing data. There are three components that make up the *mifish.physeq* phyloseq object:

otu_table is the table of unique sequences from the study. This is the list of sequences with chimeras removed. This is called *mifish.nochim.rds* in the code chunk below.\

tax_table is the taxonomy table generated from the assign taxonomy section above. Note this requires all of the slices that were generated to be bound together as in the code above. This is defined as *mifish_taxa* in the current code chunk.\

sample_data is the metadata file from the study containing all of the other information about the samples including variables such as site name, region, volume filtered, and filter type used. Any information of interest to the study can be recorded in the metadata file but this must be recorded in separate columns for each variable of interest.

```
metadata<-read.csv("C:/Users/laura.kelly/Desktop/Fish_eDNA/metadata_validation.csv")
metadata=metadata%>%mutate(sample_name=as.factor(sample_name))
rownames(metadata) = metadata$sample_name


mifish.physeq <- phyloseq(otu_table(seqtab.nochim, taxa_are_rows = FALSE),
                          tax_table(mifish_taxa),
                          sample_data(metadata))
```

Clean the data

Subtract amplicon sequence variants (ASVs) from negative controls.

Throughout the field collection, DNA extraction, PCR and sequencing process, negative control samples are included to ensure that any contamination is identifiable. How contamination is managed depends on the type and degree of any contamination of the negative controls. Ideally any significant contamination prior to sequencing will be identified at the PCR stage (using visualisation on an agarose gel) and samples re-run through the PCR steps.

Sequences in the negative controls can still occur. The following code section takes the various controls (DNA extraction, PCR, and sequencing controls) and determines the numbers of each unique sequence in them. It is assumed that in a "worst case" scenario the same number of contaminating sequences will be present in all of the samples, thus the number of each of these unique sequences is subtracted from the samples below.

For DNA extraction controls, the relevant sequences are subtracted only from samples that belong to the same DNA extraction batch. For PCR and sequencing controls, these are subtracted from all of the samples they are relevant to (i.e. the same PCR run or sequencing run).

Note, the code below needs to be modified for each sequencing run to account for the setup of controls across the plate. Set up a template to keep this consistent, which will reduce the need to modify this code.

```
Controls = subset_samples(mifish.physeq , Type == "control")
Controls = filter_taxa(Controls, function(x) sum(x) > 0, TRUE)
```

```
sample_sums(Controls)

mifish.physeq <- phyloseq(otu_table(seqtab.nochim, taxa_are_rows = FA
LSE),
                          tax_table(mifish_taxa),
                          sample_data(metadata))

## __subset phyloseq project by batches (extraction controls) ####

Batch1.chordata = subset_samples(mifish.physeq, Batch=="batch1")
Batch2.chordata = subset_samples(mifish.physeq, Batch=="batch2")
Batch3.chordata = subset_samples(mifish.physeq, Batch=="batch3")
Batch4.chordata = subset_samples(mifish.physeq, Batch=="batch4")
Batch5.chordata = subset_samples(mifish.physeq, Batch=="batch5")
Batch6.chordata = subset_samples(mifish.physeq, Batch=="batch6")
Batch12.chordata = subset_samples(mifish.physeq, Batch =="batch7")

## __subset the negative controls (not all batches contain a neg ctrl
) ####

Batch1.chordata_neg = subset_samples(Batch1.chordata, Type == "contro
l"|Type=="field")
Batch2.chordata_neg = subset_samples(Batch2.chordata, Type == "contro
l"|Type=="field")
Batch4.chordata_neg = subset_samples(Batch4.chordata, Type == "contro
l"|Type=="field")
Batch5.chordata_neg = subset_samples(Batch5.chordata, Type == "contro
l"|Type=="field")
Batch6.chordata_neg = subset_samples(Batch6.chordata, Type == "contro
l"|Type=="field")
Batch12.chordata_neg = subset_samples(Batch12.chordata, Type == "cont
rol"|Type=="field")


## calculate column sums ####

Batch1.chordata_neg_sums <- colSums(otu_table(Batch1.chordata_neg))
Batch2.chordata_neg_sums <- colSums(otu_table(Batch2.chordata_neg))
Batch4.chordata_neg_sums <- colSums(otu_table(Batch4.chordata_neg))
Batch5.chordata_neg_sums <- colSums(otu_table(Batch5.chordata_neg))
Batch6.chordata_neg_sums <- colSums(otu_table(Batch6.chordata_neg))
Batch12.chordata_neg_sums <- colSums(otu_table(Batch12.chordata_neg))

## find the max value of the controls ####

Batch1.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch1.chordata_neg)))), 1, max)
Batch2.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch2.chordata_neg)))), 1, max)
Batch4.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch4.chordata_neg)))), 1, max)
Batch5.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
```

```
Batch5.chordata_neg)))), 1, max)
Batch6.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch6.chordata_neg)))), 1, max)
Batch12.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table
(Batch12.chordata_neg)))), 1, max)


## move the max value to a vector so it can be subtracted ####

Batch1.chordata_neg_sums_vec <- as.vector(Batch1.chordata_neg_max)
Batch2.chordata_neg_sums_vec <- as.vector(Batch2.chordata_neg_max)
Batch4.chordata_neg_sums_vec <- as.vector(Batch4.chordata_neg_max)
Batch5.chordata_neg_sums_vec <- as.vector(Batch5.chordata_neg_max)
Batch6.chordata_neg_sums_vec <- as.vector(Batch6.chordata_neg_max)
Batch12.chordata_neg_sums_vec <- as.vector(Batch12.chordata_neg_max)

## move the sums to a vector so they can be subtracted ####

Batch1.chordata_neg_sums_vec <- as.vector(Batch1.chordata_neg_sums)
Batch2.chordata_neg_sums_vec <- as.vector(Batch2.chordata_neg_sums)
Batch4.chordata_neg_sums_vec <- as.vector(Batch4.chordata_neg_sums)
Batch5.chordata_neg_sums_vec <- as.vector(Batch5.chordata_neg_sums)
Batch6.chordata_neg_sums_vec <- as.vector(Batch6.chordata_neg_sums)
Batch12.chordata_neg_sums_vec <- as.vector(Batch12.chordata_neg_sums)

## make ASV table into a dataframe so to be able to subtract ####


B1.chordata = as(otu_table(Batch1.chordata), "matrix")
B1df.chordata = as.data.frame(B1.chordata)
B2.chordata = as(otu_table(Batch2.chordata), "matrix")
B2df.chordata = as.data.frame(B2.chordata)
B3.chordata = as(otu_table(Batch3.chordata), "matrix")
B3df.chordata = as.data.frame(B3.chordata)
B4.chordata = as(otu_table(Batch4.chordata), "matrix")
B4df.chordata = as.data.frame(B4.chordata)
B5.chordata = as(otu_table(Batch5.chordata), "matrix")
B5df.chordata = as.data.frame(B5.chordata)
B6.chordata = as(otu_table(Batch6.chordata), "matrix")
B6df.chordata = as.data.frame(B6.chordata)


## do the subtraction ####
## __note batch 10 and 12 subtraction should be applied to all batche
s (global negative control) ####

B1df.chordata[,1:length(B1df.chordata)] <- sweep(B1df.chordata[,1:len
gth(B1df.chordata)],2,Batch1.chordata_neg_sums_vec)
B2df.chordata[,1:length(B2df.chordata)] <- sweep(B2df.chordata[,1:len
gth(B2df.chordata)],2,Batch2.chordata_neg_sums_vec)
B4df.chordata[,1:length(B4df.chordata)] <- sweep(B4df.chordata[,1:len
gth(B4df.chordata)],2,Batch4.chordata_neg_sums_vec)
```

```
B5df.chordata[,1:length(B5df.chordata)] <- sweep(B5df.chordata[,1:len
gth(B5df.chordata)],2,Batch5.chordata_neg_sums_vec)
B6df.chordata[,1:length(B6df.chordata)] <- sweep(B6df.chordata[,1:len
gth(B6df.chordata)],2,Batch6.chordata_neg_sums_vec)


B1df.chordata[,1:length(B1df.chordata)] <- sweep(B1df.chordata[,1:len
gth(B1df.chordata)],2,Batch12.chordata_neg_sums_vec)
B2df.chordata[,1:length(B2df.chordata)] <- sweep(B2df.chordata[,1:len
gth(B2df.chordata)],2,Batch12.chordata_neg_sums_vec)
B3df.chordata[,1:length(B3df.chordata)] <- sweep(B3df.chordata[,1:len
gth(B3df.chordata)],2,Batch12.chordata_neg_sums_vec)
B4df.chordata[,1:length(B4df.chordata)] <- sweep(B4df.chordata[,1:len
gth(B4df.chordata)],2,Batch12.chordata_neg_sums_vec)
B5df.chordata[,1:length(B5df.chordata)] <- sweep(B5df.chordata[,1:len
gth(B5df.chordata)],2,Batch12.chordata_neg_sums_vec)
B6df.chordata[,1:length(B6df.chordata)] <- sweep(B6df.chordata[,1:len
gth(B6df.chordata)],2,Batch12.chordata_neg_sums_vec)


## replace the values less than zero with zero ####

B1df.chordata <- replace(B1df.chordata, B1df.chordata < 0, 0)
B2df.chordata <- replace(B2df.chordata, B2df.chordata < 0, 0)
B3df.chordata <- replace(B3df.chordata, B3df.chordata < 0, 0)
B4df.chordata <- replace(B4df.chordata, B4df.chordata < 0, 0)
B5df.chordata <- replace(B5df.chordata, B5df.chordata < 0, 0)
B6df.chordata <- replace(B6df.chordata, B6df.chordata < 0, 0)


## merge ASV tables and check the dimensions ####

cleaned.chordata <- rbind(B1df.chordata, B2df.chordata, B3df.chordata
, B4df.chordata, B5df.chordata, B6df.chordata)
dim(cleaned.chordata) # should rows and columns of match what went in
```

Create a new phyloseq object

Using the cleaned data from the previous steps, create a new phyloseq object. Then extract the taxonomy file and clean up the NA and unclassified taxonomy labels so the taxonomy will show the last taxonomic level that was positively identified (e.g. Galaxias) and "unassigned" for any downstream taxonomic levels.

The taxonomy assignments use "NA" as the default when a taxonomic level cannot be assigned. The following script tidies the data to follow a convention that if a sequence cannot be assigned at a taxonomic level (i.e. Phylum, Class, Order, Family, Genus or Species), the code will go to the highest taxonomic level that was assigned and input "unclassified" following this. This means that species assigned to genus will be identifiable as *Genus unclassified* rather than simply *NA* in further analyses.

```
mifish.physeq.cleaned <- phyloseq(otu_table(cleaned.chordata, taxa_ar
e_rows = FALSE),
                                  sample_data(metadata),
                                  tax_table(mifish_taxa))

mifish.physeq.cleaned <- subset_samples(mifish.physeq.cleaned, Type !
= "control")

tax.clean <- data.frame(tax_table(mifish.physeq.cleaned))
for (i in 1:7){ tax.clean[,i] <- as.character(tax.clean[,i])}
tax.clean[is.na(tax.clean)] <- ""
for (i in 1:nrow(tax.clean)){
  if (tax.clean[i,2] == ""){
    kingdom <- paste(tax.clean[i,1],".unclassified",  sep = "")
    tax.clean[i, 2:7] <- kingdom
  } else if (tax.clean[i,3] == ""){
    phylum <- paste(tax.clean[i,2], ".unclassified", sep = "")
    tax.clean[i, 3:7] <- phylum
  } else if (tax.clean[i,4] == ""){
    class <- paste(tax.clean[i,3], ".unclassified", sep = "")
    tax.clean[i, 4:7] <- class
  } else if (tax.clean[i,5] == ""){
    order <- paste(tax.clean[i,4], ".unclassified", sep = "")
    tax.clean[i, 5:7] <- order
  } else if (tax.clean[i,6] == ""){
    family <- paste(tax.clean[i,5], ".unclassified", sep = "")
    tax.clean[i, 6:7] <- family
  } else if (tax.clean[i,7] == ""){
    tax.clean$Species[i]<-paste(tax.clean[i,6],".unclassified", sep =
"")
  }
}

mifish_chordata1<-as.matrix(tax.clean)

mifish.chordata.physeq.cleaned <- phyloseq(otu_table(cleaned.chordata
, taxa_are_rows = FALSE),
                                           sample_data(metadata),
                                           tax_table(mifish_chordata1
))

original.chordata.physeq <- as.data.frame(sample_sums(mifish.chordata
.physeq.cleaned))
original.chordata.physeq$Names <- rownames(original.chordata.physeq)

cleaned.chordata.physeq <- as.data.frame(sample_sums(mifish.chordata.
physeq.cleaned))
cleaned.chordata.physeq$Names <- rownames(cleaned.chordata.physeq)

control.chordata.rem.sums <- dplyr::left_join(original.chordata.physe
q,cleaned.chordata.physeq, by="Names")
```

```
colnames(control.chordata.rem.sums) <- c("Original","Names","New")
control.chordata.rem.sums <- control.chordata.rem.sums %>%
  mutate(perc = New/Original*100)
```

Subset the data into fish (plus lamprey) only

```
Fish.chordata = subset_taxa(mifish.chordata.physeq.cleaned, Class=="A
ctinopterygii"|Class=="Petromyzontida")
Other.chordata = subset_taxa(mifish.chordata.physeq.cleaned, Class!="
Actinopterygii"|Class!="Petromyzontida")
```

Create data tables for unrarefied data

This code chunk converts read numbers into presence (1) or absence (0) on the un-rarefied data. Samples with less than 100 reads are removed as they may be the result of index hopping or contamination from neighbouring samples.

The taxonomy is conglomerated to merge sequences belonging to the same taxa for ease of presentation. There is a code chunk below that provides unconglomerated tables and species lists to allow for the examination of unassigned taxa. Tables are exported as .csv files.

Tables with raw read numbers and the relative abundance of species are also generated to facilitate manual checking of the results in the case of unexpected taxa presence or absence.

```
## Export presence absence matrix for fish ####

path=output

Fish.chordata2<-Fish.chordata


fish_OTUx <- tax_glom(Fish.chordata2, taxrank="Species")
fish_OTUx = transform_sample_counts(fish_OTUx, function(OTU) ifelse(O
TU > 100, OTU, 0))
fish_OTUs = transform_sample_counts(fish_OTUx, function(x) (x/sum(x))
*100)
fish_OTU = transform_sample_counts(fish_OTUx, function(OTU) ifelse(OT
U > 0, 1, OTU))

s1<-as.data.frame(otu_table(fish_OTU))
s1_t<-t(s1)
s2<-as.data.frame(tax_table(fish_OTU))

comb<-merge(s1_t,s2, by="row.names")
red_col<-ncol(comb)
comb1<-comb[-c(1,(red_col-6):(red_col-1))]
```

```
red_col2<-ncol(comb1)
comb_t = setNames(data.frame(t(comb1[,-red_col2])), comb1[,red_col2])
rows <- row.names(comb_t)
rows

comb_t=comb_t%>%mutate(sample_name=as.factor(rows))


write.csv(comb_t, file.path(path,"mifish_presence_absence_unrarefied_
20200807_nogaps_80boot.csv"))
write.csv(s2,file.path(path,"mifish_species_detected_unrarefied_20200
807_nogaps_80boot.csv"))



s1.x<-as.data.frame(otu_table(fish_OTUx))
s1.x_t<-t(s1.x)
s2.x<-as.data.frame(tax_table(fish_OTUx))

comb.x<-merge(s1.x_t,s2.x, by="row.names")
red_col.x<-ncol(comb.x)
comb1.x<-comb.x[-c(1,(red_col.x-6):(red_col.x-1))]

red_col2.x<-ncol(comb1.x)
comb_t.x = setNames(data.frame(t(comb1.x[,-red_col2.x])), comb1.x[,re
d_col2.x])
rowsx <- row.names(comb_t.x)
rowsx

comb_t.x=comb_t.x%>%mutate(sample_name=as.factor(rowsx))

write.csv(comb_t.x, file.path(path, "mifish_read_numbers_unrarefied_2
0200807_nogaps_80boot.csv"))


s1.y<-as.data.frame(otu_table(fish_OTUs))
s1.y_t<-t(s1.x)
s2.y<-as.data.frame(tax_table(fish_OTUs))

comb.y<-merge(s1.y_t,s2.y, by="row.names")
red_col.y<-ncol(comb.y)
comb1.y<-comb.y[-c(1,(red_col.y-6):(red_col.y-1))]

red_col2.y<-ncol(comb1.y)
comb_t.y = setNames(data.frame(t(comb1.y[,-red_col2.y])), comb1.y[,re
d_col2.y])
rowsy <- row.names(comb_t.y)
rowsy

comb_t.y=comb_t.y%>%mutate(sample_name=as.factor(rowsy))
```

```
write.csv(comb_t.y, file.path(path, "relative_abundance_unrarefied_20
200807_nogaps_80boot.csv"))
```

Rarefy the data

Generate rarefaction curves to check the cutoff value for the rarefaction step. Ideally this will be at least 2000, but if possible 8000-12000 is best. This reduces the risk of losing rare ASVs. Rarefaction is necessary to ensure even sampling effort across samples so the relative abundance of DNA from different taxa can be compared. Presence/absence tables using un-rarefied data may provide an indication of rare taxa presence, but the relative abundance of DNA in unrarefied data (e.g. number of reads matching that taxa) cannot be compared.

```
set.seed(100)

fish.chordata.rarefied = rarefy_even_depth(Fish.chordata, sample.size
= 4000, replace = FALSE, trimOTUs = TRUE, verbose = TRUE)
```

Create data tables for rarefied data

This code chunk converts read numbers into presence (1) or absence (0) on the rarefied data. Samples with less than 100 reads are removed as they may be the result of index hopping or contamination from neighbouring samples.

The taxonomy is conglomerated to merge sequences belonging to the same taxa for ease of presentation. There is a code chunk below that provides unconglomerated tables and species lists to allow for the examination of unassigned taxa. Tables are exported as .csv files.

Tables with raw read numbers and the relative abundance of species are also generated to facilitate manual checking of the results in the case of unexpected taxa presence or absence.

```
## Export presence absence matrix for fish ####

fish_OTU2 <- tax_glom(fish.chordata.rarefied, taxrank="Species")
fish_OTUz = transform_sample_counts(fish_OTU2, function(x) (x/sum(x))
*100)

fish_OTU3 = transform_sample_counts(fish_OTU2, function(OTU) ifelse(O
TU > 0, 1, OTU))

s1.2<-as.data.frame(otu_table(fish_OTU3))
s1.2_t<-t(s1.2)
s2.2<-as.data.frame(tax_table(fish_OTU3))

comb.2<-merge(s1.2_t,s2.2, by="row.names")
red_col.2<-ncol(comb.2)
```

```
comb1.2<-comb.2[-c(1,(red_col.2-6):(red_col.2-1))]

red_col2.2<-ncol(comb1.2)
comb_t.2 = setNames(data.frame(t(comb1.2[,-red_col2.2])), comb1.2[,re
d_col2.2])
rows2 <- row.names(comb_t.2)
rows2

comb_t.2=comb_t.2%>%mutate(sample_name=as.factor(rows2))


write.csv(comb_t.2, file.path(path, "mifish_presence_absence_rarefied
_20200807_nogaps_80boot.csv"))
write.csv(s2.2,file.path(path, "mifish_species_detected_rarefied_2020
0807_nogaps_80boot.csv"))

s1.3<-as.data.frame(otu_table(fish_OTU2))
s1.3_t<-t(s1.3)
s2.3<-as.data.frame(tax_table(fish_OTU2))

comb.3<-merge(s1.3_t,s2.3, by="row.names")
red_col.3<-ncol(comb.3)
comb1.3<-comb.3[-c(1,(red_col.3-6):(red_col.3-1))]

red_col2.3<-ncol(comb1.3)
comb_t.3 = setNames(data.frame(t(comb1.3[,-red_col2.3])), comb1.3[,re
d_col2.3])
rows3 <- row.names(comb_t.3)
rows3

comb_t.3=comb_t.3%>%mutate(sample_name=as.factor(rows3))

write.csv(comb_t.3, file.path(path, "mifish_read_numbers_rarefied_202
00807_nogaps_80boot.csv"))

s1.z<-as.data.frame(otu_table(fish_OTUz))
s1.z_t<-t(s1.z)
s2.z<-as.data.frame(tax_table(fish_OTUz))

comb.z<-merge(s1.z_t,s2.z, by="row.names")
red_col.z<-ncol(comb.z)
comb1.z<-comb.z[-c(1,(red_col.z-6):(red_col.z-1))]

red_col2.z<-ncol(comb1.z)
comb_t.z = setNames(data.frame(t(comb1.z[,-red_col2.z])), comb1.z[,re
d_col2.z])
rowsz <- row.names(comb_t.z)
rowsz

comb_t.z=comb_t.z%>%mutate(sample_name=as.factor(rowsz))
write.csv(comb_t.z, file.path(path, "relative_abundance_rarefied_2020
0807_nogaps_80boot.csv"))
```

Join the results with metadata

This code joins the presence/absence tables and relative read abundance tables with the metadata to facilitate data analysis and graphing.

```
rarefied_rel_abundance<-read.csv("C:/Users/laura.kelly/Desktop/Fish_eDNA/Mifish_validation/output/relative_abundance_rarefied_20200807_nogaps_80boot.csv")

pres_abs_raw<-read.csv("C:/Users/laura.kelly/Desktop/Fish_eDNA/Mifish_validation/output/mifish_presence_absence_unrarefied_20200807_nogaps_80boot.csv")

fish_data<-merge(metadata, pres_abs_raw, by="sample_name")

fish_data2<-merge(metadata, rarefied_rel_abundance, by="sample_name")

write.csv(fish_data, file.path(path, "fish_presenceabsence.csv"))
write.csv(fish_data2, file.path(path, "fish_rel_abundance.csv"))
```

Appendix 6. Bioinformatic pipeline for Teleo primer set

Load libraries

Libraries are the packages that contain the functions used for the bioinformatic analysis. The following code loads these into the R environment.

```
library(dada2)
library(ggplot2)
library(knitr)
library(kableExtra)
library(phyloseq)
library(tidyverse)
library(plyr)
library(dplyr)
library(viridis)
library(ranacapa)
library(ShortRead)
library(Biostrings)
library(vegan)
library(remotes)
```

Pre-processing

File-paths should be defined at the start of the session, which reduces the need for hard-coding of these throughout the code. The advantage of this is that if the code is run on a different computer or file-paths need to be changed, it minimises the amount of code requiring modification.

```
path1 <- Sys.glob("C:/Users/laura.kelly/Desktop/Fish_eDNA/AG0246-Tele
o/*")  ## CHANGE ME to the directory containing the fastq files.

path <- "C:/Users/laura.kelly/Desktop/Fish_eDNA/Teleo_test"

list.files(path1)

fnFs <- sort(list.files(path1, pattern = "L001_R1_001.fastq.gz", full
.names = TRUE))
fnRs <- sort(list.files(path1, pattern = "L001_R2_001.fastq.gz", full
.names = TRUE))
```

Primer sequences are defined at the start of the session so they can be found and trimmed from the sequences using cutadapt.

```
FWD <- "ACACCGCCCGTCACTCT"
REV <- "CTTCCGGTACACTTACCATG"
```

Vectors are made with all the possible orientations of both the forward and reverse primers.

```
allOrients <- function(primer) {
    # Create all orientations of the input sequence
    require(Biostrings)
    dna <- DNAString(primer)  # The Biostrings works w/ DNAString obj
ects rather than character vectors
    orients <- c(Forward = dna, Complement = complement(dna), Reverse
= reverse(dna),
        RevComp = reverseComplement(dna))
    return(sapply(orients, toString))  # Convert back to character ve
ctor
}
FWD.orients <- allOrients(FWD)
REV.orients <- allOrients(REV)
FWD.orients
```

The following step is a check step. This calculates the number of reads where the forward and reverse primers are found from one sample. If the number of hits are very low, this may indicate a sequencing problem or an issue with the user-defined primer sequences.

```
primerHits <- function(primer, fn) {
    # Counts number of reads in which the primer is found
    nhits <- vcountPattern(primer, sread(readFastq(fn)), fixed = FALS
E)
    return(sum(nhits > 0))
}
rbind(FWD.ForwardReads = sapply(FWD.orients, primerHits, fn = fnFs[[1
]]),
    REV.ReverseReads = sapply(REV.orients, primerHits, fn = fnRs[[1]]
))
```

Use cutadapt to trim primers

Cutadapt removes the primer sequences from the ends of the sequences. Sequences where the primers are not found are discarded from the output.

```
cutadapt <- "C:/Users/laura.kelly/AppData/Local/Continuum/miniconda3/
envs/cutadapt/Scripts/cutadapt" # CHANGE ME to the cutadapt path on y
our machine
system2(cutadapt, args = "--version") # Run shell commands from R
```

```
path.cut <- file.path(path, "cutadapt")

if(!dir.exists(path.cut)) dir.create(path.cut)

fnFs.cut <- file.path(path.cut, basename(fnFs))
```

```
fnRs.cut <- file.path(path.cut, basename(fnRs))

# Trim FWD off of R1 (forward reads) -
R1.flags <- paste0("-g", " ^", FWD)
# Trim REV off of R2 (reverse reads)
R2.flags <- paste0("-G", " ^", REV)


for(i in seq_along(fnFs)) {
  system2(cutadapt, args = c(R1.flags, R2.flags, "-e", 0.05,
                             "--discard-untrimmed",
                             "-o", fnFs.cut[i], "-p", fnRs.cut[i],
                             fnFs[i], fnRs[i]))
}
```

The code below is a check step to ensure that no primers are left on the reads. Occasionally some primers are present due to internal primer hits (primer sequences in the middle of the sequences) this is okay as these are removed later in the process. This step also checks that the same number of forward and reverse reads remain after cutadapt has processed them. If they are different an error message will be generated for the user to check the processing steps.

```
path.cut <- file.path(path, "cutadapt")

if(!dir.exists(path.cut)) dir.create(path.cut)

fnFs.cut <- file.path(path.cut, basename(fnFs))
fnRs.cut <- file.path(path.cut, basename(fnRs))

cutFs <- sort(list.files(path.cut, pattern = "R1_001.fastq.gz", full.
names = TRUE))
cutRs <- sort(list.files(path.cut, pattern = "R2_001.fastq.gz", full.
names = TRUE))

if(length(cutFs) == length(cutRs)) print("Forward and reverse files m
atch. Go forth and find nemo")
if (length(cutFs) != length(cutRs)) stop("Forward and reverse files d
o not match. Go back and have a check")
```

Extract sample names

Extract the sample names and split the sequences into forward and reverse files, which makes downstream processing easier.

```
# Extract sample names, assuming filenames have format:
get.sample.name <- function(fname) strsplit(basename(fname), "_")[[1]
][1]
sample.names <- unname(sapply(cutFs, get.sample.name))
head(sample.names)
```

```
# Split files into forward and reverse to make it easier later
path.cut.F <- file.path(path, "cutadapt", "forward")
path.cut.R <- file.path(path, "cutadapt", "reverse")

if(!dir.exists(path.cut.F)) dir.create(path.cut.F)
if(!dir.exists(path.cut.R)) dir.create(path.cut.R)

file.copy(list.files(path1, pattern = "L001_R1_001.fastq.gz", full.na
mes = TRUE), path.cut.F)
file.copy(list.files(path1, pattern = "L001_R2_001.fastq.gz", full.na
mes = TRUE), path.cut.R)
```

Quality plots

This step is critical. The read quality profiles are plotted (initially for a subset of the reads). If there are fewer than 20 samples plot all of the samples. If there are more than 20, then plot a random subset of 20. This code requires user input to change which option is used at the end (by modifying which lines have # in the front). The quality plots should show high quality across the read, with a reduction at the end. If the quality plot shows the quality score dropping early in the read or the quality being low throughout, this indicates poor quality sequences. This can occur in some samples, but if it occurs in more than one of the 20 plots, it is worth investigating the other samples as this may indicate a poor sequencing run.

```
if(length(cutFs) <= 20) {
  forplots <- plotQualityProfile(cutFs) +
  scale_x_continuous(breaks=seq(0,250,10)) +
  scale_y_continuous(breaks=seq(0,40,2)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
  revplots <- plotQualityProfile(cutRs) +
  scale_x_continuous(breaks=seq(0,250,10)) +
  scale_y_continuous(breaks=seq(0,40,2)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
} else {
  rand_samples <- sample(size = 20, 1:length(cutFs)) # grab 20 random
samples to plot
  fwd_qual_plots <- plotQualityProfile(cutFs[rand_samples]) +
  scale_x_continuous(breaks=seq(0,250,10)) +
  scale_y_continuous(breaks=seq(0,40,2)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
  rev_qual_plots <- plotQualityProfile(cutRs[rand_samples]) +
  scale_x_continuous(breaks=seq(0,250,10)) +
  scale_y_continuous(breaks=seq(0,40,2)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
}

#forplots   # use this if there are fewer than 20 samples
#revplots   # use this if there are fewer than 20 samples
```

```
fwd_qual_plots # use this if there are more than 20 samples
rev_qual_plots # use this if there are more than 20 samples
```

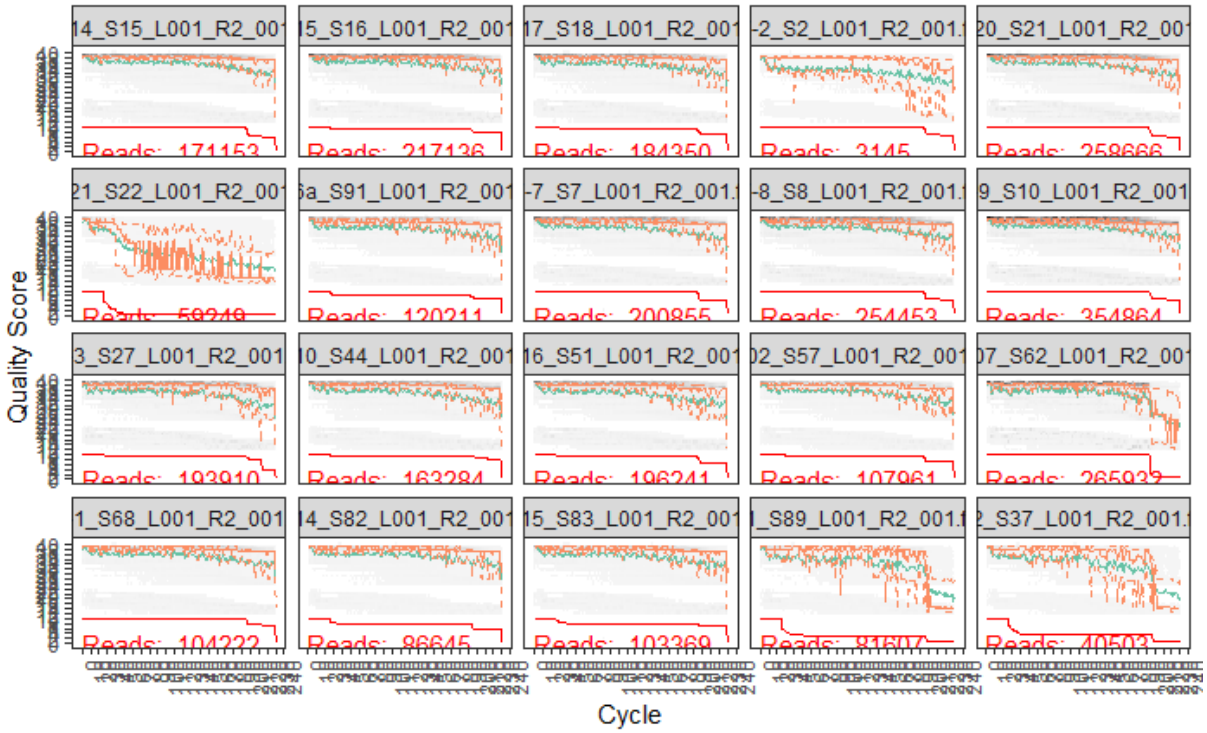


Figure A6.1. Example of quality plots. Notice that sample 6 (first column, second row) shows reduced quality early in the cycling profile indicating poor sequence quality.

Filter and trim the sequences

There are a number of options in the below code that may need to be changed.

Commands

fwd	file path of where to find the forward reads
filt	file path of where to put the filtered forward reads
rev	file path of where to find the reverse reads
filt.rev	file path of where to put the filtered reverse reads
truncLen	length of which to truncate the reads. The first number refers to the forward read and the second to the reverse. This is user and dataset variable. There is a trade-off between quality and how much you trim. The more you trim the better quality the final read will have. However,

	the forward and reverse reads still need to overlap so you can't trim too much.
maxEE	after truncation this is the maximum number of ,expected errors, allowed before a read is discarded. Expected errors are calculated as EE = sum(10^(-Q/10)) - based on the quality scores of the reads. In general the reverse reads generally have a lower quality and should be allowed more expected errors. If, however, your plots indicate that over the read length the quality remains high, this can be left the same for both forward and reverse reads.
truncQ	2 is a special quality score from Illumina denoting the start of a bad read. There is little effect of using 2 if reads are truncated. You can set it to another score but will truncate the read on the first appearance of the score which will likely end up in reads not overlapping
maxN	How many ambiguous bases are allowed. Best to keep this at 0
rm.phix	Do you want to remove phiX from the samples. Phi X is a bacteriophage genome that is spiked into samples run on Illumina machines as a quality control and to aid in mitigating issues from low diversity (amplicon) libraries. It is normally removed when samples are processed from the machine and also shouldn't get through the cutadapt stage but this option can be kept at TRUE to remove any that have somehow made it through the earlier processing stages.

```
pathF <- path.cut.F
pathR <- path.cut.R

filtpathF <- file.path(pathF, "filtered")
filtpathR <- file.path(pathR, "filtered")


fastqFs <- sort(list.files(pathF, pattern="fastq"))
fastqRs <- sort(list.files(pathR, pattern="fastq"))
if(length(fastqFs) != length(fastqRs)) stop("There's a problem. Go ba
ck and check files")

out <- filterAndTrim(fwd=file.path(pathF, fastqFs), filt=file.path(fi
ltpathF, fastqFs),
                     rev=file.path(pathR, fastqRs), filt.rev=file.
path(filtpathR, fastqRs),
                      truncLen=c(85,85), maxEE=c(2,4), truncQ=2, m
axN=0, rm.phix=TRUE,
                    compress=TRUE, verbose=TRUE, multithread=TRUE)
kable(out) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

The output table below summarises the reads in and reads out of the filter and trim process. Check to make sure not too many reads are being lost at this stage as it could indicate a problem with the data.

	reads.in	reads.out
200127-1_S1_L001_R1_001.fastq.gz	1280	947
200127-10_S11_L001_R1_001.fastq.gz	176946	159584
200127-10a_S95_L001_R1_001.fastq.gz	174904	152449
200127-11_S12_L001_R1_001.fastq.gz	240674	204282
200127-11a_S96_L001_R1_001.fastq.gz	21115	15374
200127-12_S13_L001_R1_001.fastq.gz	129319	117272
200127-13_S14_L001_R1_001.fastq.gz	182726	162385
200127-14_S15_L001_R1_001.fastq.gz	175367	161956
200127-15_S16_L001_R1_001.fastq.gz	223404	197156
200127-16_S17_L001_R1_001.fastq.gz	132148	121001
200127-17_S18_L001_R1_001.fastq.gz	188823	168926
200127-18_S19_L001_R1_001.fastq.gz	147102	134875
200127-19_S20_L001_R1_001.fastq.gz	28365	25942
200127-1a_S85_L001_R1_001.fastq.gz	103215	92296

Figure A6.2. Example output table that is generated to show the number of reads before and after quality filtering and trimming of the sequences.

Check outputs

This is another data processing step, both to check outputs and to manipulate the files for easier processing downstream. The code lists the filtered fastq files, removes the fastq.gz part of the filenames (this is a compressing option) and stores it in samples names, checks the names of the forward and reverse files match and renames the filtered output objects.

```
filtFs <- list.files(filtpathF, pattern="fastq", full.names = TRUE)
filtRs <- list.files(filtpathR, pattern="fastq", full.names = TRUE)
sample.names <- sapply(strsplit(basename(filtFs), "_"), `[`, 1) # Ass
umes filename = samplename_XXX.fastq.gz
sample.namesR <- sapply(strsplit(basename(filtRs), "_"), `[`, 1) # As
```

```
sumes filename = samplename_XXX.fastq.gz
if(!identical(sample.names, sample.namesR)) stop("Forward and reverse
files do not match - go back and check files")
names(filtFs) <- sample.names
names(filtRs) <- sample.namesR
```

Error profiles

This code lets dada2 learn the error profiles of the sequences. To correct sequences, dada2 uses a parametric error model which is specific to each data set. It achieves this model by using a machine learning technique which alternates between the estimation of error rates and inferring the sample composition until these two values converge.

We use 1×10^8 bases to calculate the error profile. The higher the number of bases used the more accurate the error profile will be, but it is a trade-off with speed.

```
errF <- learnErrors(filtFs, nbases = 1e8, MAX_CONSIST = 15, multithread=TR
UE, verbose = TRUE)
errR <- learnErrors(filtRs, nbases = 1e8, MAX_CONSIST = 15, multithread=TR
UE, verbose = TRUE)
```

Check that the error profile is sensible. The red line shows the error under the nominal definition of the Q-score. Observed points (dots) should not deviate far from the black line (estimated error rates).

There should be a general negative trend between error frequency and quality.

```
errors_f <- plotErrors(errF, nominalQ=TRUE)
errors_f

errors_r <- plotErrors(errR, nominalQ=TRUE)
errors_r
```

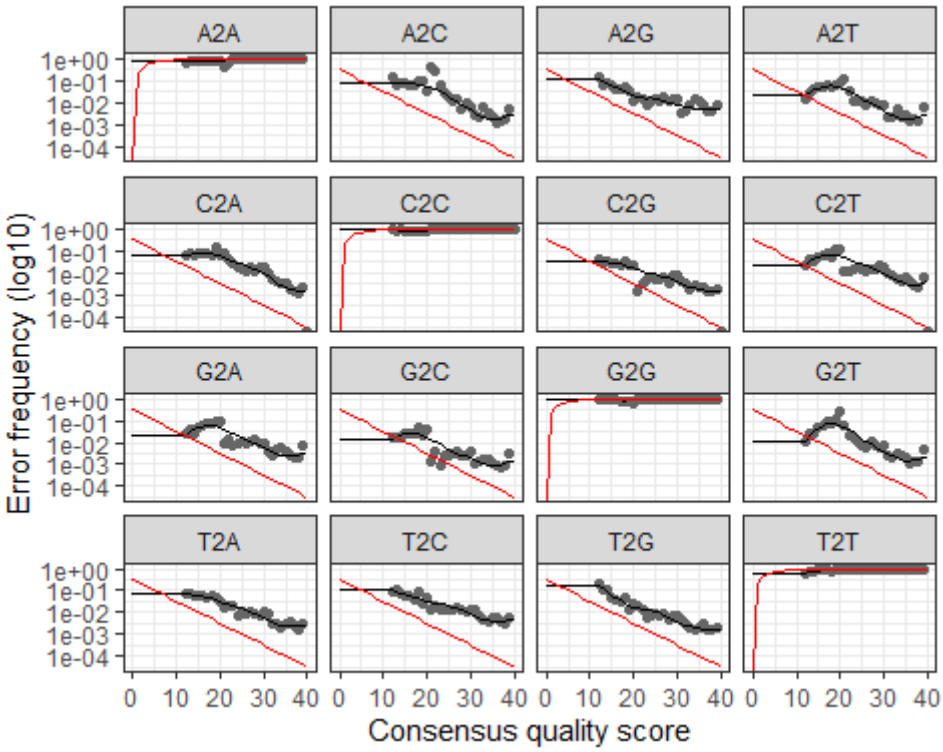


Figure A6.3. An example of the error profile plots generated by dada2. Note the red line is the nominal error based on the quality score, the black line is the dada2 error profile and the black dots are the observed points.

Dereplicate the sequences

Dereplicating combines the identical sequencing reads into ,unique sequences,.

This process gives an abundance value to the unique sequences equal to that of the number of reads with that sequence. Quality information for the unique sequences is kept as the average of that of all reads combined to form the unique sequence.

This step reduces the computational power required for future steps.

```
derepF <- derepFastq(filtFs, verbose=TRUE)
derepR <- derepFastq(filtRs, verbose=TRUE)

dadaF.pseudo <- dada(derepF, err=errF, multithread=TRUE)
dadaR.pseudo <- dada(derepR, err=errR, multithread=TRUE)
```

Merge reads and make a sequence table

Merge the forward and reverse reads and produce a sequence table.

Inputs are the results from the inference step (*dadaF.pseudo* and *dadaR.pseudo*) as well as the dereplicated sequences (*derepF* and *derepR*).

maxMismatch number of mismatches allowed in your overlap region.

minOverlap	the minimum length you want the sequences to overlap by. This is a factor of how long your amplicon is and how much trimming you did earlier. The longer the overlap the higher confidence you have of the sequence being of good quality, however setting this value too high will result in a failure of sequences to merge.

```
mergers <- mergePairs(dadaF.pseudo, derepF, dadaR.pseudo, derepR, max
Mismatch = 1, minOverlap = 40, verbose=TRUE)
seqtab <- makeSequenceTable(mergers)
saveRDS(seqtab, "C:/Users/laura.kelly/Desktop/Fish_eDNA/Teleo/teleo_e
DNA.rds")
```

Chimera removal and output checks

Check sequence lengths and run a chimera check.

The approximate size of the sequences that should be produced with a primer set are known. The filtering step removes sequences that are considerably longer or shorter than the expected size range as these are likely to be artefacts of the PCR or sequencing processes.

Following the size-filtering step, chimeras need to be removed. Chimeras are sequences that are made up of the DNA of two (or more) species. Sometimes during PCR, the enzymes end up copying the sequence of half of one DNA fragment and half of another. These need to be removed for the following analysis steps.

```
trimtable <- as.data.frame(table(nchar(getSequences(seqtab))))
colnames(trimtable) <- c("Length (bp)", "Frequency")
kable(trimtable)

seqtab2 <- seqtab[,nchar(colnames(seqtab)) %in% seq(90,140)]
seqtab.nochim <- removeBimeraDenovo(seqtab2, multithread=TRUE, verbos
e=TRUE)
saveRDS(seqtab.nochim, "C:/Users/laura.kelly/Desktop/Fish_eDNA/Teleo/
teleo_nochim_eDNA.rds")
```

Check losses

Produce a table which shows the number of reads at each stage. Most reads are likely to be lost at the trimming stage. Check to see that the merging step does not result in very few reads (this could be a sign of trimming the sequences too short).

To identify the potential losses throughout the filtering steps, the number of sequences at each step is reported in a table.

input	is the number of raw sequence reads from Illumina after the primers have been trimmed from the sequences. Note that any reads lost during the primer trimming step are not included.
filtered	is the number of sequences that remain following read-quality filtering.
denoisedF	is the number of sequences remaining in the forward reads following error profiling.
denoisedR	is the number of sequences remaining in the reverse reads following error profiling.
merged	is the number of sequences that successfully merged. Note a large loss of sequence numbers at this step indicates that truncation of sequences in the initial quality filtering steps is too harsh, meaning the sequences are not long enough to merge.
nochim	is the number of sequences remaining after chimera removal. It is normal to have a reasonable number of sequences removed during this step.

```
getN <- function(x) sum(getUniques(x))
track <- cbind(out, sapply(dadaF.pseudo, getN), sapply(dadaR.pseudo,
getN), sapply(mergers, getN), rowSums(seqtab.nochim))
colnames(track) <- c("input", "filtered", "denoisedF", "denoisedR", "
merged", "nonchim")
rownames(track) <- sample.names

kable(track) %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

Assign taxonomy

The following code chunks allow the assignment of taxonomy to the sequences from the steps above. The metadata file, reference database file and sequences are required for the following steps. Note that the formatting of the metadata file is very important, and the sample names of the metadata file and the sequence file must match in order for the following code to successfully run.

Define the location of the reference database.

```
# database location
fishDB <- "C:/Users/laura.kelly/Desktop/Fish_eDNA/Fish_12S_20200629_n
ogaps.fasta"
```

Set up input and output folders, load the datafile and set-up the output prefixes.

```
username <- "Teleo_test"
input <- paste0("C:/Users/laura.kelly/Desktop/Fish_eDNA/", username,
"/input/")
output <- paste0("C:/Users/laura.kelly/Desktop/Fish_eDNA/", username,
"/output/")
```

```
#Load Datafile
sq <- getSequences(readRDS("C:/Users/laura.kelly/Desktop/Fish_eDNA/Te
leo/teleo_nochim_eDNA.rds"))
seqtab.nochim<-readRDS("C:/Users/laura.kelly/Desktop/Fish_eDNA/Teleo/
teleo_nochim_eDNA.rds")

# Setup Output prefix for slice files (rds)
file_name_prefix <- paste0(output, 'teleo.nochimera.tax.slice_')
```

Taxonomy assignment can be a memory-intensive process. To reduce the chance of running out of memory, especially with larger datasets, it is possible to assign the taxonomy for the sequences in "chunks". These are saved as separate files and then merged at the end to produce a single file containing the taxonomy assignments for all of the sequences.

Note: There is a parameter in this code called *minBoot* which impacts the stringency of taxonomic assignments. The taxonomy is assigned by determining the kmer profile of the sequence and then matching this with the kmer profiles of the sequences in the reference database. The *minBoot* designates the number of times out of 100 the same taxonomic assignment must be made for that taxonomic assignment to be designated in the output. The default setting is 50, however, for sequences over 250 bases the recommended threshold is 80.

The following code runs as chunks. Set the chunk sizes, break dataset down into chunks and assign taxonomy for each chunk.
IMPORTANT: change the stringency of taxonomic assignment and other settings below *before* running chunk. Note that there are two parts of the loop that need to be changed.

```
# Set chunksize
CHUNKSIZE = 10000

# Calculate number of slices and remainder
NUM_SQ = length(sq)
NUM_SLICES <- as.integer(NUM_SQ/CHUNKSIZE)
LEN_REMAINDER <- as.integer(NUM_SQ%%CHUNKSIZE)

# Compute full slices
idx <- 0
while (idx < NUM_SLICES)
{
  start_idx = (idx*CHUNKSIZE)+1
  end_idx =   (idx+1)*CHUNKSIZE
  fname = paste0(file_name_prefix, start_idx, '_', end_idx, '.rds')

  result_slice <- assignTaxonomy(sq[start_idx:end_idx], fishDB, minBo
ot = 80,multithread=TRUE)
```

```
  saveRDS(result_slice, fname)
  idx <- idx+1
}
# Compute remainder if present
if (LEN_REMAINDER!=0){
  start_idx = (NUM_SLICES*CHUNKSIZE)+1
  end_idx =   NUM_SQ
  fname = paste0(file_name_prefix, start_idx, '_', end_idx, '.rds')

  result_slice <- assignTaxonomy(sq[start_idx:end_idx], fishDB, minBo
ot = 80, multithread=TRUE)
  saveRDS(result_slice, fname)
}
```

If there are multiple taxonomy slices the files need to be combined. Transfer all *.rds files to local machine if run on HPC or other computer cluster.
Run combine on desktop Computer e.g.:
IMPORTANT: change the file locations below *before* running chunk

```
setwd(output)

teleo_taxa<-readRDS("C:/Users/laura.kelly/Desktop/Fish_eDNA/Teleo_tes
t/output/teleo.nochimera.tax.slice_1_672.rds")
```

Create a phyloseq object

Phyloseq is a package that enables the manipulation of next generation sequencing data. There are three components that make up the *mifish.physeq* phyloseq object:

otu_table is the table of unique sequences from the study. This is the list of sequences with chimeras removed. This is called *mifish.nochim.rds* in the code chunk below.\
tax_table is the taxonomy table generated from the assign taxonomy section above. Note this requires all of the slices that were generated to be bound together as in the code above. This is defined as *mifish_taxa* in the current code chunk.\
sample_data is the metadata file from the study containing all of the other information about the samples including variables such as site name, region, volume filtered, and filter type used. Any information of interest to the study can be recorded in the metadata file but this must be recorded in separate columns for each variable of interest.

```
metadata<-read.csv("C:/Users/laura.kelly/Desktop/Fish_eDNA/ANDe sampl
es_v1_teleo.csv")
metadata=metadata%>%mutate(sample_name=as.factor(sample_name))
rownames(metadata) = metadata$sample_name
```

```
teleo_nochim_eDNA.rds<-readRDS("C:/Users/laura.kelly/Desktop/Fish_eDNA/Teleo/teleo_nochim_eDNA.rds")

teleo.physeq <- phyloseq(otu_table(seqtab.nochim, taxa_are_rows = FALSE),
                         tax_table(teleo_taxa),
                         sample_data(metadata))
```

Clean the data

Subtract amplicon sequence variants (ASVs) from negative controls.

Throughout the field collection, DNA extraction, PCR and sequencing process, negative control samples are included to ensure that any contamination is identifiable. How contamination is managed depends on the type and degree of any contamination of the negative controls. Ideally any significant contamination prior to sequencing will be identified at the PCR stage (using visualisation on an agarose gel) and samples re-run through the PCR steps.

Sequences in the negative controls can still occur. The following code chunk takes the various controls (DNA extraction, PCR, and sequencing controls) and determines the number of each unique sequence in them. It is assumed that in a "worst case" scenario the same number of contaminating sequences will be present in all of the samples, thus the number of each of these unique sequences is subtracted from the samples below.

For DNA extraction controls, the relevant sequences are subtracted only from samples that belong to the same DNA extraction batch. For PCR and sequencing controls, these are subtracted from all of the samples they are relevant to (i.e. the same PCR run or sequencing run).

Note, the code below needs to be modified for each sequencing run to account for the setup of controls across the plate. Set up a template to keep this consistent, which will reduce the need to modify this code.

```
Controls = subset_samples(teleo.physeq , Type == "control"|Type=="field" & Batch!="pos")
Controls = filter_taxa(Controls, function(x) sum(x) > 0, TRUE)
sample_sums(Controls)

## __subset phyloseq project by batches (extraction controls) ####

Batch1.chordata = subset_samples(teleo.physeq, Batch=="batch1")
Batch2.chordata = subset_samples(teleo.physeq, Batch=="batch2")
Batch3.chordata = subset_samples(teleo.physeq, Batch=="batch3")
Batch4.chordata = subset_samples(teleo.physeq, Batch=="batch4")
```

```
Batch5.chordata = subset_samples(teleo.physeq, Batch=="batch5")
Batch6.chordata = subset_samples(teleo.physeq, Batch=="batch6")
Batch7.chordata = subset_samples(teleo.physeq, Batch=="batch7")
Batch8.chordata = subset_samples(teleo.physeq, Batch=="batch8")
Batch9.chordata = subset_samples(teleo.physeq, Batch=="batch9")
Batch10.chordata = subset_samples(teleo.physeq, Batch=="batch10")
Batch11.chordata = subset_samples(teleo.physeq, Batch =="pos")
Batch12.chordata = subset_samples(teleo.physeq, Batch =="global")

## __subset the negative controls (not all batches contain a neg ctrl
) ####

Batch1.chordata_neg = subset_samples(Batch1.chordata, Type == "contro
l"|Type=="field")
Batch2.chordata_neg = subset_samples(Batch2.chordata, Type == "contro
l"|Type=="field")
Batch5.chordata_neg = subset_samples(Batch5.chordata, Type == "contro
l"|Type=="field")
Batch6.chordata_neg = subset_samples(Batch6.chordata, Type == "contro
l"|Type=="field")
Batch7.chordata_neg = subset_samples(Batch7.chordata, Type == "contro
l"|Type=="field")
Batch8.chordata_neg = subset_samples(Batch8.chordata, Type == "contro
l"|Type=="field")
Batch9.chordata_neg = subset_samples(Batch9.chordata, Type == "contro
l"|Type=="field")
Batch10.chordata_neg = subset_samples(Batch10.chordata, Type == "cont
rol"|Type=="field")
Batch11.chordata_neg = subset_samples(Batch11.chordata, Type == "cont
rol"|Type=="field")
Batch12.chordata_neg = subset_samples(Batch12.chordata, Type == "cont
rol"|Type=="field")


## calculate column sums ####

Batch1.chordata_neg_sums <- colSums(otu_table(Batch1.chordata_neg))
Batch2.chordata_neg_sums <- colSums(otu_table(Batch2.chordata_neg))
Batch5.chordata_neg_sums <- colSums(otu_table(Batch5.chordata_neg))
Batch6.chordata_neg_sums <- colSums(otu_table(Batch6.chordata_neg))
Batch7.chordata_neg_sums <- colSums(otu_table(Batch7.chordata_neg))
Batch8.chordata_neg_sums <- colSums(otu_table(Batch8.chordata_neg))
Batch9.chordata_neg_sums <- colSums(otu_table(Batch9.chordata_neg))
Batch10.chordata_neg_sums <- colSums(otu_table(Batch10.chordata_neg))
Batch11.chordata_neg_sums <- colSums(otu_table(Batch11.chordata_neg))
Batch12.chordata_neg_sums <- colSums(otu_table(Batch12.chordata_neg))

## find the max value of the controls ####

Batch1.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch1.chordata_neg)))), 1, max)
Batch2.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
```

```
Batch2.chordata_neg)))), 1, max)
Batch5.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch5.chordata_neg)))), 1, max)
Batch6.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch6.chordata_neg)))), 1, max)
Batch7.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch7.chordata_neg)))), 1, max)
Batch8.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch8.chordata_neg)))), 1, max)
Batch9.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table(
Batch9.chordata_neg)))), 1, max)
Batch10.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table
(Batch10.chordata_neg)))), 1, max)
Batch11.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table
(Batch11.chordata_neg)))), 1, max)
Batch12.chordata_neg_max <- apply(as.data.frame(as.matrix(t(otu_table
(Batch12.chordata_neg)))), 1, max)


## move the max value to a vector so it can be subtracted ####

Batch1.chordata_neg_sums_vec <- as.vector(Batch1.chordata_neg_max)
Batch2.chordata_neg_sums_vec <- as.vector(Batch2.chordata_neg_max)
Batch5.chordata_neg_sums_vec <- as.vector(Batch5.chordata_neg_max)
Batch6.chordata_neg_sums_vec <- as.vector(Batch6.chordata_neg_max)
Batch7.chordata_neg_sums_vec <- as.vector(Batch7.chordata_neg_max)
Batch8.chordata_neg_sums_vec <- as.vector(Batch8.chordata_neg_max)
Batch9.chordata_neg_sums_vec <- as.vector(Batch9.chordata_neg_max)
Batch10.chordata_neg_sums_vec <- as.vector(Batch10.chordata_neg_max)
Batch11.chordata_neg_sums_vec <- as.vector(Batch11.chordata_neg_max)
Batch12.chordata_neg_sums_vec <- as.vector(Batch12.chordata_neg_max)

## move the sums to a vector so they can be subtracted ####

Batch1.chordata_neg_sums_vec <- as.vector(Batch1.chordata_neg_sums)
Batch2.chordata_neg_sums_vec <- as.vector(Batch2.chordata_neg_sums)
Batch5.chordata_neg_sums_vec <- as.vector(Batch5.chordata_neg_sums)
Batch6.chordata_neg_sums_vec <- as.vector(Batch6.chordata_neg_sums)
Batch7.chordata_neg_sums_vec <- as.vector(Batch7.chordata_neg_sums)
Batch8.chordata_neg_sums_vec <- as.vector(Batch8.chordata_neg_sums)
Batch9.chordata_neg_sums_vec <- as.vector(Batch9.chordata_neg_sums)
Batch10.chordata_neg_sums_vec <- as.vector(Batch10.chordata_neg_sums)
Batch11.chordata_neg_sums_vec <- as.vector(Batch11.chordata_neg_sums)
Batch12.chordata_neg_sums_vec <- as.vector(Batch12.chordata_neg_sums)

## make ASV table into a dataframe so to be able to subtract ####


B1.chordata = as(otu_table(Batch1.chordata), "matrix")
B1df.chordata = as.data.frame(B1.chordata)
B2.chordata = as(otu_table(Batch2.chordata), "matrix")
B2df.chordata = as.data.frame(B2.chordata)
```

```
B3.chordata = as(otu_table(Batch3.chordata), "matrix")
B3df.chordata = as.data.frame(B3.chordata)
B4.chordata = as(otu_table(Batch4.chordata), "matrix")
B4df.chordata = as.data.frame(B4.chordata)
B5.chordata = as(otu_table(Batch5.chordata), "matrix")
B5df.chordata = as.data.frame(B5.chordata)
B6.chordata = as(otu_table(Batch6.chordata), "matrix")
B6df.chordata = as.data.frame(B6.chordata)
B7.chordata = as(otu_table(Batch7.chordata), "matrix")
B7df.chordata = as.data.frame(B7.chordata)
B8.chordata = as(otu_table(Batch8.chordata), "matrix")
B8df.chordata = as.data.frame(B8.chordata)
B9.chordata = as(otu_table(Batch9.chordata), "matrix")
B9df.chordata = as.data.frame(B9.chordata)



## do the subtraction ####
## __note batch 10 and 12 subtraction should be applied to all batche
s (global negative control) ####

B1df.chordata[,1:length(B1df.chordata)] <- sweep(B1df.chordata[,1:len
gth(B1df.chordata)],2,Batch1.chordata_neg_sums_vec)
B2df.chordata[,1:length(B2df.chordata)] <- sweep(B2df.chordata[,1:len
gth(B2df.chordata)],2,Batch2.chordata_neg_sums_vec)
B5df.chordata[,1:length(B5df.chordata)] <- sweep(B5df.chordata[,1:len
gth(B5df.chordata)],2,Batch5.chordata_neg_sums_vec)
B6df.chordata[,1:length(B6df.chordata)] <- sweep(B6df.chordata[,1:len
gth(B6df.chordata)],2,Batch6.chordata_neg_sums_vec)
B7df.chordata[,1:length(B7df.chordata)] <- sweep(B7df.chordata[,1:len
gth(B7df.chordata)],2,Batch7.chordata_neg_sums_vec)
B8df.chordata[,1:length(B8df.chordata)] <- sweep(B8df.chordata[,1:len
gth(B8df.chordata)],2,Batch8.chordata_neg_sums_vec)
B9df.chordata[,1:length(B9df.chordata)] <- sweep(B9df.chordata[,1:len
gth(B9df.chordata)],2,Batch9.chordata_neg_sums_vec)



B1df.chordata[,1:length(B1df.chordata)] <- sweep(B1df.chordata[,1:len
gth(B1df.chordata)],2,Batch10.chordata_neg_sums_vec)
B2df.chordata[,1:length(B2df.chordata)] <- sweep(B2df.chordata[,1:len
gth(B2df.chordata)],2,Batch10.chordata_neg_sums_vec)
B3df.chordata[,1:length(B3df.chordata)] <- sweep(B3df.chordata[,1:len
gth(B3df.chordata)],2,Batch10.chordata_neg_sums_vec)
B4df.chordata[,1:length(B4df.chordata)] <- sweep(B4df.chordata[,1:len
gth(B4df.chordata)],2,Batch10.chordata_neg_sums_vec)
B5df.chordata[,1:length(B5df.chordata)] <- sweep(B5df.chordata[,1:len
gth(B5df.chordata)],2,Batch10.chordata_neg_sums_vec)
B6df.chordata[,1:length(B6df.chordata)] <- sweep(B6df.chordata[,1:len
gth(B6df.chordata)],2,Batch10.chordata_neg_sums_vec)
B7df.chordata[,1:length(B7df.chordata)] <- sweep(B7df.chordata[,1:len
gth(B7df.chordata)],2,Batch10.chordata_neg_sums_vec)
```

```
B9df.chordata[,1:length(B9df.chordata)] <- sweep(B9df.chordata[,1:len
gth(B9df.chordata)],2,Batch10.chordata_neg_sums_vec)

B1df.chordata[,1:length(B1df.chordata)] <- sweep(B1df.chordata[,1:len
gth(B1df.chordata)],2,Batch12.chordata_neg_sums_vec)
B2df.chordata[,1:length(B2df.chordata)] <- sweep(B2df.chordata[,1:len
gth(B2df.chordata)],2,Batch12.chordata_neg_sums_vec)
B3df.chordata[,1:length(B3df.chordata)] <- sweep(B3df.chordata[,1:len
gth(B3df.chordata)],2,Batch12.chordata_neg_sums_vec)
B4df.chordata[,1:length(B4df.chordata)] <- sweep(B4df.chordata[,1:len
gth(B4df.chordata)],2,Batch12.chordata_neg_sums_vec)
B5df.chordata[,1:length(B5df.chordata)] <- sweep(B5df.chordata[,1:len
gth(B5df.chordata)],2,Batch12.chordata_neg_sums_vec)
B6df.chordata[,1:length(B6df.chordata)] <- sweep(B6df.chordata[,1:len
gth(B6df.chordata)],2,Batch12.chordata_neg_sums_vec)
B7df.chordata[,1:length(B7df.chordata)] <- sweep(B7df.chordata[,1:len
gth(B7df.chordata)],2,Batch12.chordata_neg_sums_vec)
B9df.chordata[,1:length(B9df.chordata)] <- sweep(B9df.chordata[,1:len
gth(B9df.chordata)],2,Batch12.chordata_neg_sums_vec)


## replace the values less than zero with zero ####

B1df.chordata <- replace(B1df.chordata, B1df.chordata < 0, 0)
B2df.chordata <- replace(B2df.chordata, B2df.chordata < 0, 0)
B3df.chordata <- replace(B3df.chordata, B3df.chordata < 0, 0)
B4df.chordata <- replace(B4df.chordata, B4df.chordata < 0, 0)
B5df.chordata <- replace(B5df.chordata, B5df.chordata < 0, 0)
B6df.chordata <- replace(B6df.chordata, B6df.chordata < 0, 0)
B7df.chordata <- replace(B7df.chordata, B7df.chordata < 0, 0)
B8df.chordata <- replace(B8df.chordata, B8df.chordata < 0, 0)
B9df.chordata <- replace(B9df.chordata, B9df.chordata < 0, 0)


## merge ASV tables and check the dimensions ####

cleaned.chordata <- rbind(B1df.chordata, B2df.chordata, B3df.chordata
, B4df.chordata, B5df.chordata, B6df.chordata, B7df.chordata, B8df.ch
ordata, B9df.chordata)
dim(cleaned.chordata) # should rows and columns of match what went in
```

Create a new phyloseq object

Using the cleaned data from the previous steps, create a new phyloseq object. Then extract the taxonomy file and clean up the NA and unclassified taxonomy labels so the taxonomy will show the last taxonomic level that was positively identified (e.g. Galaxias) and "unidentified" for any downstream taxonomic levels.

The taxonomy assignments use "NA" as the default when a taxonomic level cannot be assigned. The following script tidies the data to follow a convention that if a sequence

cannot be assigned at a taxonomic level (i.e. Phylum, Class, Order, Family, Genus or Species), the code will go to the highest taxonomic level that was assigned and input "unclassified" following this. This means that species assigned to genus will be identifiable as *Genus unclassified* rather than simply *NA* in further analyses.

```
teleo.physeq.cleaned <- phyloseq(otu_table(cleaned.chordata, taxa_are
_rows = FALSE),
                                 sample_data(metadata),
                                 tax_table(teleo_taxa))

teleo.physeq.cleaned = subset_samples(teleo.physeq.cleaned, Type !="c
ontrol")

tax.clean <- data.frame(tax_table(teleo.physeq.cleaned))
for (i in 1:7){ tax.clean[,i] <- as.character(tax.clean[,i])}
tax.clean[is.na(tax.clean)] <- ""
for (i in 1:nrow(tax.clean)){
  if (tax.clean[i,2] == ""){
    kingdom <- paste(tax.clean[i,1],".unclassified",  sep = "")
    tax.clean[i, 2:7] <- kingdom
  } else if (tax.clean[i,3] == ""){
    phylum <- paste(tax.clean[i,2], ".unclassified", sep = "")
    tax.clean[i, 3:7] <- phylum
  } else if (tax.clean[i,4] == ""){
    class <- paste(tax.clean[i,3], ".unclassified", sep = "")
    tax.clean[i, 4:7] <- class
  } else if (tax.clean[i,5] == ""){
    order <- paste(tax.clean[i,4], ".unclassified", sep = "")
    tax.clean[i, 5:7] <- order
  } else if (tax.clean[i,6] == ""){
    family <- paste(tax.clean[i,5], ".unclassified", sep = "")
    tax.clean[i, 6:7] <- family
  } else if (tax.clean[i,7] == ""){
    tax.clean$Species[i] <- paste(tax.clean$Genus[i],".unclassified",
sep = "")
  }
}

teleo_chordata1<-as.matrix(tax.clean)

teleo.chordata.physeq.cleaned <- phyloseq(otu_table(teleo_nochim_eDNA
.rds, taxa_are_rows = FALSE),
                                          sample_data(metadata),
                                          tax_table(teleo_chordata1)
)


teleo.chordata.physeq.cleaned = subset_samples(teleo.chordata.physeq.
cleaned, Type !="control")

original.chordata.physeq <- as.data.frame(sample_sums(teleo.chordata.
```

```
physeq.cleaned))
original.chordata.physeq$Names <- rownames(original.chordata.physeq)

cleaned.chordata.physeq <- as.data.frame(sample_sums(teleo.chordata.p
hyseq.cleaned))
cleaned.chordata.physeq$Names <- rownames(cleaned.chordata.physeq)

control.chordata.rem.sums <- dplyr::left_join(original.chordata.physe
q,cleaned.chordata.physeq, by="Names")

colnames(control.chordata.rem.sums) <- c("Original","Names","New")
control.chordata.rem.sums <- control.chordata.rem.sums %>%
  mutate(perc = New/Original*100)
```

Subset the data into fish (plus lamprey) only

```
Fish.chordata = subset_taxa(mifish.chordata.physeq.cleaned, Class=="A
ctinopterygii"|Class=="Petromyzontida")
Other.chordata = subset_taxa(mifish.chordata.physeq.cleaned, Class!='
Actinopterygii'|Class!="Petromyzontida")
```

Create data tables for unrarefied data

This code chunk converts read numbers into presence (1) or absence (0) on the unrarefied data. Samples with less than 100 reads are removed as they may be the result of index hopping or contamination from neighbouring samples.

The taxonomy is conglomerated to merge sequences belonging to the same taxa for ease of presentation. There is a code chunk below that provides unconglomerated tables and species lists to allow for the examination of unassigned taxa. Tables are exported as .csv files.

Tables with raw read numbers and the relative abundance of species are also generated to facilitate manual checking of the results in the case of unexpected taxa presence or absence.

```
## Export presence absence matrix for fish ####

Fish.chordata2<-Fish.chordata


fish_OTUx <- tax_glom(Fish.chordata2, taxrank="Species")
fish_OTU = transform_sample_counts(fish_OTUx, function(OTU) ifelse(OT
U > 0, 1, OTU))

s1<-as.data.frame(otu_table(fish_OTU))
s1_t<-t(s1)
s2<-as.data.frame(tax_table(fish_OTU))
```

```
comb<-merge(s1_t,s2, by="row.names")
red_col<-ncol(comb)
comb1<-comb[-c(1,(red_col-6):(red_col-1))]

red_col2<-ncol(comb1)
comb_t = setNames(data.frame(t(comb1[,-red_col2])), comb1[,red_col2])
rows <- row.names(comb_t)
rows

comb_t=comb_t%>%mutate(sample_name=as.factor(rows))


write.csv(comb_t, file.path(path, "teleo_presence_absence_unrarefied_
20200728.csv"))
write.csv(s2,file.path(path, "teleo_species_detected_unrarefied_20200
728.csv"))


s1.x<-as.data.frame(otu_table(fish_OTUx))
s1.x_t<-t(s1.x)
s2.x<-as.data.frame(tax_table(fish_OTUx))

comb.x<-merge(s1.x_t,s2.x, by="row.names")
red_col.x<-ncol(comb.x)
comb1.x<-comb.x[-c(1,(red_col.x-6):(red_col.x-1))]

red_col2.x<-ncol(comb1.x)
comb_t.x = setNames(data.frame(t(comb1.x[,-red_col2.x])), comb1.x[,re
d_col2.x])
rowsx <- row.names(comb_t.x)
rowsx

comb_t.x=comb_t.x%>%mutate(sample_name=as.factor(rowsx))

write.csv(comb_t.x, file.path(path, "teleo_read_numbers_unrarefied_20
200728.csv"))
```

Rarefy the data

Generate rarefaction curves to check the cutoff value for the rarefaction step. Ideally this will be at least 2000, but if possible 8000-12000 is best. This reduces the risk of losing rare ASVs. Rarefaction is necessary to ensure even sampling effort across samples so the relative abundance of DNA from different taxa can be compared. Presence/absence tables using un-rarefied data may provide an indication of rare taxa presence, but the relative abundance of DNA in unrarefied data (e.g. number of reads matching that taxa) cannot be compared.

```
set.seed(100)
```

```
fish.chordata.rarefied = rarefy_even_depth(Fish.chordata, sample.size
= 5500, replace = FALSE, trimOTUs = TRUE, verbose = TRUE)
```

Create data tables for rarefied data

This code chunk converts read numbers into presence (1) or absence (0) on the rarefied data. Samples with less than 100 reads are removed as they may be the result of index hopping or contamination from neighbouring samples.

The taxonomy is conglomerated to merge sequences belonging to the same taxa for ease of presentation. There is a code chunk below that provides unconglomerated tables and species lists to allow for the examination of unassigned taxa. Tables are exported as .csv files.

Tables with raw read numbers and the relative abundance of species are also generated to facilitate manual checking of the results in the case of unexpected taxa presence or absence.

```
## Export presence absence matrix for fish ####


fish_OTU2 <- tax_glom(fish.chordata.rarefied, taxrank="Species")
fish_OTU3 = transform_sample_counts(fish_OTU2, function(OTU) ifelse(O
TU > 0, 1, OTU))

s1.2<-as.data.frame(otu_table(fish_OTU3))
s1.2_t<-t(s1.2)
s2.2<-as.data.frame(tax_table(fish_OTU3))

comb.2<-merge(s1.2_t,s2.2, by="row.names")
red_col.2<-ncol(comb.2)
comb1.2<-comb.2[-c(1,(red_col.2-6):(red_col.2-1))]

red_col2.2<-ncol(comb1.2)
comb_t.2 = setNames(data.frame(t(comb1.2[,-red_col2.2])), comb1.2[,re
d_col2.2])
rows2 <- row.names(comb_t.2)
rows2

comb_t.2=comb_t.2%>%mutate(sample_name=as.factor(rows2))


write.csv(comb_t.2, file.path(path, "teleo_presence_absence_rarefied_
20200728.csv"))
write.csv(s2.2,file.path(path, "teleo_species_detected_rarefied_20200
728.csv"))

s1.3<-as.data.frame(otu_table(fish_OTU2))
s1.3_t<-t(s1.3)
s2.3<-as.data.frame(tax_table(fish_OTU2))
```

```
comb.3<-merge(s1.3_t,s2.3, by="row.names")
red_col.3<-ncol(comb.3)
comb1.3<-comb.3[-c(1,(red_col.3-6):(red_col.3-1))]

red_col2.3<-ncol(comb1.3)
comb_t.3 = setNames(data.frame(t(comb1.3[,-red_col2.3])), comb1.3[,re
d_col2.3])
rows3 <- row.names(comb_t.3)
rows3

comb_t.3=comb_t.3%>%mutate(sample_name=as.factor(rows3))

write.csv(comb_t.3, file.path(path, "teleo_read_numbers_rarefied_2020
0728.csv"))
```

Appendix 7. Reference database for the MiFish primer pair. Sequences have been trimmed to the primer regions of interest with the exception of the non-fish out-groups.

>Animalia;Chordata;Actinopterygii;Trachiniformes;Cheimarrichthyidae;Cheimarrichthys;Cheimarrichthys.fosteri
CACCGCGGTTATACGAGAGGCCCAAGCTGATAGATTTCGGCGTAAAGAGTGGTTAAGGAAGTCCTAAAACTA
AAGCCGAACATCCTCAAAGCTGTTATACGCACCCGAAGACAAGAAGTTCAACCACGAAAGTGG
CTTTATCCCCCTGAACCCACGAAAGCTAAGGCA
>Animalia;Chordata;Actinopterygii;Trachiniformes;Cheimarrichthyidae;Cheimarrichthys;Cheimarrichthys.fosteri
CACCGCGGTTATACGAGAGGCCCAAGCTGATAGATTTCGGCGTAAAGAGTGGTTAAGGAAATCCTAAAACTA
AAGCCGAACATCCTCAAAGCTGTTATACGCACCCGAAGACAAGAAGTTCAACCACGAAAGTGG
CTTTATTCCCCTGAACCCACGAAAGCTAAGGCA
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Neochanna;Neochanna.apoda
CACCGCGGTTATACGAGAGGCTCAAGTAGATAGACATCGGCGTAAAGTGTGGTTAGGGCCTAAAAAACTAAA
GCCAAATACCCCCAAGGCTGTTATACGCGCCCGGAGGGACGAAGCCCTCTCACGAAAGTAGC
TTTATCTACCTCGCCTGAATCCACGACAGCTAAGAAA
>Animalia;Chordata;Actinopterygii;Cypriniformes;Cyprinidae;Scardinius;Scardinius.erythrophthalamus
CACCGCGGTTAAACGAGAGGCCCCAGTTAATAATACACGGCGTAAAGGGTGGTTAAGGAAAGCATAACGATA
AAGCCGAATGGCCCTTTGGCTGTCATACGCTTCTAGGTGTCCGAAGCCCAATATACGAAAGTA
GCTTTAGTAAAGCCCACCTGACCCCACGAAAGCTGAGAAA
>Animalia;Chordata;Actinopterygii;Pleuronectiformes;Pleuronectidae;Rhombosolea;Rhombosolea.retiaria
CACCGCGGTTACACGTGAGACCCAAGATGATAGACTACGGCGTAAAGGGTGGTTAGGGGTAAACAAAAACT
AAAGTCAAACGCTTTAAATGCTGTCAAAAGCGCTCAAAACTATGAAGCCCAACCACGAAAGTG
ACTTTAATAACCCTGACTCCACGAAAGCTGGGGAA
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
CGGCGTAAGAGTGGTTAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAGTA
ATGAAGAACCCCTACGAAAGTGGCTTTAAAACTTCTGACCCCACGAAAGCTAGGAAACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
CGCCGCGGTTATACGAGGGGCTCAAGTTGATAGCCACCGGCGTAAAGAGTGGTTAGGAACCCCAACACTAA
AGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGG
CTTTAAAACTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
GTAACGGGAGCTCAGTGTAGTCACCGGCGTAAGAGTGGTTAGGAACCCCAACACTAAAGCCGAACATCTTCA
GGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGAC
CCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
CGCCGCGTTTACGAGGGTCAGTGATAGCCACCGGCGTAAGAGTGGTTAGGACCCCAACACTAAAGCCGAAC
ATCTTCAGGGCTGTCATACGCACCCGAAGATTGAAGAACCCCTACGAAAGTGGCTTTAAAACT
TCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
CGCGCGTTTCGAGAGCTCAGTGATATCACCGGCGTAAGAGTGGTTAGGAACCCCAACACTAAAGCCGAACA
TCTTCAGGGCTGTCATACGCACCCGAAGATAGAAGAACCCCTACGAAAGTGGCTTTAAAATTT
CTGACCCCACGAAAGCTAGGAAACAAACTGGGATAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
CGCGCGTATACGAGGGCTCAGTTGAAGCCACCGGCGTAAGAGTGGTTAGGAGCCCCAACACTAAAGCCGAA
CATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGGCTTTAAAA
CTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
CGCCGCGGTTATACGAGGGGGCTCAAGTTGATAGCCACCGGCGTAAAGAGTGGTTAGGAACCCCAACACTA
AAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGG
CTTTAAAACTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
GTCGGTAAAACTCGTGCCAGCCGCCGCGGTTATACGAGAGGCTCAAGTTGATAGTCACCGGCGTAAAGAGT
GGTTAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATG
AAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGG
ATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.breviceps
GTCGGTAAAACTCGTGCCAGCCGCCGCGGTTATACGAGGGGGCTCAAGTTGATAGTCACCGGCGTAAAGAG
TGGTTAGGAGCCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATAT
GAAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGG
GATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus

GTCGGTAAAACTCGTGCCAGCCGCCGCGGTATACGAGAGCTCAAGTTGATAGTCACCGGCGTAAGAGTGGT
TAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAG
AACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATT
AGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
GTCGGTAAAACTCGTCCAGCCGCCGCGTTATACGAGAGGTCAAGTTGATAGTCACCGGCGTAAGAGTGGTT
AGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGA
ACCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAG
ATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
GTCGGTAAAACTCGTGCCAGCCGCCGCGTTTACGAGGGGCTCAAGTTGATAGCCACCGGCGTAAGAGTGGT
TAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAG
AACCCCTACGAAAGTGGCTTTAAAACTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATT
AGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
GTCGGTAAAACTCGTGCCAGCnnnnnnnnGAGGGCTCAGTGTAGCCACCGGCGTAAGAGTGGTTAGGAACCC
CAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTAC
GAAAGTGGCTTTAAAACTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCC
ACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
GTCGGTAAAACTCGTGCCAGCCGCCGCGGTTATACGAGAGGCTCAAGTTGATAGTCACCGGCGTAAAGAGT
GGTTAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATG
AAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGG
ATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
GTCGGTAAAACTCGTGCCAGCCGCCGCGGTATACGAGAGGCTCAAGTGATAGTCACCGGCGTAAGAGTGGT
TAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAG
AACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATT
AGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
GTCGGTAAAACTCGTGCCAGCCGCCGCGGTTATACGAGAGGCTCAAGTTGATAGTCACCGGCGTAAGAGTGGTT
AGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGA
ACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTA
GATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
GTCGGTAAAACTCGTGCCAGCCGCCGCGGTTATACGAGAGGGCTCAAGTTGATAGTCACCGGCGTAAAGAG
TGGTTAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATAT
GAAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGG
GATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
GCTCAAGTTGATAGTCACCGGCGTAAAGAGTGGTTAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCT
GTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCAC
GAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.cotidianus
GTCGGTAAAACTCGTGCCAGCCGCCGCGGTTATACGAGAGGCTCAAGTTGATAGTCACCGGCGTAAAGAGT
GGTTAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATG
AAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGG
ATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
GTCGTAAACTGTGCCGCCGCCGCGGTATACGAGGGCTCAAGTGATAGCCACCGCGTAAGAGTGGTTAGGAA
CCCCAACACTAAAGCCGAACTCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCTAC
GAAAGTGGCTTTAAATTCTTACCCCCAAACTAGGAACAAATGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
GTCGGTAAAACTCGTGCCAGCCGCCGCGGTATACGAGGGCTCAAGTTGATAGCCACCGGCGTAAAGAGTGG
TTAGGAACCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAA
GAACCCCTACGAAAGTGGCTTTAAAACTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGAT
AGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.depressiceps
ATAGACATCGGCGTAAGTGTGGTTAGGGTATTAGGGACTAAAGCCGAATATCTTCAAGGCTGTTATACGCAC
CCGAAGGAACGAAGACCCTAAGCGAAAGTAGCTTTATTTATTTAGCCTGAACCCACGACAGCT
ATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.depressiceps

ATAGACATCGGCGTAAGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATATCTCCAAAGCTGTTATACGCAC
CCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGACAGCT
ATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.cobitinis
TATCGGCGTAAGTGTGGTTAGGGTATTAGAAACTAAAGCCGAATTTATCCAAGGCTGTTATACGCACCCGGA
GAAACGAAGACCCTCTGCGAAAGTAGCTTTATATTTAGCCTGAACCCACGACAGCTATGGAAC
AAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.basalis
GGCGTAAGAGTGGTTAGGAGCCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGAT
ATGAAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACT
GGGATTAGATACCCCACTATGCCTAGCCAAAACAAAAGTAGCAACCT
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.gobioides
TGGTTAGGAGCCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCC
CTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATAC
CCCACTATGCCTAGCCA
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.hubbsi
TGGTTAGGAGCCCGAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCC
CTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATAC
CCCACTATGCCTAGCCC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.aff.breviceps
CGCCGCGGTTATACGAGGGCTCAAGTTGATAGTCACGGCGTAAAGAGTGGTTAGGAGCCCCAACACTAAAG
CCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGGCTT
TAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGCCTAG
CCCAAACAAAAGTAGCAACCT
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.aff.breviceps
AGTTGATAGTCACCGGCGTAAGAGTGGTTAGGAGCCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATA
CGCACCCGAAGATATGAAGAACCCCTACGAAGTGGCTTTAAATTTCTGACCCCACGAAAGCTA
GGAACAAACTGGGATTAGATACCCCACTATGCCTAGCCCAAACAAAAGTAGCAACCT
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.breviceps
CAGTTGATAGTCACCGGCGTAAGAGTGGTTAGGAAGCCCAACACTAAAGCCGAACATCTTCACGGCTGTCAT
ACGCACCCGGAGATATGAAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAGAG
CTAGGAAACAAACTGGGATTAGATACCCCACTATGCCTAGCCCAAACAAAAGTAGCAACCT
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.breviceps
TAAGAGTGGTTAGGAGCCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAA
GAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGAT
TAGATACCCCACTATGCCTAGCCCAAACAAAAGTAGCAACCT
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.breviceps
CGTGCCAGCCGTCTGCGGTTATACGAGGGGCTCAAGTTGATAGTCACCGGCGTAAAGAGTGGTTAGGAGCC
CCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTA
CGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCC
CACTATGCCTAGCCCTAAACAAAAGTGCAACCTCACACCT
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
CGCCGCGGTTATACGAGGGGCTCAAGTTGATAGCCACCGGCGTAAAGAGTGGTTAGGAACCCCAACACTAA
AGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGG
CTTTAAAACTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGCCT
AGCCCTAAACAAAAGTGCAACCTCACACCT
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.anomalus
ATAGACATCGGCGTAAGTGTGGTTAGGGTATTAGGGACTAAAGCCGAATATCTTCAAGGCTGTTATACGCAC
CCGAAGGAACGAAGACCCTAAGCGAAAGTAGCTTTATTTATTTAGCCTGAACCCACGACAGCT
ATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.anomalus
GAGAGGACCAAGTAGATAGACATCGGCGTAAGTGTGGTTAGGGTATTAGGGACTAAAGCCGAATATCTTCAA
GGCTGTTATACGCACCCGAAGGAACGAAGACCCTAAGCGAAAGTAGCTTTATTTATTTAGCCT
GAACCCACGACAGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.anomalus
AGACATCGGCGTAAGTGTGGTTAGGGTATTAGGGACTAAAGCCGAATATCTTCAAGGCTGTTATACGCACCC
GAAGGAACGAAGACCCTAAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGACAGCTAT
GGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.divergens.northern
ATATCGGCGTAAGTGTGGTTAGGGCATTAAAAACTAAAGCCGAATATCTCCAAGGCTGTTATACGCACCCGG
AGGAACGAAGACCCTCAGCGAAAGTAGCTTTATTGTTTAGCCTGAACCCACGACAGCTATAAA
ACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.divergens.northern

AGATATCGGCGTAAAGTGTGGTTAGGGCATTAAAAACTAAAGCCGAATATCTCCAAGGCTGTTATACGCACC
CGGAGGAACGAAGACCCTCAGCGAAAGTAGCTTTATTGTTTAGCCTGAACCCACGACAGCTAT
AAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.divergens.northern
AGCCACCGCGGTTATACGAGAGGCCCAAGTAGATAGATATCGGCGTAAAGTGTGGTTAGGGCATTAAAAACT
AAAGCCGAATATCTCCAAGGCTGTTATACGCACCCGGAGGAACGAAGACCCTCAGCGAAAGT
AGCTTTATTGTTTAGCCTGAACCCACGACAGCTATAAAACAAACTGGGATTAGATACCCCACTA
TGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.cobitinis.Waitaki
ATATCGGCGTAAGTGTGGTTAGGGTATTAGAAACTAAAGCCGAATGTGTCCAAGGCTGTTATACGCACCCGG
ATAAACGAAGACCCTCAGCGAAAGTAGCTTTATATTTAGCCTGAACCCACGACAGCTATGAAAC
AAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.cobitinis.Waitaki
CGAGAGGACCAAGTAGATAGATATCGGCGTAAAGTGTGGTTAGGGTATTAGAAACTAAAGCCGAATGTGTCC
AAGGCTGTTATACGCACCCGGATAAACGAAGACCCTCAGCGAAAGTAGCTTTATATTTAGCCT
GAACCCACGACAGCTATGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.prognathus.Waitaki
CGGCGTAAGTGTGGTTAGGGTATAAAATACTAAAGCCGAATACCCCCAAGGCTGTTATACGCACCCGGAGGT
ACGAAGACCCTTAGCGAAAGTAGCTTTATTTAGCTAGCCTGAACCCACGACAGCTATGTAACA
AACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.prognathus.Waitaki
AGCTTTATTTAGCTAGCCTGAACCCACGACAGCTATGTAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.prognathus.Waitaki
AGATAGACATCGGCGTAAGTGTGGTTAGGGTATAAAATACTAAAGCCGAATACCCCCAAGGCTGTTATACGC
ACCCGGAGGTACGAAGACCCTTAGCGAAAGTAGCTTTATTTAGCTAGCCTGAACCCACGACAG
CTATGTAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Teviot
AGACATCGGCGTAAGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATACCTCCAAAGCTGTTATACGCACCC
GGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTTGTCTAGCCTGAACCCACGACAGCTAA
GGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Teviot
AACCCACGACAGCTAAGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Teviot
GTATTAAGGACTAAAGCCGAATACCTCCAAAGCTGTTATACGCACCCGGAGGAACGAAGACCCTTAGCGAAA
GTAGCTTTATTTGTCTAGCCTGAACCCACGACAGCTAAGGAACAAACTGGGATTAGATACCCC
ACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Teviot
GCGTAAAGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATACCTCCAAAGCTGTTATACGCACCCGGAGGAA
CGAAGACCCTTAGCGAAAGTAGCTTTATTTGTCTAGCCTGAACCCACGACAGCTAAGGAACAA
ACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Pomohaka
GTGGTTTAGGGTTATTTAAGAGACTAAAGCCGAATACCTCCAAAGCTGTTATACGCACCCGGAGGAACGAAG
ACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGACAGCTATGGAACAAACTGG
GATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Pomohaka
GCCAGCCACCGCGGTTATACGAGAGGACCAAGTAGATAGACATCGGCGTAAAGTGTGGTTAGGGTATTAAG
GACTAAAGCCGAATACCTCCAAAGCTGTTATACGCACCCGGAGGAACGAAGACCCTTAGCGAA
AGTAGCTTTATTTGTTTAGCCTGAACCCACGACAGCTATGGAACAAACTGGGATTAGATACCCC
ACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Pomohaka
AAGCCGAATACCTCCAAAGCTTGTTATACGCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTT
GTTTAGCCTGAACCCACGACAGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Nevis_cf_G_gollumoides
GACATCGGCGTAAGTGTGGTTAGGGTATTAGGGACTAAAGCCGAATACCTCCAAGGCTGTTATACGCACCCG
GAGGAACGAAGACCCTTGACGAAAGTAGCTTTATTTATTTAGCCTGAACCCACGACAGCTATG
GAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Nevis_cf_G_gollumoides
AGCCACCGCGGTTATACGAGAGGACCAAGTGGATAGACATCGGCGTAAAGTGTGGTTAGGGTATTAGGGAC
TAAAGCCGAATACCTCCAAGGCTGTTATACGCACCCGGAGGAACGAAGACCCTTGACGAAAGT
AGCTTTATTTATTTAGCCTGAACCCACGACAGCTATGGAACAAACTGGGATTAGATACCCCACT
ATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.Nevis_cf_G_gollumoides

TCGGCGTAAGTGTGGTTAGGGTATTAGGGACTAAAGCCGAATACCTCCAAGGCTGTTATACGCACCCGGAG
GAACGAAGACCCTTGACGAAAGTAGCTTTATTTATTTAGCCTGAACCCACGACAGCTATGGAA
CAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.lower.Clutha
AGATAGACATCGGCGTAAGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATACCTCCAAAGCTGTTATACGC
ACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGACAG
CTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.lower.Clutha
ACCGCGGTTATACGAGAGGACCAAGTAGATAGACATCGGCGTAAAGTGTGGTTAGGGTATTAAGGACTAAAG
CCGAATACCTCCAAAGCTGTTATACGCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTT
TATTTGTTTAGCCTGAACCCACGACAGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.lower.Clutha
GTAGATAGACATCGGCGTAAGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATACCTCCAAAGCTGTTATAC
GCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGAC
AGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.lower.Clutha
TAGATAGACATCGGCGTAAAGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATACCTCCAAAGCTGTTATAC
GCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGAC
AGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.argentus
TCGGCGTAAGTGTGGTTAGGGCATTAAAAACTGAAGCCGAATACCTCCAAGGCTGTTATACGCACCCGGGG
GAACGAAGACCCTTAGCGAAAGTAGCTTTATTTAATTCGCCCGAACCCACGACAGCTATGGAA
CAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.argentus
CACCTCTGTTCCACGGTAGATAAGTGGAAGACATCGGCGTAAAGTGTGGTTAGGGCATTAAAAACTGAAGCC
GAATACCTCCAAGGCTGTTATACGCACCCGGGGGAACGAAGACCCTTAGCGAAAGTAGCTTTA
TTTAATTCGCCCGAACCCACGACAGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.postvectis
CACCGCGGTTATACGAGAGGACCAAGTAGATAGAAATCGGCGTAAAGTGTGGTTAGGGTATTAAGGACTAAA
GCCGAATATCTCCAAGGCTGTTATACGCACCCGGAGGAACGAAGCCCCTTAGCGAAAGTAGC
TTTATTTGTTTAGCCTGAACCCACGACAGCTATGGAACAAACTGGGATTAGATACCCCACTATG
C
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.argentus
TCGGCGTAAGTGTGGTTAGGGCATTAAAAACTGAAGCCGAATACCTCCAAGGCTGTTATACGCACCCGGGG
GAACGAAGACCCTTAGCGAAAGTAGCTTTATTTAATTCGCCCGAACCCACGACAGCTATGGAA
CAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.basalis
GGCGTAAGAGTGGTTAGGAGCCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGAT
ATGAAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.breviceps
TAAGAGTGGTTAGGAGCCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAA
GAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGAT
TAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.breviceps
CGTGCCAGCCGTCTGCGGTTATACGAGGGGCTCAAGTTGATAGTCACCGGCGTAAAGAGTGGTTAGGAGCC
CCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTA
CGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCC
CACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.gobioides
TGGTTAGGAGCCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCC
CTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATAC
CCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.hubbsi
TGGTTAGGAGCCCGAACACTAAAGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCC
CTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATAC
CCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.huttoni
CGCCGCGGTTATACGAGGGGCTCAAGTTGATAGCCACCGGCGTAAAGAGTGGTTAGGAACCCCAACACTAA
AGCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGG
CTTTAAAACTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Tripterygiidae;Grahamina;Grahamina.nigripenne

CGCCTTGGTTATACGAGGGGCTCAAGTTGATAGCCACCGGCGTAAAGAGTGGTTAGGAACCCCAACACTAAA
GCCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGGCT
TTAAAACTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Tripterygiidae;Grahamina;Grahamina.nigripenne
TTCGGCGTAAGAGTGGTTAGAACGATATATAACTAAAGCCGAACCTTCTCAGAACTGTTATACGTACTCGAGA
GTAAGAAGCCCCTCAACGAAAGTAGCTTTAACATTTTGACCCCACGAAAGCTGCGAAACAAAC
TGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Tripterygiidae;Grahamina;Grahamina.nigripenne
AACGATATATAACTAAAGCCGAACCTTCTCAGAACTGTTATACGTACTCGAGAGTAAGAAGCCCCTCAACGAA
AGTAGCTTTAACATTTTGACCCCACGAAAGCTGCGAAACAAACTGGGATTAGATACCCCACTAT
GC
>Animalia;Chordata;Actinopterygii;Cypriniformes;Cyprinidae;Tinca;Tinca.tinca
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTAAACGAGAGGCCCTAGTTGATATTACTACGGCGTAAAGGG
TGGTTAAGGAAAGCATTATAATAAAGCCAAATGGCCCCTTGGCCGTCATACGCTTCTAGGTGC
CCGAAGCCCAACCACACGAAAGTAGCTTTAACAAAGCCCACCTGACCCCACGAAAGCTGAGA
AACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cypriniformes;Cyprinidae;Gobio;Gobio.gobio
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTAAACGAGAGGCCCTAGTTGATTTTATCACGGCGTAAAGGG
TGGTTAAGGAAGACAAAACAATAAAGCCGAATGGCCCTTTGGCCGTCATACGCTTCTAGGTGT
CCGAAGCCCAATAGTACGAAAGTAGCTTTAATAAAACCCACCTGACCCCACGAAAGCTAAGAA
ACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cyprinodontiformes;Poeciliidae;Poecilia;Poecilia.reticulata
GCCGGTCAAATTCGTGCCAGCCACCGCGGTTATACGAAAGGCTCAAGTTGATAATCTTCGGCGTAAAGAGTG
GTTAAAAGACATCTTAAACTAAGGCTGAACCACCCCAAAGCTGTCATACGCTACTGGGAGTGT
GAAATACAACCACGAAGGTGGCCTTAAATAATCTTGACCCCACGAAAGCTGTGAAACAAACTG
GGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Salmonidae;Salvelinus;Salvelinus.fontinalis
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCTAGTTGATAAATACCGGCGTAAAGAGT
GGTTACGAAAAAATGTTTAATAAAGCCGAACACCCCCTCAGCCGTCATACGCACCTGGAGGCA
CGAAGACCTACTGCGAAAGCAGCTTTAATTATACCCGAATCCACGACAGCTACGACACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Notocanthiformes;Aguillidae;Anguilla;Anguilla.dieffenbachii
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGGGGCTCAAATTGATATTACACGGCGTAAAGCGT
GATTAAAAAACAAACAAACTAAAGCCAAACACTTCCCATGCTGTCATACGCTACCGGACAAAAC
GAAGCCCCATAACGAAAGTAGCTTTAACACCTTTGAACTCACGACAGTTGAGAAACAAACTGG
GATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Salmonidae;Oncorhynchus;Oncorhynchus.tschawytscha
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCTAGTTGATAACTACCGGCGTAAAGAGT
GGTTATGGAAAAATATTTAATAAAGCCGAACACCCCCTCAGCCGTCATACGCACCTGGGGGCA
CGAAGACCTACTGCGAAAGCAGCTTTAATTACACCTGACCCCACGACAGCTAAGAAACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Salmonidae;Salmo;Salmo.trutta
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGACCCTAGTTGATAACTACCGGCGTAAAGAGT
GGTTACGGAAAAATATTCAATAAAGCCGAACACCCCCTCAGCCGTCATACGCACCTGGGGGCA
CGAAGATCTACTGCGAAAGCAGCTTTAATTATGCCTGAACCCACGACAGCTACGACACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Retropinnidae;Retropinna;Retropinna.retropinna
GCCGGTTAATCTCGTGCCAGCCACCGCGGTTATACGAGTGGCCCAAGTTGAAAGTCGCCGGCGTAAAGAGT
GGTTAGGAAAAGATCAAACTAAAGTTGAATAACCCCTAGGCCGTTGTACGCTCCTGGGGTAAT
GAAAATCTACCACGAAAGTGACTTTACACCTACTTCTGAACCCACGACAACTAAGATACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cyprinodontiformes;Poeciliidae;Gambusia;Gambusia.affinis
GCCGGTTAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCAAGTTGATAAAATACGGCGTAAAGCGT
GGTTAAAAGCCCCCACTAAACTAAGACTAAACCTTTCCAAAGCTGTTATACGCACCCGGAAATA
TGAAACTCAACTACGAAAGTGGCCTTAATTTCCCCTTGACCCCACGAAAGCTGCGAAACAAAC
TGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cypriniformes;Cyprinidae;Scardinius;Scardinius.erythrophthalamus
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTAAACGAGAGGCCCCAGTTAATAATACACGGCGTAAAGGGT
GGTTAAGGAAAGCATAACGATAAAGCCGAATGGCCCTTTGGCTGTCATACGCTTCTAGGTGTC
CGAAGCCCAATATACGAAAGTAGCTTTAGTAAAGCCCACCTGACCCCACGAAAGCTGAGAAAC
AAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Salmonidae;Oncorhynchus;Oncorhynchus.mykiss
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCTAGTTGATAACTACCGGCGTAAAGAGT
GGTTATGGAAAAATATTTAATAAAGCCGAACACCCCCTCAGCCGTCATACGCACCTGGGAGCA

CGAAGACCTACTGCAAAAGCAGCTTTAACTATGCCTGACCCCACGACAGCTAAGAAACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Salmonidae;Oncorhynchus;Oncorhynchus.nerka
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCTAGTTGATAACTACCGGCGTAAAGAGT
GGTTATGGAAAAATATTTAATAAAGCCGAACACCCCCTCAGCCGTCATACGCACCTGGGAGCA
CGAAGACCTACTGCGAAAGCAGCTTTAATTATGCCTGACCCCACGACAGCTAAGAAACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cypriniformes;Cyprinidae;Hypophthalmicthys;Hypophthalmicthys.molitrix
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTAAACGAGAGGCCCTAGTTGATAAAACCACGGCGTAAAGGG
TGGTTAAGGAAAGCAAACAAATTTTAAAGCCAAATGGCCCTTTGGCCGTCATACGCTTCTAGGT
GTCCGAAGCCCAATTACACGAAAGTAGCTTTATTAAAGCCCACCTGACCCCACGAAAGCTGAG
AAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cyprinodontiformes;Poeciliidae;Xiphophorus;Xiphophorus.hellerii
GCCGGTTAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCAAGTTGACAGTCTTCGGCGTAAAGCGT
GGTTAAAGATATACTAAACTAAGGCTAAACTTCCCCAAGGCTGTCATACGCACCCGGAAACAT
GAGACCCGACCACGAAAGTGGCCTTAATACCCCCCCCCTTGACCCCACGAAAGCTATGAAAC
AAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.gollumoides
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGACCAAGTAGATAGACATCGGCGTAAAGTGT
GGTTAGGGTAATAGGGACTAAAGCCGAATACTTCCATGGCTGTTATACGCACCCGGAGGAAC
GAAGACCCTTCACGAAAGTAGCTTTATTAATTTAGCCTGAACCCACGACAGCTATGGAACAAAC
TGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Salmonidae;Salmo;Salmo.salar
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCTAGTTGATAACTACCGGCGTAAAGAGT
GGTTACGGAAAAATATTTAATAAAGCCGAACACCCCCTCAGCCGTCATACGCACCTGGGGACA
CGAAGACCTACTACGAAAGCAGCTTTAATTGTACCTGAACCCACGACAGCTACGACACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Gobiidae;Acentrogobius;Acentrogobius.pflaumii
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCAAGTTGACAGAATTCGGCGTAAAGAGT
GGTTAATGAATATTATACTAAAGCCGAACACCCTCAAGACTGTTATACGTGTTCGAGGGCAGG
AAGCCCTTCAACGAAAGTGGCTTTAATAAGCATGAACCCACGAAAGCTAGGGCACAAACTGGG
ATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Mugiliformes;Mugilidae;Aldrichetta;Aldrichetta.forsteri
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCAAGTTGATAGTCATCGGCGTAAAGAGT
GGTTAAGTTAATCCTAATAAAACTAAAGCCGAACGCCCCCAAAACCGTTATACGTACTCGGAG
GTATGAAGCCCAACTACGAAAGTGGCTTTAAAATACCTGACCCCACGAAAGCTGTGAAACAAA
CTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cypriniformes;Cyprinidae;Leuciscus;Leuciscus.idus
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTAAACGAGAGGCCCTAGTTAATAATACACGGCGTAAAGGGT
GGTTAAGGAAAGCATAACAATAAAGCCGAATGGCCCTTTGGCTGTCATACGCTTCTAGGTGTC
CGAAGCCCAATATACGAAAGTAGCTTTAATAAAGCCCACCTGACCCCACGAAAGCTGAGAAAC
AAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cypriniformes;Cyprinidae;Cyprinus;Cyprinus.carpio
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTAGACGAGAGGCCCTAGTTGATATTACAACGGCGTAAAGGG
TGGTTAAGGATAAACAAAAATAAAGTCAAATGGCCCCTTGGCCGTCATACGCTTCTAGGAGTC
CGAAGCCCTAATACGAAAGTAACTTTAATAAACCCACCTGACCCCACGAAAGCTGAGAAACAA
ACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Percidae;Perca;Perca.fluviatilis
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCAAGTTGATAGACATCGGCGTAAAGCGT
GGTTAAGATTAAAGACAATACTAAAGCCGAACACCTTCAGAGCTGTTATACGCATCCGAAGGTA
AGAAGTTCAACCACGAAAGTGGCTTTATAGCCCCTGAACCCACGAAAGCTACGATACAAACTG
GGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Mugiliformes;Mugilidae;Mugil;Mugil.cephalus
GCCGGTAAATCTCGTGCCAGCCACCGCGGTTATACGAAAGACCCAAGCTGATAGATGCCGGCGTAAAGAGT
GGTTAAGTATTTTGATAGAACTAAAGCCGAACGCCCTCAAGACCGTTATACGTTTCCGAAGGTA
TGAAGCCCAACTACGAAAGTAACTTTAACTATATCTGATTCCACGAAAGCTGCGAAACAAACTG
GGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cyprinodontiformes;Poeciliidae;Poecilia;Poecilia.latipinna
GCCGGTCAAATTCGTGCCAGCCACCGCGGTTATACGAAAGGCTCAAGTTGATAATCTTCGGCGTAAAGCGTG
GTTAAAAGACCTGTTAAACTAAGGCTGAACTCCCCCAAAGCCGTCATACGCTCCCGGGAGCAT
GAAACCCGACCACGAAGGTGGCCTTAAACCCTCTTGACCCCACGAAAGCTGTGAAACAAACTG
GGATTAGATACCCCACTATGC
>Animalia;Chordata;Petromyzontida;Petromyzontiformes;Geotriidae;Geotria;Geotria.australis

GCTGGTAAACCTCGTGCCAGCCACCGCGGTTACACGAGGGGCTCAAGTTGATACCTCCGGCACAAAGCGTG
ATTAAAATACTAACTATTCTATACTATAGAAGCCCCCAATGCCTGCTAGTTGAATAGGTATGCAC
AAATATTTCAACATCGAAAGAATCTATACTAAACAGACTTTATTTGACATCACGAAAGCAAAGCT
ACAAACCGGGATTAGATACCCCGCTATGC
>Animalia;Chordata;Actinopterygii;Notocanthiformes;Aguillidae;Anguilla;Anguilla.reinhardtii
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGGGGCTCAAATTGATATTTCATCGGCGTAAAGCGT
GATTAAAAAATAAACAAACTAAAGCCAAACACTTCCCAAGCTGTCATACGCTACCGGATAAAAC
GAAGCCCCACCACGAAAGTGGCTTTAACACCTTTGAACTCACGACAGTTGAGAAACAAACTGG
GATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Notocanthiformes;Aguillidae;Anguilla;Anguilla.australis
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGGGACTCAAATTGATATTACACGGCGTAAAGCGT
GATTAGAAAACAAATAAACTAAAGCCAAACACTTCCCAAGCTGTCATACGCTACCGGACAAAAC
GAAGCCCTATAACGAAAGTAGCTTTAACACCTTTGAACTCACGACAGTTGAGAAACAAACTGG
GATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.brevipinnis
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGACCAAGTAGATAGACATCGGCGTAAAGTGT
GGTTAGGGTATTAAGGACTAAAGCCGAATATCCCCAAGGCTGTTATACGCACCCGGGGAAAC
GAAGCCCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGACAGCTATGGAACAAA
CTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.maculatus
CCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGGCCAAGTAGATAGACATCGGCGTAAAGTGT
GGTTAGGGATTTGTCAGCTAAAGCCGAATACCTCCAAGGCTGTTATACGCACCCGGAGGTCTG
AAGCCCCCTTACGAAAGTAGCTTTATTACTAGACCTGAACCCACGACAGCCACGAGACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cypriniformes;Cyprinidae;Carassius;Carassius.auratus
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTAGACGAGAGGCCCTAGTTGATATTACAACGGCGTAAAGGG
TGGTTAAGGATAAATAAAAATAAAGTCAAATGGCCCCTTGGCCGTCATACGCTTCTAGGCGTC
CGAAGCCCTAATACGAAAGTAACTTTAATGAACCCACCTGACCCCACGAAAGCTGAGGAACAA
ACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Siluriformes;Ictaluridae;Ameiurus;Ameiurus.nebulosus
GCCGGTAAAATTCGTGCCAGCCACCGCGGTTATACGAAAGGCCCTAGTTGCTAGCCACGGCGTAAAGGGTG
GTTAAGGACAACAACAAATAAAGCTAAAGATCCCCTAAGCCGTCATACGCATTCCGGGGGCAC
GAAGCCCTAACACGAAAGTAGCTTTAAAAAATATACCTGACCCCACGAAAGCTAAGAAACAAAC
TGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Salmonidae;Salvelinus;Salvelinus.namaycush
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGCCCTAGTTGATAACTACCGGCGTAAAGAGT
GGTTACGGAAAAATGTTTAATAAAGCCGAACACCCCCTCAGCCGTCATACGCACCTGGGGGCA
CGAAGACCTACTGCGAAAGCAGCTTTAATTGTACCCGAATCCACGACAGCTACGACACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Cypriniformes;Cyprinidae;Ctenopharyngodon;Ctenopharyngodon.idella
GCCGGTAAAACTCGTGCCAGCCACCGCGGTTAAACGAGAGGCCCCAGTTGATAACACCACGGCGTAAAGGG
TGGTTAAGGAAAGCAAAACAATAAAGCCAAATGGCCCTTTGGCCGTCATACGCTTCTAGGTGT
CCGAAGCCCGATCATACGAAAGTAGCTTTAACAAAGCCCACCTGACCCCACGAAAGCTGAGAA
ACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Neochanna;Neochanna.rekohua
TCGGCGTAAGCGTGGTTAGGGCACAGAAACTAGAGCCAAACACCCCCAAGGCTGTTACACGCACCCGGGG
GAACGAAGCCCTCTCACGAAAGTAGCTTTATCTACCTCGCCTGACCCACGACAGCTAAGAACA
AACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Neochanna;Neochanna.rekohua
CCAAGTAGATAGACATTCGGCGTAAGCGTGGTTAGGGCACAGAAACTAGAGCCAAACACCCCCAAGGCTGT
TACACGCACCCGGGGGAACGAAGCCCTCTCACGAAAGTAGCTTTATCTACCTCGCCTGACCC
ACGACAGCTAAGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Neochanna;Neochanna.burrowsius
GAGTAGATAGACATCGGCGTAAAGAGTGGTTAGGGCACAGAAAACTAAAGCCAAATACCCTCAAAGCTGTTA
TACGCACCCGAGGGGACGAAGCCCTATCGCGAAAGTAGCTTTATCTACCTCGCCTGAACCCA
CGACAGCTAAGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Neochanna;Neochanna.burrowsius
GAGTAGATAGACATCGGCGTAAAGAGTGGTTAGGGCACAGAAAACTAAAGCCAAATACCCTCAAAGCTGTTA
TACGCACCCGAGGGGACGAAGCCCTATCGCGAAAGTAGCTTTATCTACCTCGCCTGAACCCA
CGACAGCTAAGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.breviceps
CAGTTGATAGTCACCGGCGTAAGAGTGGTTAGGAAGCCCAACACTAAAGCCGAACATCTTCACGGCTGTCAT
ACGCACCCGGAGATATGAAGAACCCCTACGAAAGTGGCTTTAAAATTTCTGACCCCACGAGAG
CTAGGAAACAAACTGGGATTAGATACCCCACTATGC

>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.aff.breviceps
CGCCGCGGTTATACGAGGGCTCAAGTTGATAGTCACGGCGTAAAGAGTGGTTAGGAGCCCCAACACTAAAG
CCGAACATCTTCAGGGCTGTCATACGCACCCGAAGATATGAAGAACCCCTACGAAAGTGGCTT
TAAAATTTCTGACCCCACGAAAGCTAGGAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Eleotridae;Gobiomorphus;Gobiomorphus.aff.breviceps
AGTTGATAGTCACCGGCGTAAGAGTGGTTAGGAGCCCCAACACTAAAGCCGAACATCTTCAGGGCTGTCATA
CGCACCCGAAGATATGAAGAACCCCTACGAAGTGGCTTTAAATTTCTGACCCCACGAAAGCTA
GGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.vulgaris
AGTAGATAGACATCGGCGTAAGTGTGGTTAGGGCATTAAGGACTAAAGCCGAATACCCCCAAAGCTGTTATA
CGCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGA
CAGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.vulgaris
AGTAGATAGACATCGGCGTAAGTGTGGTTAGGGCATTAAGGACTAAAGCCGAATACCCCCAAAGCTGTTATA
CGCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGA
CAGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.pullus
AGTAGATAGACATCGGCGTAAGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATATCTTCAAGGCTGTTATAC
GCACCCGAAGGAACGAAGACCCTAAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGAC
AGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.pullus
AGTAGATAGACATCGGCGTAAGTGTGGTTAGGGTATTAGGGACTAAAGCCGAATATCTTCAAGGCTGTTATA
CGCACCCGAAGGAACGAAGACCCTAAGCGAAAGTAGCTTTATTTATTTAGCCTGAACCCACGA
CAGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.prognathus
CAGTAGATAGACATCGGCGTAAGTGTGGTTAGGGTATAAGATACTAAAGCCGAATACCCCCAAGGCTGTTAT
ACGCACCCGGAGGTACGAAGACCCTTAGCGAAAGTAGCTTTATTTAGCTAGCCTGAACCCACG
ACAGCTATGTAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.prognathus
AGCAGTAGATAGACATCGGCGTAAAGTGTGGTTAGGGTATAAAATACTAAAGCCGAATACCCCCAAGGCTGT
TATACGCACCCGGAGGTACGAAGACCCTTAGCGAAAGTAGCTTTATTTAGCTAGCCTGAACCC
ACGACAGCTATGTAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.paucispondylus.Manuherikia
AGATAGACATCGGCGTAAGTGTGGTTAGGGTATTAAGGACTAAAGCCAAATATCTCCAAGGCTGTTATACGC
ACCCGGAGGAACGAAGCCCCCTAGCGAAAGTAGCTTTATAAGTTTAGCCTGAACCCACGACA
GCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.paucispondylus
GTAGATAGACATCGGCGTAAGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATATCTCCAAGGCTGTTATAC
GCACCCGGAGGAGCGAAGCCCCCTAGCGAAAGTAGCTTTATTAGTTTAGCCTGAACCCACGA
CAGCTATGTAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.paucispondylis.Manuherikia
TCGGCGTAAGTGTGGTTAGGGTATTAAGGACTAAAGCCAAATATCTCCAAGCTGTTATACGCACCCGGAGGA
CGAAGCCCCTAGCGAAAGTAGCTTTATAAGTTTAGCCTGACCCACGACAGCTATGGACAACTG
GGATTAGATACCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.northern.flathead
CAGTAGATAGACATCGGCGTAAGTGTGGTTAGGGTATTAAAAACTAAAGCCGAATACCTCCAAAGCTGTTATA
CGCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAATCCACGA
CAGCTATGGGACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.n.sp.Waitaki.alpine
GTCGGTAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGACCAAGTAGATAGACATCGGCGTAAGTGTG
GTTAGGGTATTAAGGACTAAAGCCGATATCTCCAAGGCTGTTATACGCACCCGGAGGAGCGAA
GCCCCTAGCGAAAGTAGCTTTATTAGTTTAGCCTGAACCCACGACAGCTATGTAACAAACTGG
GATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.n.sp.Waitaki.alpine
ATCGGCGTAAGTGTGGTTAGGGTATTAAGGACTAAAGCCGATATCTCCAAGGCTGTTATACGCACCCGGAGG
AGCGAAGCCCCTAGCGAAAGTAGCTTTATTAGTTTAGCCTGAACCCACGACAGCTATGTACAA
ACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.macronasus
CACCGCGGTTATACGAGAGGACCAAGTGGATAGACATCGGCGTAAAGTGTGGTTAGGGTATTGGAGACTAA
AGCCGAATATTTCCAAGGCTGTTATACGCACCCGGAGGAACGAAGCCCCTTAGCGAAAGTAG
CTTTATATGTTTAGCCTGAGCCCACGACAGCTATGAAACAAACTGGGATTAGATACCCCACTAT
GC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.fasciatus

TCGGGTAAAGTGTGGTTAGGGTATTAATAACTGAAGCCGAATACCTCCAAGGCTGTTATACGCGCCCGGGGG
ACGAAGACCCTTAGGAAAGGAGCTTTATTTAATTCGCCTGAACCCACGACAGCTATGGAACAA
ACTGGGATTAGATACCCCACTTTGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.fasciatus
GTCGGTAAAAATCGTGCCAGCCACCGCGGTTATACGAGAGGACCAAGTGGATAGACATCGGCGTAAAGTGT
GGTTAGGGTATTAATAACTGAAGCCGAATACCTCCAAGGCTGTTATACGCGCCCGGGGAACGA
AGACCCTTAGCGAAAGTAGCTTTATTTAATTCGCCTGAACCCACGACAGCTATGGAACAAACTG
GGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.eldoni
AAGTAGATAGACATCGGCGTAAGTGTGGTTAGGGTATTAGGAACTAAAGCCGAATACCTTCAAGGCTGTTATA
CGCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGA
CAGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.eldoni
TCGGTTAAAAACTCGTGCCAGCCACCGCGGTTATACGAGAGGACCAAGTAGATAGACATCGGCGTAAAGTGT
GGTTAGGGTATTAGGAACTAAAGCCGAATACCTTCAAGGCTGTTATACGCACCCGGAGGAACG
AAGACCCTTAGCGAAAGTAGCTTTATTTGTTTAGCCTGAACCCACGACAGCTATGGAACAAACT
GGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.divergens
AGTAGATAGATATCGGCGTAAGTGTGGTTAGGGCATTAAAAACTAAAGCCGAATATCTCCAAGGCTGTTATAC
GCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTGTCTAGCCTGAATCCACGACA
GCTATAAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.divergens
AGCCAGTAGATAGACATCGGCGTAAGTGTGGTTAGGGCATTACTAACTAAAGCCGAATATCTCCAAGGCTGT
TATACGCACCCGGAGGAACGAAGACCCTTAGCGAAAGTAGCTTTATTGTTTAGCCTGAACCCA
CGACAGCTATAAAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.paucispondylus
TCGGCGTAGGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATATCTCCAAGGCTGTTATACGCACCTGGAGG
TACGAAGCCCCCTAGCGAAAGTAGCTTTATAAGTTTAGCCTGAACCCACGACAGCTATGGAAC
AAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Salmoniformes;Galaxiidae;Galaxias;Galaxias.aff.paucispondylus
CCAAGTAGATAGACATCGGCGTAAGGTGTGGTTAGGGTATTAAGGACTAAAGCCGAATATCTCCAAGGCTGT
TATACGCACCCGGAGGTACGAAGCCCCCTAGCGAAAGTAGCTTTATAAGTTTAGCCTGAACCC
ACGACAGCTATGGAACAAACTGGGATTAGATACCCCACTATGC
>Animalia;Chordata;Actinopterygii;Perciformes;Arripidae;Arripis;Arripis.trutta
GAGGCCCAAGTTGACAGACTGCGGCGTAAAGCGTGGTTAAGACAGATCAACAACCTAGACCACCTATAAACC
TAAATATAGGATATCTAATTAAAACTATAGCCGAATGACCTCAAAGCAGTTATACGCATATGAGA
CCACGAAGCCCCCTTACGAAAGTAGCTCTAACCTAATCTGACTCCACGAAAGCTGAGATACAA
ACTGGGATTAGATACCCCATTATGC
>Animalia;Chordata;Actinopterygii;Uranoscopiformes;Cheimarrichthyidae;Cheimarrichthys;Cheimarrichthys.fosteri
GCATCTCCCTTACACCGAGAAGATTATTCGTTAGAGTCGAATCACCCTAACACCCAACAGCTAGCCCCACAG
CCAAAAACAACAATTCAACATAAATACCCCCAAATACACTAAACTAACGTTTAACAAACCATTTT
ACCCCCCTAGTATGGGCGACAGAAAAGGACTAATGGAGCAATAGAAAAAGTACCGCAAGGGA
AAGATGAAAGAGCAATGAAATAACCCAGTAAAGTATAAAAAAGCAGAGATTTTAACTCGTACCT
TTTGCATCATGATTTAGCTAGCAAACTTCAAGCAAAGTGTACTTTAGTTTGATACCCCGAAACTA
AGTGAGCTACTCCAAGGCAGCCTATTAATAGGGCCAACCCGTCTCTGTGGCAAAAGAGTGGG
AAGAACTTTGAGTAGAGGTGACAGACCTACCGAACTTAGTTATAGCTGGTTGCCTAAGAAATG
AATAGAAGTTCAGCCTCACGGCTTCTTTCTTGAAACACACCTTAAAACCCCAAGACACCCAAAG
AAACCGCGAGAGTTAGTCAAAGGAGGTACAGCTCCTTTGAAACAAAATACAACTTTACCAGGA
GGATAAAGATCATA
>Animalia;Chordata;Mammalia;Primates;Hominidae;Homo;Homo.sapiens
GGCCCATAACACTTGGGGGTAGCTAAAGTGAACTGTATCCGACATCTGGTTCCTACTTCAGGGCCATAAAGC
CTAAATAGCCCACACGTTCCCCTTAAATAAGACATCACGATGGATCACAGGTCTATCACCCTAT
TAACCACTCACGGGAGCTCTCCATGCATTTGGTATTTTCGTCTGGGGGGTGTGCACGCGATAG
CATTGCGAGACGCTGGAGCCGGAGCACCCTATGTCGCAGTATCTGTCTTTGATTCCTGCCTCA
TCCTATTATTTATCGCACCTACGTTCAATATTACAGGCGAACATACTTACTAAAGTGTGTTAATT
AATTAATGCTTGTAGGACATAATAATAACAATTGAATGTCTGCACAGCCGCTTTCCACACAGAC
ATCATAACAAAAAATTTCCACCAAACCCCCCTCCCCCCGCTTCTGGCCACAGCACTTAAACACA
TCTCTGCCAAACCCCAAAAACAAAGAACCCTAACACCAGCCTAACCAGATTTCAAATTTTATCT
TTTGGCGGTATGCACTTTTAACAGTCACCCCCCAACTAACACATTATTTTCCCCTCCCACTCCC
ATACTACTAATCTCATCAATACAACCCCCGCCCATCCTACCCAGCACACACACCGCTGCTAAC
CCCATACCCCGAACCAACCAAACCCCAAAGACACCCCCCACAGTTTATGTAGCTTACCTCCTC
AAAGCAATACACTGAAAATGTTTAGACGGGCTCACATCACCCCATAAACAAATAGGTTTGGTCC
TAGCCTTTCTATTAGCTCTTAGTAAGATTACACATGCAAGCATCCCCGTTCCAGTGAGTTCACC
CTCTAAATCACCACGATCAAAAGGGACAAGCATCAAGCACGCAGCAATGCAGCTCAAAACGCT

TAGCCTAGCCACACCCCCACGGGAAACAGCAGTGATTAACCTTTAGCAATAAACGAAAGTTTA
ACTAAGCTATACTAACCCCAGGGTTGGTCAATTTCGTGCCAGCCACCGCGGTCACACGATTAA
CCCAAGT

>Animalia;Chordata;Mammalia;Cetartiodactyla;Bovidae;Bos;Bos.taurus

ACTAGTGATTCTGCAGTCTCACCATCAACCCCCAAAGCTGAAGTTCTATTTAAACTATTCCCTGAACACTATTA
ATATAGTTCCATAAATACAAAGAGCCTTATCAGTATTAAATTTATCAAAAATCCCAATAACTCAA
CACAGAATTTGCACCCTAACCAAATATTACAAACACCACTAGCTAACATAACACGCCCATACAC
AGACCACAGAATGAATTACCCAGGCAAGGGGTAATGTACATAACATTAATGTAATAAAGACATG
ATATGTATATAGTACATTAAATTATATGCCCCATGCATATAAGCAAGTACATGATCCCTATAGTA
GTACATAATACATACAATTATTGACCGTACATAGTACATTATGTCAAATTCATTCTTGATAGCAT
ATCTATTATATATTCCTTACCATTAGATCACGAGCTTAATTACCATGCCGCGTGAAACCAGCAA
CCCGCTAGGCAGGGATCCCTCTTCTCGCTCCGGGCCCATAAATCGGGGGGTCGCTATTCAAT
GAACTTTACCAGACATCTGGTTCTTTCTTCAGGGCCATCTCATCTAAAACGGCCATTATTTCCT
CTTAAATAAGACATCTCGATGGACTAATGGCTAATCAGCCCATGCTTACACATAACTATCCTGT
CATACATTTGGAATTTTTTTATTTTGGGGGATGCTTGGACTCAGCTATGGCCGTCAAAGGCCCT
GACCCGGAGCATCTATTGTAGCTGGACTTAACTGCATCTTGAGCACCAGCATAATGATAAGCG
TGGACATTACAGTCAATGGTCACAGGACATAAATTATATTATATATCCCCCCCTTCATAAAAATT
TCCCCCTTAAATATCTACCACCACTTTTAACAGACTTTTCCCTAAATACTTATTTAAATTTTTCAC
GCTTTCAATACTCAATTTAGCACTCCAAACAAAGTCAATATATAAACGCAGGCCCCCCCCCCCC
GTTGATGTAGCTTAACCCAAAGCAAGGCACTGAAAATGCCTAGATGAGTCTCCCAACTCCATA
AACACATAGGTTTGGTCCCAGCCTTCCTGTTAACTCTTAATAAACTTACACATGCAAGCATCTA
CACCCCAGTGAGAATGCCCTCTAGGTTATTAAAACTAAGAGGAGCTGGCATCAAGCACACACC
CTGTAGCTCACGACGCCTTGCTTAACCACACCCCCACGGGAAACAGCAGTGACAAAAATTAAG
CCATAAACGAAAGTTTGACTAAGTTATATTAATTAGGGTTGGTAAATCTCGTGCCAGCCACCGC
GGTCATACGATTAACCCAAGCTAACAGGAGTACGGCGTAAAACGTGTTAAAGCACCATACCAA
ATAGGGTTAAATTCTAACTAAGCTGTAAAAAGCCATGATTAAAATAAAAATAAATGACGAAAGTG
ACCCTACAATAGCCGACGCACTATAGCTAAGACCCAAACTGGGATTAGATACCCCACTATGCT
TAGCCCTAAACACAGATAATTACATAAACAAAATTATTCGCCAGAGTACTTACTAGCAACAGCTT
AAAACTCAAAGGACTTGGCGGTGCTTTATATCCTTCTAGAGGAGCCTGTTCTATATAATC

>Animalia;Chordata;Mammalia;Cetartiodactyla;Bovidae;Ovis;Ovis.aries

GCTTGGCAAGGATCCCTCTTCTCGCTCCGGGCCCATTAACTGTGGGGGTAGCTATTTAATGAAGTTTAACAG
GCATCTGGTTCTTTCTTCAGGGCCATCTCATCTAAAATCGCCCATTCTTTCCTCTTAAATAAGAC
ATCTCGATGGACTAATGACTAATCAGCCCATGCCTAACATAACTGTGGTGTCATGCATTTGGTA
TTTTTTAATTTTTGGGGATGCTTGGACTCAGCTATGGCCGTCTGAGGCCCCGACCCGGAGCAT
GAATTGTAGCTGGACTTAACTGCATCTTGAGCATCCTCATAATGGTAAGCATGGGCATAATATA
ATTAATGGTCACAGGACATACCTGCTGTATCGTACATTTATATATTCTTTTTCCCCCCTTCCCCT
TAAATATTTATCACCATTTTTAACACGCTTCCCCCTAGATATTAATATAAATTTATCCCGCCCTCA
ATACTCAAATTCGTACTCCAACCGAAGTAAATATATAGGCACCTGGGTCATATACATAACGCAT
AGTTAATGTAGCTTAAACTTAAAGCAAGGCACTGAAAATGCCTAGATGAGTCTACTGACTCCAT
GAACATATAGGTTTGGTCCCAGCCTTCCTGTTAACTTTCAATAGACTTATACATGCAAGCATCC
ACGCCCCGGTGAGTAACGCCCTTCGAATCACACAGGACTAAAAGGAGCAGGTATCAAGCACA
CACTCTTGTAGCTCACAACGCCTTGCTTAACCACACCCCCACGGGAGACAGCAGTAACAAAAA
TTAAGCCATAAACGAAAGTTTGACTAAGCCATATTGACTAGGGTTGGTAAATCTCGTGCCAGCC
ACCGCGGTCATACGATTGACCCAAGCTAACAGGAGTACGGCGTAAAGCGTGTTTAAGCATCAT
ACTAAATAGAGTTAAATTTTAATTAAACTGTTAAAAGCCATAATTATAACAAAAATAAATGACGAA
AGTAACCCTACAATAGCTGATACACCATAGCTAAGACCCAAACTGGGATTAGATACCCCACTAT
GCTTAGCCCTAAACACAAATAATTAT

>Animalia;Chordata;Mammalia;Artiodactyla;Suidae;Sus;Sus.scrofa

GTATCCGGGCCCCGGTGAGAATGCCCTGCAGATCCTAAAGATCAAAAGGAGCAGGTTTCAAGCACACCTTTC
ACGGTAGCTCATACCGCCTTGCTCAACCACGCCCCCAGGGGAAACAGCAGTGATAAAAATTAA
GCCATGAACGAAAGTTTGACTAAGTTATATTAATTAGAGTTGGTAAATCTCGTGCCAGCCACCG
CGGTCGTACAATTAACCCACATTTTTATATCCACGGCGTAAAGAGTGTTTAAGAAAAAAAACCA
CAATAGAGTTAAATTATAACTAAGCTGTAAAAAGCCCTAGTTAAAATAAAATAACCCACGAAAGT
GACTCTAATAATCCTGACACACGATAGCTAGGACCCAAACTGGGATTAGATACCCCACTATGC
CTAGCCCTAAACCCAAATAGTTACATAACAAAACTATTCGCCAGAGTACTACTCGCAACTGCCT
AAAACTCAAAGGACTTGGCGGTGCTTCACATCCACCTAGAGGAGCCTGTTCTATAATCGATAA
ACCCCGATAGACCTTACCAACCCTTGCCAATTCAGCCTATATACCGCCATCTTCAGCAAACCCT
AAAAAGGAACAATAGTAAGCACAATCATAACACATAAAATCGTTAGGTCAAGGTGTAGCTTATG
GGTTGGAAAGAAATGGGCTACATTTTGTACATAAGAATACCCACCATACGAAAGTTTTTATGAA
ACTAAAAACCAAAGGAGGATTTAGCAGTAAATCAAGAATAGAGTGCTTGATTGAATAAGGCCAT
GAAGCACGCACACCCCGCCCGTCACCCTCCTCAAGCATGTAGTAATAAAAATAACCTATATTC
AATTACACGGCCAGCAAGAAGAGTCAAGTCGTAACAAGGTAAGC

>Animalia;Chordata;Mammalia;Diprotodontia;Phalangeridae;Trichosurus;Trichosurus.vulpecula

CYTAGATGGACCATGACAAGTCCCATAAACACAAAGGTTTGGTCCTAGCCTTACTGTTAATTATAATTAAACCT
ACACATGCAAGTTTCCGCTGCCCAGTGAGAATGCCCTCAAAATTATTCATAAATCAACAGGAGC
AGGCATCAGGCACACCACAGGTAGCCCACCACGCCTTGCTTAACCACACCCCCACGGGATAC
AGCAGTGACTAACATTAAGCCATAAACGAAAGTTTGACTAAATCATAATTATTAGGGTTGGTAA
ATTTCGTGCCAGCCACCGCGGCCATACGATTAACCCAAATTAACAGAAAACCGGCGTAAAGTG
TGTTTAAGCACTAACAACCAATAAAGCTAAAATTCAACTAAACTGTAATACGCTATAGTTGACAC
TAAAATACACAACGAAAGTGGCTTTATCTACGCTGAAGACACTATAGCTAAGAAACAAACTGGG
ATTAGAGACCCCACTATGCTTAGCCCTAAACCAAGATAATCCAATAACAATATTATTCGCCAGA
GAACTACTAGCCAGCGCTTAAAACTCAAAGGACTTGGCGGTGCCCTAAACCCACCTAGAGGA
GCCTGTTCTATAATCGATAAACCCCGATAAACCTCACCCATTCTTGCCAATACAGCCTATATAC
CGCCATCGTCAGCTTACCCCATAGGGAAAAAAAGTAAGCAGGATCATAAATCATAAAAACGTTA
GGTCAAGGTGTAGCATATGAATGGGAAAGAAATGGGCTACATTTTCTAAATTAGAATATAACGA
ACTACCTTATGAAACCTAAGATACTGAAGGAGGATTTAGTAGTAAATTAAGAATAGAGAGCTTA
ATTGAAATAGGCAATAGGACGCGCACACACCGCCCGTCACCCTCCTCAATTAAAACCCAAAAC
ATAAATAATAAAACTCAGACAAAAAGAGGAGAAAAGTCGTAACATGGTAAGTGTACTGGAAGGT
GCACTTGGAGTATCAAAATGTAGCTTATAGTAAAGCATTTAGCTTACACCTAAAAGATTTCAGTT
AATACTGACCATTTTGAGCCAATCACAGCCCTAACACCCATCAAAAGAATTATCTCAACAAACA
AAAAAAAACATTTAACCTATCACAGTATAGGAGATAGAACAGATAAATAGGCGCAATAACATTA

>Animalia;Chordata;Aves;Anseriformes;Anatidae;Anas;Anas.platyrhynchos

TAGAGGAGCCTGTTCTGTAATCGATAACCCACGATCAACCCAACCACCCCTTGCCAAAACAGCCTACATACC
GCCGTCGCCAGCCCACCTCGAATGAGAGCACAACAGTGAGCGCAACAGCACCCCCGCTAATA
AGACAGGTCAAGGTATAGCCCATGGGGCGGAAGAAATGGGCTACATTCCCTATACACTAGGG
CAGCACGAAAAGAAGCATGAAACTGCTTCTGGAAGGAGGATTTAGCAGTAAAGTGGGACAATA
GAGCCTACTTTAAGCCGGCCCTAGGGCACGTACATACCGCCCGTCACCCTCCTCACAAGCCA
CACCCCCACATAACTAATACCCTAAACATGCCAAAGATGAGGTAAGTCGTAACAAGGTAAGTG
TACCGGAAGGTGTACTTAGAATACTCAAGACGTAGCTATAACACCCAAAGCACTCAGCTTACG
CCTGAAAGATATCTGCCAAACCAGATCGTCTTGAAGCCTCCCTCTAGCTCAGCCGCCCAAACA
ACGC

>Animalia;Chordata;Aves;Passeriformes;Turdidae;Turdus;Turdus.merula

TGGGATTAGATACCCCACTATGCCTGGCCCTAAATCTTGATGCTCGATATTACCTGAGCATCCGCCCGAGAA
CTACGAGCACAAACGCTTAAAACTCTAAGGACTTGGCGGTGCTCCAAACCCACCTAGAGGAGC
CTGTTCTGTAATCGATGATCCACGATATTACCTGACCATTCCTTGCACGAAACAGCCTATATAC
CGCCGTCGCCAGCCCACCTTTCCTGATAGCCCAACAGTGGACGCAATAGCCTAACCCGCTAG
CAAGACAGGTCAAGGTATAGCCCACGGAATGGAAGCAATGGGCTACATTTTCTAGACTAGAAC
ATACGGATAAGGGTATGAAACTGCCCTTGGAAGGCGGATTTAGCAGTAAAGAGAGACAATTGA
GCCCTCTTTAAGCCGGCTCTGGAGCACGTACATACCGCCCGTCACCCTCCTCATAA

>Animalia;Chordata;Mammalia;Carnivora;Canidae;Canis;Canis.lupus

AGCTGAAATTCTTCTTAAACTATTCCCTGACACCCCTACATTCATATATTGAATCACCCCTACTGTGCCATGTC
AGTATCTCCAGGTAAACCCTTCTCCCCTCCCCTATGTACGTCGTGCATTAATGGTTTGCCCCAT
GCATATAAGCATGTACATAATATTATATCCTTACATAGGACATATTAACTCAATCTCATAATTCAC
TGATCTATCAACAGTAATCAAATGCATATCACTTAGTCCAATAAGGGCTTAATCACCATGCCTC
GAGAAACCATCAACCCTTGCTCGTAATGTCCCTCTTCTCGCTCCGGGCCCATACTAACGTGGG
GGTTACTATCATGAAACTATACCTGGCATCTGGTTCTTACTTCAGGGCCATAACTTTATTTACTC
CAATCCTACTAATTCTCGCAAATGGGACATCTCGATGGACTAATGACTAATCAGCCCATGATCA
CACATAACTGTGGTGTCATGCATCTGGTATCTTTTAATTTTTAGGGGGGGAATCTGCTATCACT
CACCTACGACCGCAACGGCACTAACTCTAACTTATCTTCTGCTCTCAGGGAATATGCCCGTCG
CGGCCCTAATGCAGTCAAATAACTTGTAGCTGGACTTATTCATTATCATTTATCAACTCACGCAT
AAAATCAAGGTGCTATTCAGTCAATGGTTTCAGGACATATAGTTTTAGGGTACACGTACGTACA
CGTACGTACACGTACGTACACGTACGTACACGTACGTACACGTACGTACACGTGCGTACACGT
GCGTACACGTACGTACACGTACGTACACGTGCGTACACGTGCGTACACGTGCGTACACGTGC
GTACACGTACGTACACGTACGTACACGTACGTACACGTGCGTACACGTACGTACACGTACGTA
CACGTACGTACACGTACGTACACGTGCGTACACGTGCGTACACGTGCGTACACGTGCGTACA
CGTGCGTACACGTGCGTACACGTGCGTACACGTACGTACGCGCGTAAGACATTAAGTTAACTT
ATACAAACCCCCCTTACCCCCCGTAAACTCATGTCATCTATTATACACTTATTTATGTCCCGCCA
AACCCCAAAAACAGGACTAAGTGCATACAATACTCACAAGCTTTATTTAAATTATATACAAATGT
ATTGCTACTCTAGTTAACTTAACACAACAGTCTTACACGCATTTGGTCTCGTAGTCTATCTATAG
ATAGCATTCCCTTTTTTTCCCCTCTCATATTTACTATGTATTTTATTTATTACGCACACTACAATT
TCAGTATAAGTTAATGTAGCTTAATTAATAAAGCAAGGCACTGAAAATGCCAAGATGAGTCGCA
CGACTCCATAAACATAAAGGTTTGGTCCTAGCCTTCCTATTAGTTTTTAGTAGACTTACACATGC
AAGCCTCCACGCCCCAGTGAGAATGCCCTTAAAATCACCAGTGATCTAAAGGAGCAGGTATCA
AGCACACTCTTAAGTAGCTCATAACACCTTGCTAAGCCACAC

>Animalia;Chordata;Mammalia;Cetartiodactyla;Cervidae;Cervus;Cervus.elaphus

CATAGGTTTGGTCCCAGCCTTCCTATTAACCCTTAATAGACTTACACATGCAAGCATCCGCACCCCGGTGAAA
ATGCCCTCCAAGTTAATAAGACTAAGAGGAGCTGGTATCAAGCACACATCCGTAGCTCACGAC

ACCTTGCACAGCCACACCCCCACGGGAGACAGCAGTGATAAAAATTAAGCCATAAACGAAAGT
TTGACTAAGCCATATTAATTAGGGTTGGTAAATTTCGTGCCAGCCACCGCGGTCATACGATTAA
CCCAAGTTAATAGGCATACGGCGTAAAGTGTGTTAAAGCACTATACTAAATAAAGTTAAATTCC
AATTAAGCTGTAAAAAGCCATAATTGCAACAAAAATATATAACGAAAGTAACTTTACAACCGCTG
AAACACGATAGCTAGGACCCAAACTGGGATTAGATACCCCACTATGCCTAGCCTTAAACACAA
ATAGTTATGCAAACAAAACTATTCGCCAGAGTACTACCGGCAATAGCTTAAAACTCAAAGGACT
TGGCGGTGCTTTATACCCTTCTAGAGGAGCCTGTTCTATAATCGATAAACCCCGATAAACCTCA
CCATTCCTTGCTAATACAGTCTATATACCGCCATCTTCAGCGAACCCTAAAAAGGTACAAAAGT
AAGCACAATCATAATACATAAAGACGTTAGGTCAAGGTGTAACCTATGGAACGGAAAGAAATG
GGCTACATTTTCTAATCTAAGAAAATCCAACACGAAAGTTATTATGAAATTAATAACCAAAGGAG
GATTTAGCAGTAAACTAAGAATAGAGTGCTTAGTTGAACTAGGCCATGAAGCACGCACACACC
GCCCGTCACCCTCCTCAAGTAGGCACAGTACACTCAAACTTATTTACACGTATTAATCATATGA
GAGGAGACAAGTCGTAACAAGGTAAGCATACTGGAAAGTGTGCTTGGATAAAT

>Animalia;Chordata;Mammalia;Cetartiodactyla;Bovidae;Capra;Capra.hircus

AGCACACATCTTGTAGCTTACAACGCCTCGCTTAACCACACCCCTACGGGAGACAGCAGTGACAAAAATTAA
GCTATAAACGAAAGTTTGACTAAGCCATGTTGACCAGGGTTGGTAAATCTCGTGCCAGCCACC
GCGGTCATACGATTAACCCAAGCTAACAGGAATACGGCGTAAAACGTGTTAAAGCACTACATC
AAATAGAGTTAAATTCTAATTAAACTGTAAAAAGCCATAATTACAACAAAAATAGATGACGAAAG
TAACCCTACTGCAGCTGATACACTATAGCTAAGACCCAAACTGGGATTAGATACCCCACTATGC
TTAGCCCTAAACACAAATAATTACAGAAACAAAATTATTCGCCAGAGTACTACCGGCAACAGCC
CGAAACTCAAAGGACTTGGCG

>Animalia;Chordata;Mammalia;Rodentia;Muridae;Rattus;Rattus.norvegicus

AGGTTTGGTCCTGGCCTTATAATTAATTGGAGGTAAGATTACACATGCAAACATCCATAAACCGGTGTAAAAT
CCCTTAAAGATTTGCCTAAAACTTAAGGAGAGGGCATCAAGCACATAATATAGCTCAAGACGC
CTTGCCTAGCCACACCCCCACGGGACTCAGCAGTGATAAATATTAAGCAATGAACGAAAGTTT
GACTAAGCTAGTACCTCTCAGGGTTGGTAAATTTCGTGCCAGCCACCGCGGTCATACGATTAA
CCCAAACTAATTATTTTCGGCGTAAAACGTGCCAACTATAAATCTCATAATAGAATTAAAATCCA
ACTTATATGTGAAAATTCATTGTTAGGACCTAAGCCCAATAACGAAAGTAATTCTAATCATTTAT
ATAATGCACGATAGCTAAGACCCAAACTGGGATTAGATACCCCACTATGCTTAGCCCTAAACCT
TAATAATTAAACCTACAAAATTATTTGCCAGAGAACTACTAGCTACAGCTTAAAACTCAAAGGAC
TTGGCGGTACTTTATATCCATCTAGAGGAGCCTGTTCTATAATCGATAAACCCCGTTCTACCTT
ACCCCTTCTCGCTAATTCAGCCTATATACCGCCATCTTCAGCAAACCCTAAAAAGGCACTAAAG
TAAGCACAAGAACAAACATAAAAACGTTAGGTCAAGGTGTAGCCAATGAAGCGGAAAGAAATG
GGCTACATTTTCTTTTCCCAGAGAACATTACGAAACCTTTATGAAACTAAAGGACAAAGGAGGA
TTTAGTAGTAAATTAAGAATAGAGAGCTTAATTGAATAGAGCAATGAAGTACGCACACACCGCC
CGTCACCCTCCTCAAATTAGATTGACATTCACATATACATAATTTCACTAACAAATTTATGAGAG
GAGATAAGTCGTAACAAGGTAAGCATACTGGAAAGTGTGCTTGGAATAATCACAGTGTAGCTT
AATCACAAAGCATCTGGCCTACACCCAGAAGAATTCATAAAAAATGAACACTTTGA

>Animalia;Chordata;Aves;Passeriformes;Zosteropidae;Zosterops;Zosterops.lateralis

TGTAGATGCCCTGGGCACCCCTAATCTTAGGTCGACAGGAGCGGGTATCAGGCACACCACTTCCACTGTAG
CCCAAGACGCCTTGCAATTGCCACACCCCCACGGGTATTCAGCAGTAGTTAACATTAAGCAAT
GAGTGTAAACTTGACTTAGTCATAGCAATCCCAAGGGTCGGTAAATCCTGTGCCAGCCACCGC
GGTCATACAGGAGACCCAAATCAACATTATAACGGCGTAAAGCGTGGTCACATGTTATCCAAG
TAGCTAAGATTAAAAAGCAACTGAGCCGTCATAAGCCCAAGATGCGTCATAAGGCCTCTATTCA
AAGAAAATCTTAGACCAACGATCAATTAAAGCCACGAAAGCCAGGACCCAAACTGGGATTAGA
TACCCCACTATGCCTGGCCCTAAATCTTGATGCTCGATCTAACCGGAGCATCCGCCCGAGAAC
TACGAGCACAAACGCTTAAAACTCTAAGGACTTGGCGGTGCTCCAAATCCACCTAGAGGAGCC
TGTTCTGTAATCGATGATCCACGATATACCTGACCATTCCTTGCCCAAAACAGCCTATATACCG
CCGTCTCCAGCCCACCCCGCATGAAGGTTCAACAGTGGACGCAATAGCCGAGTCGCGCTAAT
AAGACAGGTCAAGGTATAGCCTATGGAATGGAAGTAATGGGCTACATTTTCTAGTTTAGAACAC
AACGGCAAAGGCGCATGAAACTGCACCTAGAAGGAGGATTTAGCAGTAAAGAGAGATTAGCG
AGCCCTCTTTAAGCCGGCTCTGGAGCACGTACATACCGCCCGTCGCCCTCCTCAAAAGCGAC
CCCCCCCCCCCCCATACATAATAAGTTTCTCAGCCAAAGAGGAGGTAAGTCGTAACAAGGTAAG
TGTACCGGAAGGTGCACTTAG

>Animalia;Chordata;Actinopterygii;Osmeriformes;Retropinnidae;Prototroctes;Prototroctes.maraena

CCCAAAGGACTTGGCGGTGCCTCATACCCACCTAGAGGAGCCTGTTCTTGAATCGATACTCCCCGTTCAACC
TCACCACCCCTTGTTCAACCCGCCTATATACCGCCGTCGTCAGCTTACCCTGTGAAGGTCTCA
TAGTAAGCAAAATGGGCACAGCCCAGAACGTCAGGTCAAGGTGCAGCGTATGGGGTGGGAAG
AAATGGGCTACATTCCCTAGTTCAGGGTACTACAGATGGGGCTGTGAAACCAGCCCCTGAAG
GTGGATTTAGCAGTAAGAAGGAAATAGAGTGTTCTTCTGAAGCCGGCTCTGAGGCGCGCACA
CACCGC

Table A7.1 Read numbers for each sample in the validation trial throughout the bioinformatic filtering and sequence preparation process. Steps progress from each column left to right across the table, with the sequences present in the Nonchim column carried through to the assignment of taxonomy.

Sample ID	Input	Filtered	DenoisedF	DenoisedR	Merged	Nonchim
200127-1	1280	947	640	610	177	177
200127-10	176946	159584	157719	155874	56828	56506
200127-10a	174904	152449	150608	147972	44708	44304
200127-11	240674	204282	202301	200099	20019	19481
200127-11a	21115	15374	15004	14844	2028	2028
200127-12	129319	117272	115749	114730	35356	34906
200127-13	182726	162385	160482	159314	39424	39168
200127-14	175367	161956	159276	158537	69364	68266
200127-15	223404	197156	194933	193517	32057	32045
200127-16	132148	121001	119196	117948	46962	46455
200127-17	188823	168926	166984	164474	38825	38732
200127-18	147102	134875	133357	132027	39888	39261
200127-19	28365	25942	25203	24935	10969	10928
200127-1a	103215	92296	90807	89796	31858	31674
200127-2	3308	2851	2545	2421	989	989
200127-20	265313	243617	240432	237663	104843	102755
200127-21	63957	2863	930	985	541	518
200127-22	61235	2544	588	656	130	130
200127-2a	120317	108399	105398	103460	35544	35170
200127-3	336615	312600	309461	305710	103831	102101
200127-3a	274928	246769	243877	240587	71183	69828
200127-4	254721	231733	229268	226330	64707	63893
200127-4a	160841	131660	129065	127612	33300	32688
200127-5	217485	199913	197678	195100	70357	69371
200127-5a	172388	155091	153242	150663	50477	50166
200127-6	121815	108713	107269	105774	18334	18310
200127-6a	123155	92923	91501	90683	11071	11071
200127-7	205025	188508	186476	184652	76588	75409
200127-7a	124653	112840	111517	109422	37516	36627
200127-8	259946	240667	238354	235419	85757	84857
200127-8a	226933	205162	202620	199854	67656	67144
200127-9	363089	333063	329326	326591	123294	119223
200127-9a	74123	68683	67645	66739	27026	26824
200205-1	5888	4980	4528	4564	3907	3907
200205-10	76500	2291	219	156	98	98
200205-2	117343	105941	105161	103855	61127	61094
200205-3	197887	185271	183777	181969	124974	124896
200205-4	93164	87963	86889	85831	46766	46407
200205-5	172489	162277	160484	158152	84374	83469

200205-6	255152	243408	241920	238186	234252	228391
200205-7	203576	190300	188008	186244	84747	83495
200205-8	130852	119097	117221	115946	40813	40585
200205-9	192564	176442	174144	172224	75205	74541
200319-07	212752	191828	188947	186924	76971	74817
200319-08	132702	122661	120677	119373	44066	43061
200319-09	114498	104723	103309	102577	35381	35368
200319-10	167027	155608	153588	151624	29212	28792
200319-11	28582	26336	25486	25008	5495	5494
200319-12	105598	89349	88079	87302	7242	7230
200319-13	100816	88221	86330	85761	32770	32659
200319-14	996	713	467	507	157	157
200319-15	33023	29220	28752	28468	13842	13833
200319-16	202324	177344	175844	172915	56375	54374
200319-17	164414	138825	137340	135866	50864	50408
200319-18	252514	216086	213662	209295	112978	103105
200319-19	188946	160307	158745	157292	81	81
200319-20	43241	1403	258	316	30	30
200323-01	77603	70889	69796	68898	11590	11590
200323-02	110929	103608	102098	100725	30397	30392
200323-03	105823	96517	95133	94222	39218	37575
200323-04	100925	92069	90660	89605	27849	27677
200323-05	473	288	120	131	0	0
200323-06	139599	120067	117959	117488	47073	44888
200323-07	270481	254538	252183	251070	240144	208165
200323-08	200318	185128	182248	181912	153649	127727
200323-09	155511	143436	140212	140167	105082	97954
200323-10	155195	142297	138913	139871	128291	120255
200323-11	284095	266195	262507	261369	232828	218544
200323-12	63208	2612	664	514	137	137
200527-1	106908	98295	96292	95337	29214	29008
200527-10	124882	109796	108491	107585	50526	49998
200527-11	169693	142495	141172	139972	83652	83471
200527-12	139490	111127	109829	108693	43564	40762
200527-13	132324	104182	102597	101811	311	311
200527-14	89044	63581	62453	61771	16281	16281
200527-15	106027	79807	78457	77889	9742	9737
200527-16	49780	1605	93	139	38	38
200527-2	226155	197294	193727	192309	32866	32715
200527-3	61950	56603	55095	54592	16007	15960
200527-4	128745	111151	109506	108576	9112	9112
200527-5	137504	117794	116377	115308	21213	21213
200527-6	16738	14852	14369	14245	1663	1663
200527-7	68392	60598	59595	58909	9213	9213

200527-8	121539	102374	100917	100105	6111	6044
200527-9	128932	119337	117679	116808	44477	43981
Clean-b1	910	560	306	251	111	111
Clean-b2	641	425	203	127	52	52
Clean-b3	1908	1306	794	683	139	139

Appendix 8. Cost breakdown.

Capital cost (correct March 2020)

	USD	NZD
Smith Root sampler backpack	$5995	$8961
eDNA telescopic pole	$1630	$2437
Freight	$900	$1345

Consumables cost (correct September 2020)

Item	USD (cost per sample)	NZD
Filters (each)	$15.00	$22.00
Freight (per sample; 100 filter order)	$4.00	$5.97
DNeasy Blood and Tissue DNA extraction kits (per reaction)		$6.67
Qiacube reagents		$3.33
PCR reagents (per reaction)		$5.00
Normalisation plates (per reaction)		$30.87
HTS run (per reaction)		$40.00
Subtotal consumables (per sample)		$113.84

Labour cost (at $155 per hour)

Task	Time required (hours per sample)	Cost
Filtering	0.30	$46.50
DNA extraction	0.06	$9.30
PCR set up	0.04	$6.20
Clean up	0.04	$6.20
Bioinformatics	0.08	$12.40
Subtotal consumables (per sample)		$34.10

Total for one gene (per sample) $147.94

Total cost for a second gene (per sample) as delineated below $94.47

Consumables

Item		NZD
PCR reagents (per reaction)		$5.00
Normalisation plates (per reaction)		$30.87
HTS run (per reaction)		$40.00
Subtotal consumables (per sample)		$75.87

Labour cost

Task	Time required (hours per sample)	Cost
PCR set up	0.04	$6.20
Clean up	0.04	$6.20
Bioinformatics	0.08	$12.40
Subtotal labour (per sample)		$18.60

Appendix 9. Comparison of Wilderlab method.

An alternative method for collecting and processing eDNA for metabarcoding fish communities was developed by Wilderlab (www.wilderlab.co.nz) in 2019. Interest from stakeholders led to the collection of matched samples at a single site: Poorman Valley Stream. The data consisted of:

1. electric fishing results from a 150 m reach immediately upstream of the sample collection site
2. five replicate samples collected using the Wilderlab collection kits, with a total of 500 mL of stream water filtered. These samples were sent to Wilderlab and were processed using the entire Wilderlab pipeline (DNA extraction, PCR amplification, sequencing and bioinformatics)
3. DNA from the five Wilderlab samples was obtained following the Wilderlab extraction and underwent PCR amplification (using the MiFish primers), sequencing and bioinformatic processing according to the methods described in this report
4. five replicate samples were collected using the Backpack sampling method described in this report, with a total sample volume of 5 L. DNA was extracted and the PCR (using MiFish primers), sequencing and bioinformatics were undertaken using the methods described in this report

Table A9.1 Species lists for samples collected from Poorman Valley Stream using different eDNA collection methods or processed using different laboratory and bioinformatic methods

Species detected with electric fishing	Species list from Wilderlab[#]	Species list for DNA obtained from Wilderlab	Species list using backpack sampler method
Shortfin eel	Shortfin eel	Shortfin eel	Shortfin eel
Longfin eel	Longfin eel	Longfin eel	Longfin eel
Upland bully	Upland bully	Common bully	Common bully
Common bully	Common bully	Smelt	Redfin bully
Redfin bully	Redfin bully		Inanga
Inanga	Inanga		Smelt
Smelt	Smelt		Chinook salmon
	Giant bully		
Unidentified eels	Unidentified bully	Unidentified bully	Unidentified bully
Unidentified bullies	Unidentified galaxid		

[#]Note that Wilderlab results for this site are as indicated by Wilderlab on 17/08/2020.

A full comparison of these approaches was outside the scope of the current study. The differences in species lists among methods raises questions about which parts of the process are influencing species detections. As the Wilderlab method and the methods developed in this project differ in every component of the process (eDNA collection method, volume filtered, DNA extraction method, primer choice and bioinformatic algorithms), it is not possible to infer from these results what the effects would be of interchanging any

components of these methods. It does, however, appear that using the Wilderlab eDNA collection method combined with this project method's laboratory and bioinformatic approach yielded fewer species detected. This result reinforces the need to use the entire pipeline for each method if simple overall method comparisons are desired. To understand why methods might lead to different results, full comparisons for each component are required.