Summer 8-22-2021

Phylogenomics and Population History of Cichlid and Live-bearing Fish Species in Lowland Neotropical Rivers

Konrad Taube
DePaul University, taube2@illinois.edu

Phylogenomics and Population History of Cichlid and Live-bearing Fish Species in Lowland Neotropical Rivers

A Thesis Presented in

Partial Fulfillment of

The Requirements for the Degree of

Masters of Science

August 2021

By:

Konrad Taube

Thesis Advisors: Caleb D. McMahan, Ph.D & Windsor A. Aguirre, Ph.D

Department of Biological Sciences

College of Science and Health

DePaul University

Chicago, Illinois

Table of Contents

Acknowledgements:

Throughout the writing of this thesis I have received a tremendous amount of support and assistance from so many people. I will do my best to recognize them here.

I would first like to thank my advisors, Dr. Caleb McMahan and Dr. Windsor Aguirre, whose incredible expertise was invaluable in formulating the research questions and methodology. The time and energy spent in discussing this project pushed me to elevate my work to a higher level and show me that my continued growth after graduating from DePaul is both essential and necessary. I would also like to thank Dr. Bystriansky for serving on my committee.

I would like to thank the staff from the Grainger Bioinformatics Center at the Field Museum of Natural History and especially the fishes staff (Susan Mochel, Kevin Swagel) at the museum for their wonderful assistance and collaboration. I would particularly like to single out Dr. Felix Grewe, Dr. Kevin Feldheim, and Isabel Distefano. Each of you were immensely helpful whenever I needed help with my work, and I am so grateful for that.

I would also like to thank the members of my cohort, the Devil Dawgs. Your positive energy and critiques of my work early on helped me to grow in my scientific writing.

I would like to thank my family and loved ones (Mom, Dad, Mara, Flo, Haley) for their unending love and support. You are always there for me. Finally, I could not have completed this thesis without the support of my best friends Dewey and Chichi, who provided meaningful discussions as well as needed happy distractions to rest my mind outside of my research.

Abstract:

Within Middle America, cichlids and poeciliids account for more than half of the fish biodiversity. This richness in fish fauna highlights the complexity of Middle American biogeography: no other continental area on earth contains within its range the unparalleled abundance of secondary freshwater fish species (fish that can tolerate both saltwater and freshwater). Research into the biogeography of widely distributed Middle American freshwater fish is essential to understanding this unique region.

Three species of freshwater fishes (*Belonesox belizanus* – Pike killifish, *Vieja maculicauda* – Black belt cichlid, and *Gambusia nicaraguensis* – Nicaraguan mosquitofish) are widely distributed across rivers on the Caribbean slope of Central America (Matamoros *et al*., 2014). *Belonesox belizanus* and *G. nicaraguensis* are poecilids (live-bearing fish), while *V. maculicauda* is a cichlid (a diverse family of fishes primarily found in Africa, South America, and Central America). The overlapping distributions of these species allow for a comparative population genomics approach to understand their biogeographic history and evolution. Past research used individual loci to assess general phylogeographic patterns with little structure detected within each species; however, these data lacked power to properly test hypotheses of population subdivision, gene flow, and recent expansion. Greater genomic coverage and an increase in sample sizes (geographic coverage and number of individuals) are essential for the objective of this proposed research: to test hypotheses of biogeographic and evolutionary patterns of these three species across their Middle American distribution.

Our results using the mitochondrial COI gene suggest four clades of *G. nicaraguensis*, while more comprehensive sampling using genomic data supports only three populations. Two populations were recovered for both *B. belizanus* and *V. maculicauda* using genomic data. Divergence among populations was associated with geographic breaks for the two poeciliids although the location of the geographic breaks differed between species. The two populations detected for *V. maculicauda* were highly divergent genetically but sympatric. This study gives insight into the historical biogeography of the region, showing that population structure is complex and varies across widespread species.

Chapter 1

Review of the Literature:

I want to begin this chapter with a question: why study widespread species? We study widespread species – species whose habitat extends across a large geographic range, coinciding with geological features like mountain ranges, drainages, geological blocks – because these species can provide insights relating to different biological and population level patterns that species with restricted ranges cannot. This can be harder to study – research on widespread species require many samples across a wider geographic area – but it is necessary in order to understand the biogeographic history of not only the species being studied, but the region itself.

Geographical distributions of a species are not random. There are patterns of species distribution that can be shared across many species or be species-specific. The goal of this thesis was to add to the growing body of knowledge of Middle American biogeography by learning what those patterns are. Specifically, I use RADSeq, a method of reduced-representation genomic sequencing, to conduct a phylogeographic analysis of three widespread Neotropical fish species co-distributed along the Atlantic coast of Middle America: two poeciliids, *Gambusia nicaraguensis* and *Belenesox belizanus,* and one cichlid, *Vieja maculicauda*. The results of my research are presented in Chapter 2.

In this chapter, I review the literature on basic topics related to my thesis research to provide a broader context for my research. These topics include: Biodiversity, The Middle America, The Great American Interchange, Myer's Hypothesis, The Freshwater Fishes of the Neotropics highlighting the Cichliformes and Cyprinodontiformes, Phylogeography,

Mitochondrial DNA analysis, restriction site associated DNA sequencing (RADSeq), and the Bioinformatics tools that I am using to analyze my data.

Biodiversity:

Biological diversity (biodiversity) is a broad term that describes living organisms, both macro- and microscopic, and the ecosystems they are a part of (Manokaran, 1992). Biodiversity can be distinguished from genetic diversity, species diversity, and ecosystem diversity, in that these three terms are different levels that differ in scale, from the genetic level (genes, nucleotides, etc.), to species (interspecies and intraspecies), to the level of ecosystem (community of organisms and their environment). The primary force behind decreases in biodiversity is habitat loss (Manokaran, 1992) in the form of deforestation and other human activities, like agriculture, mining, human population settlements, *et cetera* (Chapin et al., 2000).

In order to understand biodiversity, it is necessary to also understand phylogenies and how diversity is organized. A phylogeny shows evolutionary relationships and histories of organisms by compiling these organisms into groups, or clades. In constructing phylogenies, morphological, physiological, biological, behavioral, or molecular characteristics can be used to show evolutionary history of organisms (Malabarba & Malabarba, 2020).

Due to under-sampling in many regions throughout Central America and the Neotropics, research in these areas is essential to both correct this knowledge deficit and better understand the evolution and biogeographic history of a region, such as the freshwaters of the neotropics. Freshwaters of the Neotropics include 20-25% of all world fish diversity, currently including

more than 6000 described species with final estimates varying between 8 and 9 thousand species. This incredible density of species diversity is contained within under 0.003% of available water resources on the planet (Malabarba & Malabarba, 2020). Actinopterygian fish, such as the three species in this thesis, are the richest group among vertebrates, corresponding to approximately half the number of species of animals with backbones (Malabarba & Malabarba, 2019).

A consequence of this under-sampling is that many undescribed species may become extinct without our knowing of their existence. Myers et al. (2000) suggested that studies pertaining to biodiversity and conservation would benefit by focusing on areas where positive impact is maximized, and identified these areas with the term `biodiversity hotspots`. One such area is the Caribbean slope of Middle America, the region relevant to this thesis.

Middle America:

The focal region of this research encompasses Central America, including the Caribbean islands or Greater Antilles, also referred to as Middle America (Winker, 2011). Central America has been recognized by prior research as a politically defined subregion of Middle America that includes the following seven Central American countries: Panama, Costa Rica, Nicaragua, Honduras, El Salvador, Belize, and Guatemala. Middle America begins from the Panama–Colombia border and extends northwest to the Mexico–Guatemala border and Mexico–Belize border, and often including southern Mexico (Matamoros *et al*., 2014). Since political borders are not necessarily congruent with biogeographic patterns and distributions, Middle America is a more accurate term than Central America.

The Great American Interchange

The Great American Interchange is a term used to describe the exchange of animals between North and South America through the formation of the Central American land bridge about 3.5 million years ago (Webb, 2006). It had a profound effect on the dispersal, exchange, and evolution of mammals, reptiles, amphibians, birds, and fish in the region. This land bridge is located between Nicaragua and northern Colombia, and connects the continents of North America and South America. This event, which took place over the course of millions of years, demonstrates the combined effects of dispersal, interspecific interaction, extinction, and evolution on biodiversity (Brown and Lomolino 1998).

However, research by Montes et al. (2015) suggests that the land bridge is at least 13-15 million years old, a far older estimate than 3.5 million years, suggesting that species dispersal and evolution have occurred for far longer than previous estimates.

Myers' hypothesis:

Myers (1966) hypothesized that the Central American ichthyofauna is unique in the Americas, as a result of Plio-Pleistocene tectonics and the rise of the Isthmus of Panama. It was suggested that secondary freshwater fish like cichlids and poeciliids dispersed into Middle America much earlier than their primary freshwater counterparts, as a part of something called the Great American Biotic Interchange, and Myers hypothesized that the geological changes that took place millions of years ago is responsible for why we see this pattern. Myers's (1966) hypothesis is prescient given the dearth of biological and geological information available at the time. Modern phylogenetic systematics – the field of biology concerning the reconstruction of

evolutionary history and relationships among organisms – was still then in its infancy. Since then, the body of knowledge surrounding Central American biodiversity and geological history has increased, and hypotheses of the systematic relationships are now available for many groups of freshwater fishes of this region. The overall understanding of species distribution across Central America has similarly improved. Ichthyological investigations have generated an exponential increase in museum holdings from the region, especially in the Honduran and Nicaraguan Mosquitia (Caribbean coastal plains) that were largely inaccessible in decades prior (Miller, 1966).

The Freshwater Fishes of the Neotropics:

The freshwaters of the Neotropics include roughly 20%–25% of all world fish diversity, with some estimates varying between 8000 and 9000 species (Reis et al., 2016). This incredible level of biodiversity is contained within under than 0.003% of the available water resources on the planet, the Neotropical freshwaters (Malabarba et al., 1998). Many of the fish in the Neotropics belong to two Orders: Cyprinodontiformes (about 13%) and Cichliformes (about 9%). Interestingly, while cichlids and cyprinodonts dominate the diversity of Middle America, tetras, knifefish, and catfishes dominate the diversity in South American freshwaters. Myers (1966) suggested that secondary freshwater fish like cichlids and poeciliids dispersed into Middle America much earlier than their “primary” freshwater counterparts (Matamoros et al., 2014).

Cichliformes:

The family Cichlidae, Order Cichliformes, contains nearly 1900 species (Kullander, 1998), 95 of which have been recognized within isthmian Central America (Family Cichlidae – Cichlids, 2012). The cichlid included in this study, the black-belt cichlid, *Vieja maculicauda*, was selected because it is common in this region of interest is widely distributed across the lowland reaches of rivers in the Atlantic slope of Central America from Belize south to the Rio Chagres drainage in Panama (McMahan *et al*., 2017). The distribution of this cichlid species is illustrated in Figure 1 below:

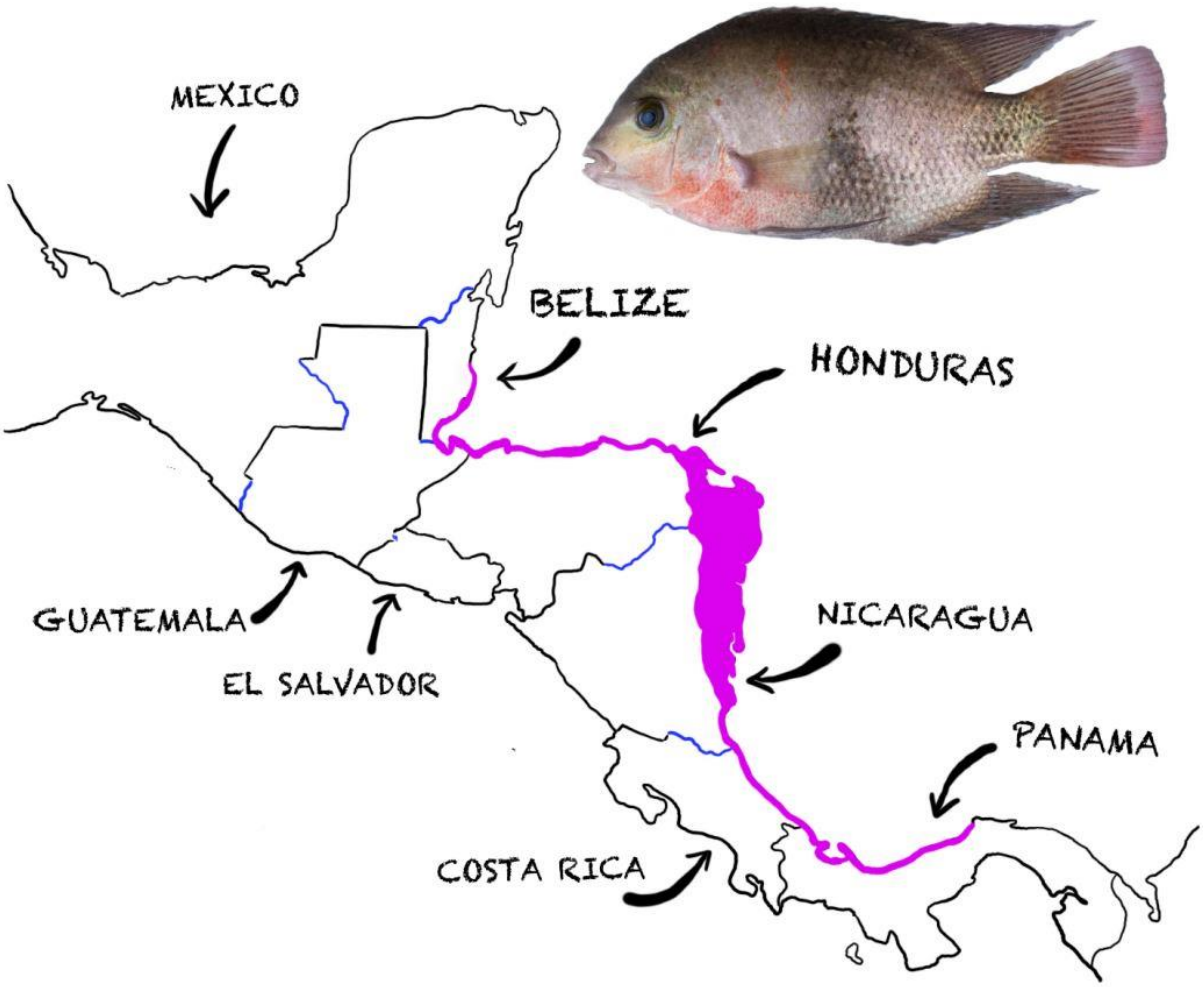


Figure 1: Illustration by Taube (2020) showing the respective distributions of *V. maculicauda*. Adapted from McMahan *et al*. (2017).

Cyprinodontiformes:

The viviparous Family Poeciliidae, Order Cyprinodontiformes, includes more than 20 genera and more than 200 species (Lucinda, 2003). Poeciliids may occur in both fresh and brackish waters of North, Central, and South America, and are found in lakes, rivers, streams, and estuaries (Malabarba *et al*., 1998). Although typically small in size, poeciliids are extremely important ecologically, both as predator and prey species. The poeciliids included in this study, the pike killifish, *Belonesox belizanus* and the Nicaraguan mosquitofish, *Gambusia nicaraguensis*, were also selected because of their wide and overlapping lowland Central American distributions. *Belonesox belizanus* is a predatory poeciliid species that was described by Kner (1860). *Belonesox* is a monotypic genus and is the largest poeciliid, reaching a maximum length of 200 mm (Bussing, 1998). *Gambusia nicaraguensis*, is a planktivorous poeciliid species first described by Günther (1864). *Gambusia nicaraguensis* is a poorly understood species, having been sampled only on three occasions in the mouth of slow-moving streams (Bussing, 1998). The distributions of both poecilid species are illustrated in Figure 2 and Figure 3.

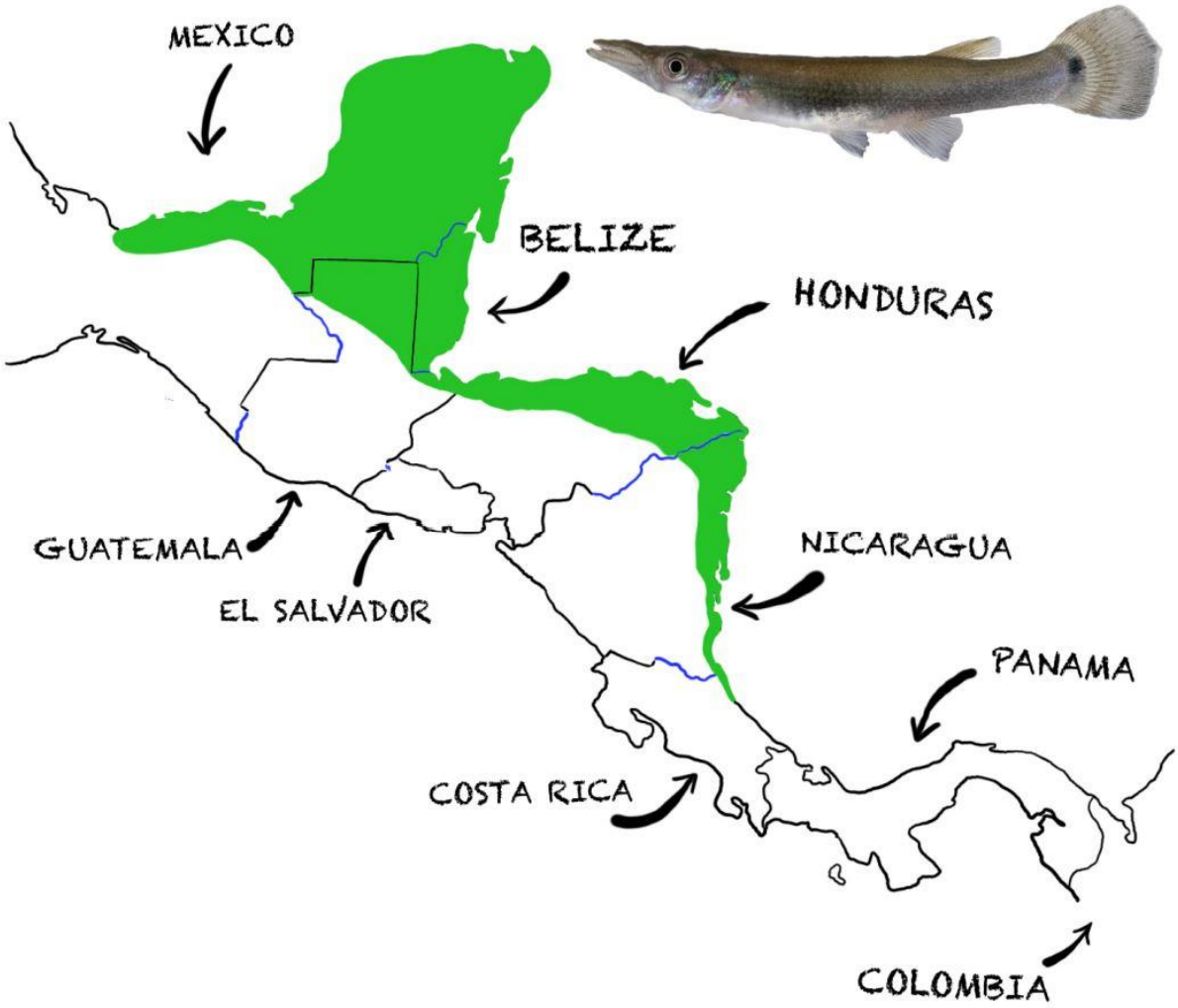


Figure 2: Illustration by Taube (2020) of the distribution of *B. belizanus*. Adapted from Bussing (1998) and Günther (1864).

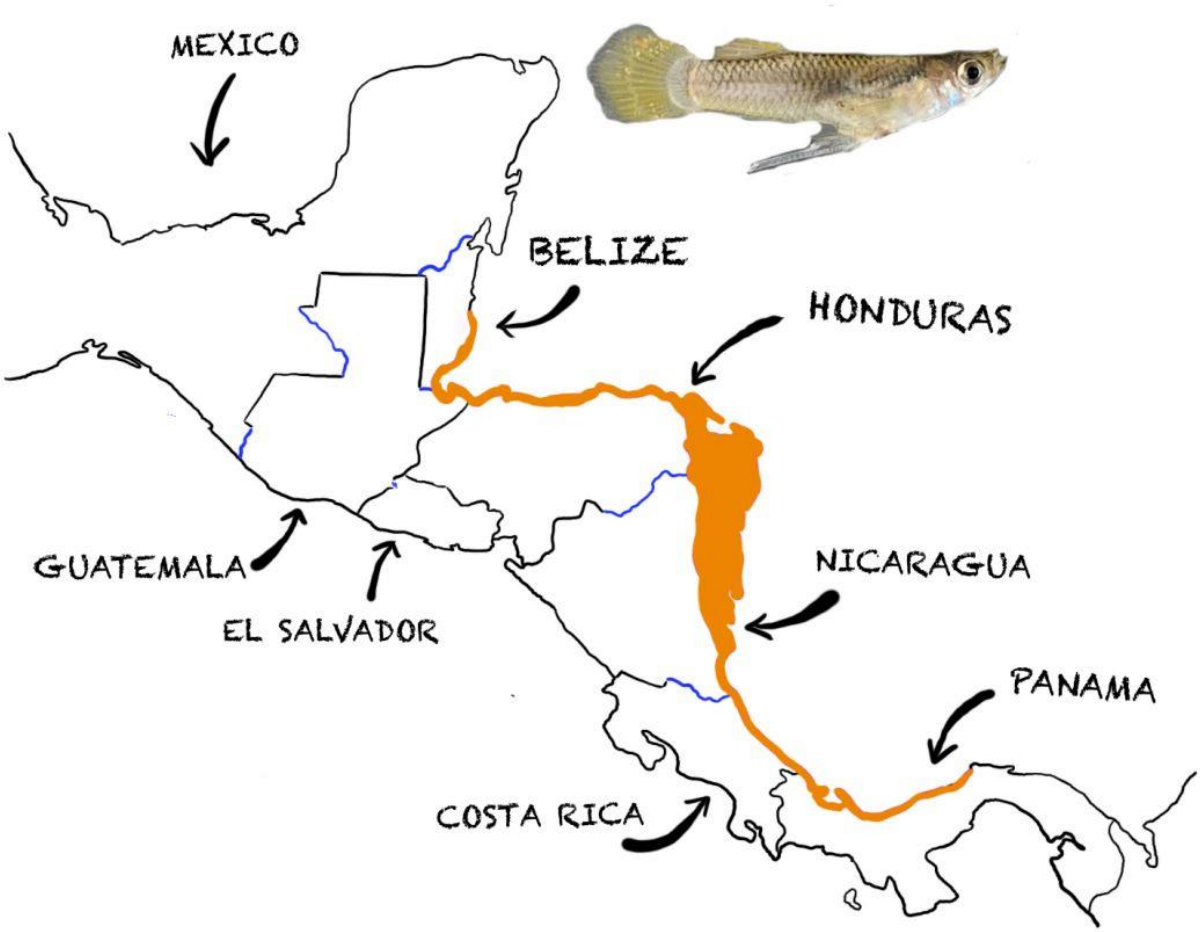


Figure 3: Illustration by Taube (2020) of the distribution of *G. nicaraguensis*. Adapted from Bussing (1998) and Günther (1864).

Phylogeography

As described by Emerson and Hewitt (2005), phylogeography is a "field that analyses the geographical distribution of genealogical lineages". This multidisciplinary field seeks to understand the contemporary distributions of taxa in the context of intrinsic biological and extrinsic geological and climatic factors (Bermingham and Martin, 1998). The spatial relationships of such genealogies may be displayed geographically and analyzed to deduce the evolutionary history of populations, subspecies and species (Emerson and Hewitt, 2005).

Studying the various processes that may influence current distributions of organisms may provide insights as to how a given species responds to environmental change. The ability to analyze DNA sequences – in particular, mitochondrial DNA – has been an essential innovation within this field Next Generation Sequencing in conjunction with various biological, geological, ecological software analytics have provided researchers with the ability to assess the distribution of species diversity, and to test hypotheses as to how this diversity may have arisen.

Mitochondrial DNA (mDNA or mtDNA)

Analysis of mitochondrial DNA (mtDNA) is used to characterize phylogenetic relationships among individuals in order to study biological diversity (Avise *et al*., 1987). Historically, it has been one of the most important tools used to infer relationships among species and populations and it continues to be widely used today, despite the increase in genomic data. Mitochondrial DNA has many advantages as a molecular marker for population genetics studies. For example, Avise *et al*. (1987) wrote that mtDNA of higher animals are

distinctive, yet ubiquitously distributed, are easy to isolate and assay, have a simple genetic structure, exhibit a straightforward mode of transmission, and evolve at a rapid pace such that new character states commonly arise within the lifespan of the species.

Additionally, and perhaps most importantly, mtDNA is maternally inherited, meaning that mtDNA mutations that arise in individuals are not recombined during sexual reproduction (Avise *et al*., 1987). Due to the widespread use of mtDNA in population genetics studies, there is a large database of mtDNA sequences for most animals including freshwater fishes that can be used in comparative analyses. The mitochondrial cytochrome oxidase I (COI) gene, also known as the DNA barcoding gene, is likely the most sequenced gene across animals. This gene encodes a protein that forms a large subunit of the cytochrome c oxidase complex and has an essential role in cellular respiration.

Restriction-site associated DNA Sequencing (RADSeq)

Restriction-site associated DNA Sequencing (RADSeq) is a procedure developed by Baird *et al*. (2008). RADSeq utilizes single nucleotide polymorphisms (SNPs) – the most abundant genomic marker – to study areas of inheritance across a target genome (Baird *et al*., 2008). Using restriction enzymes to cut DNA molecules at target sites, RADSeq's reduced representation sequencing approach targets a subset of the genome to provide a cost-effective procedure for SNP discovery and genotyping.

Further procedures have been developed from RADSeq, such as double digest RADSeq (ddRADSeq) (Peterson *et al*., 2012). A ddRADSeq protocol uses two restriction enzymes (instead of a single restriction enzyme, as in standard RADSeq protocols) that cut the sample DNA at

more locations than a standard RADSeq protocol, giving even greater coverage of each specimen's genome; additionally, ddRADSeq does not require a reference genome for the multiplexing stage (the amplification stage of targeted fragments) of RADSeq data analysis (Figure 4) (Peterson *et al*., 2012).

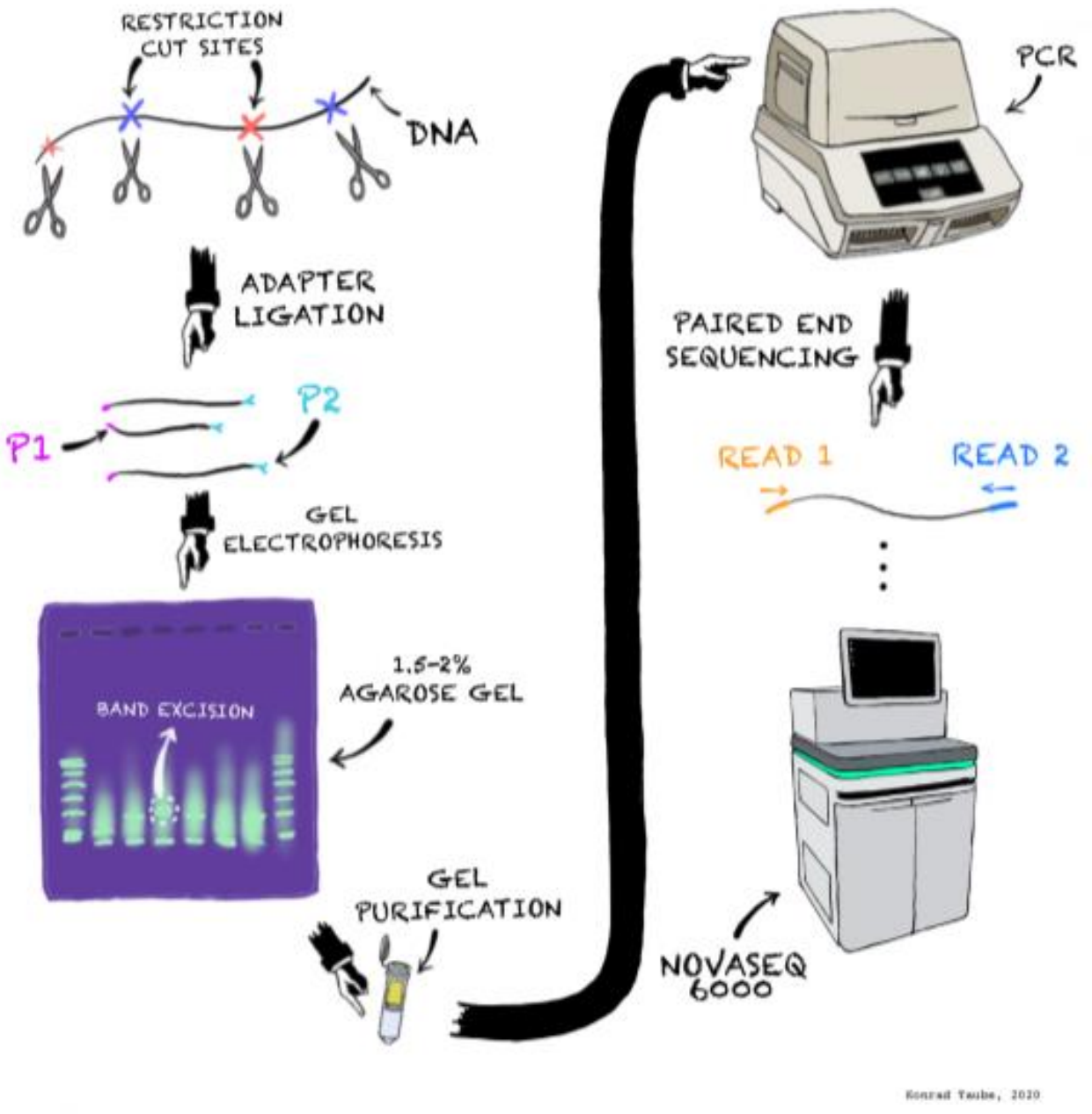


Figure 4: The process of ddRADSeq. The first step is a restriction enzyme double digest, followed by the addition of two adapters, named P1 and P2 in the figure. After gel electrophoresis is the size selection, band excision, and PCR amplification of the targeted sample. The final step in the figure is sequencing DNA on a NovaSeq 6000 (Illumina). Illustration by Taube (2020).

Bioinformatics:

Analyzing complex genetic and genomic data is at the heart of bioinformatics. Working with genomic data for these analyses will typically involve enormous datasets. It is necessary to use many different programs to prepare, process, and analyze this data. Bioinformatics is an ever-evolving field, with programs, pipelines, and protocols to analyze many different aspects of these huge datafiles. Below I present a brief explanation of some of the programs I have used for the completion of this thesis.

ipyrad:

Sequence data analysis will be performed first using the `ipyrad` (Eaton & Overcast, 2020) software, which can be used to identify SNPs from large datasets, generating summary statistics for population genetics. There are 28 parameters to the `ipyrad` program and are described in the "Supplementary files" section of this thesis, adapted from the `ipyrad` website (2019). Broadly speaking, `ipyrad` first processes the short-read DNA sequences, then constructs loci, catalogs the loci, and matches against existing DNA catalogs where they exist. Following these steps, the contigs (contiguous regions of DNA segments comprising a consensus region of DNA) are assembled together.

After analysis in `ipyrad`, the data file is reformatted and exported to other programs, concluding with `STRUCTURE` v2.3.4 (Pritchard et al., 2000). `STRUCTURE` is used to infer distinct population structure and assigning sample individuals to populations, and can be applied to SNP data. Using a maximum likelihood model, `STRUCTURE` constructs a probability distribution of how many populations it detects in a dataset, assigning individuals to groups

based on their sequence composition. This chain of input-output processing data is referred to as a pipeline, where the output of one function becomes the input of the following function (Figure 5).

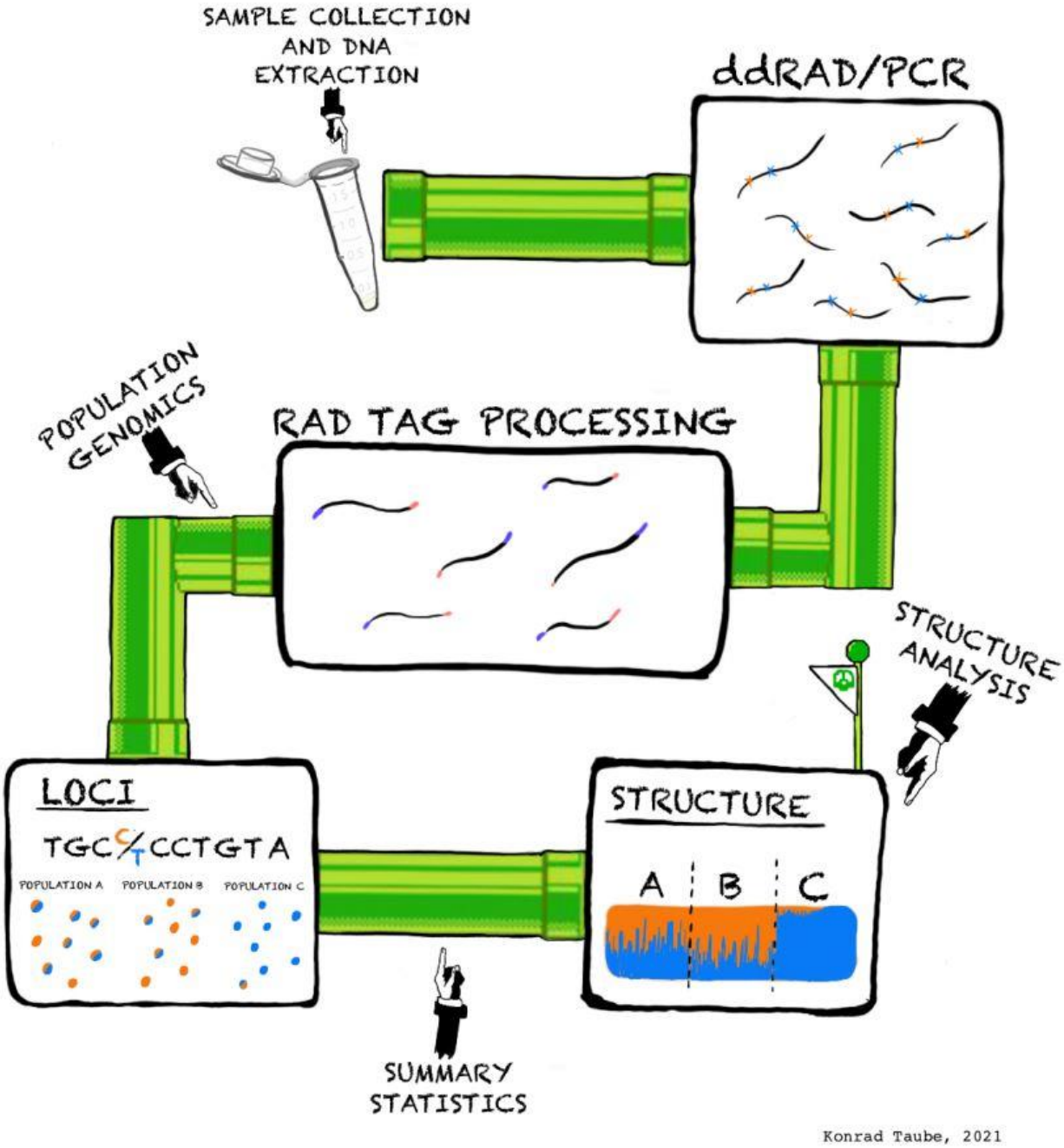


Figure 5: Process of a RADSeq experiment pipeline. Sample DNA is prepared and sequenced, and population genomic data is processed through a sequence of steps via ipyrad, concluding with a structural analysis barplot of the example population data. Illustration by Taube (2021).

The other programs used in the completion of this thesis are named and described below, with links to the program website in parenthesis. Also included are programs used in Bayesian analysis for the purpose of constructing trees and phylogenies using mitochondrial sequence data.

Supplementary Programs:

The programs described below were used in analyzing population structure using SNP data and in the analysis of mitochondrial sequence data for the purpose of phylogenetic inference. `FastQC` (Andrews, 2010), `PGDSpider` (Lischer & Excoffier, 2012), `Plink` (Purcell et al. 2007), and `VCFtools` (Danecek et al., 2011) were used for analyzing the former, while `MEGA` (v.6) (Tamura et al., 2013) was used for the latter. Selection of the optimal value for K was done using `Structure Harvester` (Earl & von Holdt, 2012).

Program 1: FastQC (`www.bioinformatics.babraham.ac.uk/projects/fastqc/`):

`FastQC` is a program that allows researchers the ability to conduct preliminary quality control checks on raw sequence data from high throughput sequencing. It provides analyses which give a quick impression of your data, and can be particularly useful to determine if the

data has any problems that may need to be addressed before continuing further analyses. Some of the functions of FastQC are providing a precursory overview to tell you in which areas there may be problems, presenting summary graphs and tables to quickly assess your data, and an export of these results to an HTML report, and offering an offline operation to allow generation of reports without running the interactive online application.

Program 2: PGDSpider (www.cmpg.unibe.ch/software/PGDSpider/):

PGDSpider is a tool for population genetics data and genomics programs. It can translate datatypes between programs depending on the desired results outcome, and can handle conventional population genetics formats or NGS data. PGDSpider can be run on a CLI or through its own GUI.

Program 3: Plink (zzz.bwh.harvard.edu/plink/):

PLINK is a whole genome analysis toolset that performs a range of analyses of genotypic and phenotypic data. PLINK can: read and compress data in a variety of formats, perform summary statistics for quality control, determine allele and genotype frequencies, run isolation by distance (IBD) statistics, detect population stratification, handle virtually unlimited numbers of SNPs, significance test for whether two individuals belong to the same population, Fisher's exact test, Cochran-Armitage trend test, Mantel-Haenszel and Breslow-Day tests for stratified samples, post-analysis annotation of result files, extensions with R function plug-ins Web-based SNP and gene annotation lookup feature, and many other analyses and annotations.

Program 4: Geneious Prime (www.geneious.com/prime/):

Geneious Prime is a program for Sanger, NGS and long read sequence analysis, including pairwise and multiple alignments, de novo assembly, mapping, expression analysis, variant calling, NGS visualization, sequence and chromatogram analysis, automatic annotation, and phylogenetic tree building. Geneious Prime can import, export and convert sequences, annotations and notes in common file formats (Genbank, SnapGene, FASTQ, FASTA, BAM, VCF, and more).

Program 5: VCFtools (https://vcftools.github.io/):

VCFtools is a software package designed to work with VCF files, such as those generated by ipyrad. Genetic data can be very complex, and VCFtools can simplify them. It can filter out specific variants, summarize variants, merge files, convert to different file types (an important feature), and other operations.

Program 6: Structure Harvester (http://taylor0.biology.ucla.edu/structureHarvester/):

Structure Harvester is a popular website designed to take output files from STRUCTURE and calculate the optimal value for K. A variety of graphs and a table of calculated values are provided that can easily be viewed.

Program 7: Pophelper (https://rshiny.nbis.se/shiny-server-apps/pophelperShiny/inst/app/):

Pophelper is a versatile website that allows you to visualize population structure files. It can detect a range of input formats, and is fairly interactive. Users can select colors,

individuals, populations, edit image size, and more. The resulting plots (similar to a Distruct plot) can then be downloaded.

Program 8: MEGA (https://www.megasoftware.net/):

MEGA is a program used for phylogenetic inference of genetic data. It can align sequences, run analyses for tree construction, creating trees that researchers can use in their published research. It supports both command line (CLI) and desktop (GUI) options.

Chapter 2

Abstract:

Three species of freshwater fishes (*Belonesox belizanus* – Pike killifish, *Vieja maculicauda* – Black belt cichlid, and *Gambusia nicaraguensis* – Nicaraguan mosquitofish) are widely distributed across rivers on the Caribbean slope of Central America. *Belonesox belizanus* and *G. nicaraguensis* are poeciliids (live-bearing fish), while *V. maculacauda* is a cichlid (a diverse family of fishes primarily found in Africa, South America, and Central America). The overlapping distributions of these species allow for a comparative population genomics approach to understand the biogeographic history and evolution of these fishes. Past work used individual loci to assess general phylogeographic patterns with little structure detected within each species; however, these data from prior research lacked power to properly test hypotheses of population subdivision, gene flow, and recent expansion. The use of RADSeq data can be a powerful tool to overcome these obstacles, and when used in conjunction with other data can give insight into the historical biogeography of the region where other data may not be able to. Our results using the mitochondrial COI gene suggest four clades of *G. nicaraguensis*, while more comprehensive sampling using genomic data supports only three populations. Two populations were recovered for both *B. belizanus* and *V. maculicauda* using genomic data. Divergence among populations was associated with geographic breaks for the two poeciliids although the location of the geographic breaks differed between species. The two populations detected for *V. maculicauda* were highly divergent genetically but sympatric. This study gives insight into the historical biogeography of the region, showing that population structure is complex and varies across widespread species.

Introduction:

It has long been understood that in determining the biogeographic history of species, recovering evidence of shared evolutionary history reflects the degree to which different species are spatially congruent (Bermingham & Avise, 1986). Research may employ a variety of methods to elucidate both how much variation is present between and within species in order to discover more of the biological story of species. Even when examining conspecific populations, recovering evidence of differentiation can provide valuable information about a region's history (Bermingham & Avise, 1986). It is through such endeavors that researchers can develop a framework to create effective conservation efforts. However, these endeavors are currently stymied, in part due to under-sampling in many regions such as in Central American Neotropical rivers; therefore, research in these areas is essential to better understand the evolution and biogeographic history of Neotropical freshwater fishes. An additional consequence of this under-sampling is that many undescribed species may become extinct without our knowing of their existence. Myers et al. (2000) suggested that studies pertaining to biodiversity and conservation would benefit by focusing on areas where positive impact is maximized, and identified these areas with the term `biodiversity hotspots`. One such area is Middle America, the region relevant to this study.

The field of biogeography is interdisciplinary, and much remains to be learned about the biogeography of Middle America. For example, recent research by Matamoros *et al*. (2014) presented the first synthetic regional analysis of newly acquired taxonomic and distributional datasets that generated a high-resolution biogeographic analysis of Central American freshwater fishes. These results updated and validated the main conclusions of Myers (1966)

with the wealth of empirical data now available. McMahan *et al*. (2017) provided evidence of North to South range expansion and estimated levels of genetic divergence of *Vieja. maculicauda*, demonstrating the potential interdisciplinary nature of biogeographic studies, as it was hypothesized that the expansion of *V. maculicauda* is associated with an increase in habitat within the neotropics via Pleistocene glacial cycles.

Three species of freshwater fishes (*Belonesox belizanus* – Pike killifish, *V. maculicauda* – Black belt cichlid, and *Gambusia nicaraguensis* – Nicaraguan mosquitofish) share the characteristic of being widely distributed across rivers on the Caribbean slope of Middle America (Matamoros et al., 2014). *Belonesox belizanus* and *G. nicaraguensis* are poeciliids (live-bearing fish), while *V. maculacauda* is a cichlid (a diverse family of fishes primarily found in Africa, South America, and Central America). The overlapping distributions of these species allow for a comparative population genomics approach to understand the biogeographic history and evolution of these fishes. Past work used individual loci to assess general phylogeographic patterns with little structure detected within each species (Marchio and Piller 2013, McMahan et al., 2017); however, these data from prior research lacked power to properly test hypotheses of population subdivision, gene flow, and recent expansion (Meier et al., 2017). Additionally, past targeted work on one of the cichlids has incorporated species-distribution modeling to demonstrate the north to south dispersal of this species was likely associated with sea-level changes in the Pleistocene and Holocene (McMahan et al. 2017). Greater genomic coverage and an increase in sample sizes (geographic coverage and number of individuals) are essential for the objective of this research: to test hypotheses of population-level biogeographic patterns of these three species across their Middle American distribution.

In this study, we use ddRAD (Peterson et al., 2012) sequencing to conduct a phylogenetic analysis of *G. nicaraguensis*, *B. belizanus*, and *V. maculicauda* along the Caribbean slopes of Middle America. For *G. nicaraguensis*, we also sequenced the mitochondrial COI gene because these data were not available from previous studies. Given that our objective was to assess population structure in each species, we hypothesized that due to low genetic structure in previous single gene approaches (Marchio & Piller, 2013; McMahan et al., 2017) that there will not be many distinct populations recovered. However, given the power of genomic methods, we expected to find detectable levels of genomic divergence and evidence of population subdivision for each species. Due to possible differences in size, life history, and dispersal potential, more population structure is expected in *G. nicaraguensis* than *B. belizanus* and *V. maculicauda* (Lucinda, 2003; Marchio & Piller, 2013; McMahan et al., 2017; Malabarba & Malabarba, 2019). In addition to elucidating biogeographic events impacting the most diverse lineages of fishes in Middle America, there are profound implications for impact of sea-level rise. Such lowland- restricted freshwater fishes will likely be among the earliest impacted by habitat loss due to oceanic incursion. Thorough understanding of population dynamics is paramount to planning and execution of robust conservation efforts.

Methods:

Sample preparation

Specimens of *G. nicaraguensis* (n=32), *B. belizanus* (n=46), and *V. maculicauda* (n=54) were collected from throughout their respective Middle American distributions using seines, cast nets, and electrofishers (Table 1, Table 2, & Table 3, respectively).

Table 1: Museum collection accession information and locality data for samples of *Gambusia nicaraguensis*. FMNH=Field Museum of Natural History, LSUMZ-F=Louisiana State University Museum of Natural Science.

Collection	Catalog No.	RadSeq	COI	Country	Drainage	Latitude	Longitude
FMNH	179157	X	X	Costa Rica	Matina	9.94379	-83.01145
FMNH	179158	X	X	Costa Rica	Matina	9.94379	-83.01145
FMNH	179159	X	X	Costa Rica	Matina	9.94379	-83.01145
FMNH	179160	X	X	Costa Rica	Matina	9.94379	-83.01145
FMNH	179161	X	X	Costa Rica	Matina	9.94379	-83.01145
FMNH	179180	X	X	Costa Rica	Matina	9 53.627N	82 58.396W
FMNH	179181	X	X	Costa Rica	Matina	9 53.627N	82 58.396W
FMNH	179182	X	X	Costa Rica	Matina	9 53.627N	82 58.396W
LSUMZ-F	1038	X	X	Honduras	Salado	16.78027778	-87.035
LSUMZ-F	1187	X	X	Honduras	Tela	15.78236667	-87.4336
LSUMZ-F	2699	X		Nicaragua	Wounta	13.55813889	-83.53425
LSUMZ-F	2700	X		Nicaragua	Wounta	13.55813889	-83.53425
LSUMZ-F	2701	X		Nicaragua	Wounta	13.55813889	-83.53425
LSUMZ-F	2654	X		Nicaragua	Prinzapolka	13.43408333	-83.60158333
LSUMZ-F	2655	X		Nicaragua	Prinzapolka	13.43408333	-83.60158333
LSUMZ-F	2656	X		Nicaragua	Prinzapolka	13.43408333	-83.60158333
LSUMZ-F	2781	X		Nicaragua	Karata	13.91758333	-83.49508333
LSUMZ-F	2782	X		Nicaragua	Karata	13.91758333	-83.49508333
LSUMZ-F	3341	X	X	Honduras	Lis lis	15.89572222	-86.05205556
LSUMZ-F	3515	X		Honduras	Guaimoreto	15.65812556	-88.15077333
LSUMZ-F	3516	X	X	Honduras	Guaimoreto	15.65812556	-88.15077333

LSUMZ-F	3517		X	Honduras	Guaimoreto	15.65812556	-88.15077333
LSUMZ-F	3518	X	X	Honduras	Guaimoreto	15.65812556	-88.15077333
FMNH	08-2030	X		Honduras	Roatan		
FMNH	08-2031	X		Honduras	Roatan		
FMNH	08-2032	X		Honduras	Roatan		
FMNH	08-2033	X	X	Honduras	Roatan		
FMNH	08-2034	X	X	Honduras	Roatan		
FMNH	08-2035	X	X	Honduras	Roatan		
FMNH	08-2036	X	X	Honduras	Roatan		
LSUMZ-F	3316	X	X	Honduras	Motagua	15.72267222	-88.24449722
LSUMZ-F	1188	X	X	Honduras	Tela	15.78236667	-87.4336
LSUMZ-F	3352	X		Honduras	Lean	15.89844444	-86.12336111

Table 2: Museum collection accession information and locality data for samples of *Belonesox belizanus*. LSUMZ-F=Louisiana State University Museum of Natural Science, SLU-TC=Southeastern Louisiana University Vertebrate Museum, FMNH=Field Museum of Natural History, MT=Michi Tobler.

Museum	Catalog No.	Country	Drainage	Latitude	Longitude
LSUMZ-F	1052	Honduras	Salado	15.766	-86.999
LSUMZ-F	1059	Honduras	Salado	15.766	-86.999
SLU-TC	1607	Mexico	Papaloapan	18.519	-96.429
SLU-TC	1830	Mexico	Quintana Roo	18.444639	-89.101583
SLU-TC	1840	Mexico	Hondo	18.166	-88.683
SLU-TC	1841	Mexico	Hondo	18.166	-88.683
LSUMZ-F	2638	Nicaragua	Prinzapolka	13.421	-83.599
LSUMZ-F	2639	Nicaragua	Prinzapolka	13.421	-83.599
LSUMZ-F	2740	Nicaragua	Prinzapolka	13.91758333	-83.49508333
LSUMZ-F	2741	Nicaragua	Prinzapolka	13.91758333	-83.49508333
LSUMZ-F	2777	Nicaragua	Prinzapolka	13.91758333	-83.49508333

LSUMZ-F	2800	Nicaragua	Wawa	14.30911111	-83.71713889
MT	2941	Mexico	Grijalva	18.134	-93.285
SLU-TC	3039	Belize	Grande	16.23	-88.944
SLU-TC	3096	Belize	Grande	16.23	-88.944
SLU-TC	3100	Belize	Stann	17.02747	-88.32841
SLU-TC	3154	Belize	SilkGrass	16.919	-88.344
SLU-TC	3155	Belize	SilkGrass	16.919	-88.344
SLU-TC	3168	Belize	Sibun	17.404	-88.458
SLU-TC	3201	Belize	Golden	16.36	-88.793
SLU-TC	3217	Belize	Sibun	17.301	-88.554
SLU-TC	3218	Belize	Sibun	17.301	-88.554
SLU-TC	3219	Belize	Sibun	17.301	-88.554
SLU-TC	3289	Belize	Belize	17.087	-89.127
SLU-TC	3292	Belize	Belize	17.087	-89.127
SLU-TC	3297	Belize	Grande	16.219	-88.928
SLU-TC	3352	Belize	Stann	16.80533	-88.3707
SLU-TC	3378	Belize	Belize	17.187	-88.999
SLU-TC	3395	Mexico	Tonala	17.977	-94.114
LSUMZ-F	3485	Honduras	Guaimoreto	15.969	-85.861
LSUMZ-F	3486	Honduras	Guaimoreto	15.969	-85.861
LSUMZ-F	3487	Honduras	Guaimoreto	15.969	-85.861
LSUMZ-F	3701	Honduras	Laguna Cacao	15.717	-87.6
FMNH	130815	Guatemala	Peten-Itza	16.99288	-89.69354
MT	2941; 11-03	Mexico	Grijalva	18.134	-93.285
LSUMZ-F	5797	Guatemala	Peten-Itza	16.94475	-89.97425
LSUMZ-F	5815	Guatemala	Peten-Itza	16.94333333	-89.96447222
LSUMZ-F	5842	Guatemala	Peten-Itza	16.94333333	-89.96447222
LSUMZ-F	5899	Guatemala	Yaxha	17.06027778	-89.38816667
LSUMZ-F	5956	Guatemala	Sacnab	17.06380556	-89.37038889
LSUMZ-F	9591	Guatemala	Pasion	16.636787	-90.182241
LSUMZ-F	9590	Guatemala	Pasion	16.636787	-90.182241
LSUMZ-F	9636	Guatemala	San Pedro	17.260972	-90.864806
LSUMZ-F	9825	Guatemala	Izabal	15.734056	-89.078056

FMNH	179086	Costa Rica	Tortuguero	10.329829	-83.378632
FMNH	179090	Costa Rica	Tortuguero	10.329829	-83.378632
FMNH	179091	Costa Rica	Tortuguero	10.329829	-83.378632

Table 3: Museum collection accession information and locality data for samples of *Vieja maculicauda*. FMNH=Field Museum of Natural History, LSUMZ-F=Louisiana State University Museum of Natural Science, WAM=Wilfredo Matamoros, STRI=Smithsonian Tropical Research Institute.

Collection	Catalog No.	Country	Drainage	Latitude	Longitude
LSUMZ-F	2599	Nicaragua	Prinzapolka	13.4205	-83.59858333
LSUMZ-F	2600	Nicaragua	Prinzapolka	13.4205	-83.59858333
LSUMZ-F	2601	Nicaragua	Prinzapolka	13.4205	-83.59858333
LSUMZ-F	2702	Nicaragua	Bibiskira	13.76941667	-83.55741667
LSUMZ-F	2703	Nicaragua	Bibiskira	13.76941667	-83.55741667
LSUMZ-F	2704	Nicaragua	Bibiskira	13.76941667	-83.55741667
LSUMZ-F	2775	Nicaragua	Wawa	13.93869444	-83.53997222
LSUMZ-F	2828	Nicaragua	Wawa	14.30241667	-83.67613889
LSUMZ-F	3309	Honduras	Motagua	15.72267222	-88.24449722
LSUMZ-F	3310	Honduras	Motagua	15.72267222	-88.24449722
LSUMZ-F	3311	Honduras	Motagua	15.72267222	-88.24449722
LSUMZ-F	3312	Honduras	Motagua	15.72267222	-88.24449722
LSUMZ-F	3313	Honduras	Motagua	15.72267222	-88.24449722
LSUMZ-F	3544	Honduras	Guaimoreto	15.980694	-85.887194
LSUMZ-F	3545	Honduras	Guaimoreto	15.980694	-85.887194
LSUMZ-F	3546	Honduras	Guaimoreto	15.980694	-85.887194
LSUMZ-F	3696	Honduras	LagunaCacao	15.794444	-86.546028
LSUMZ-F	3697	Honduras	LagunaCacao	15.794444	-86.546028
LSUMZ-F	3699	Honduras	LagunaCacao	15.794444	-86.546028
LSUMZ-F	3700	Honduras	LagunaCacao	15.794444	-86.546028

LSUMZ-F	5590	Guatemala	Izabal	15.65908333	-89.00302778
LSUMZ-F	5591	Guatemala	Izabal	15.65908333	-89.00302778
FMNH	130871.1	Guatemala	Izabal	15.65908333	-89.00302778
FMNH	130871.2	Guatemala	Izabal	15.65908333	-89.00302778
WAM	06-98	Honduras	Danto	15.77338	-86.81538
WAM	06-99	Honduras	Danto	15.77338	-86.81538
WAM	06-100	Honduras	Danto	15.77338	-86.81538
WAM	06-174	Honduras	Salado	15.76825	-87.00183
WAM	06-175	Honduras	Salado	15.76825	-87.00183
WAM	07-345	Honduras	Aguan	15.745177	-85.885362
LSUMZ-F	3364	Honduras	Aguan	15.8984444	-86.123361
WAM	08-1260	Honduras	Coco	14.82353	-84.49922
WAM	08-1261	Honduras	Coco	14.82353	-84.49922
WAM	08-1262	Honduras	Coco	14.82353	-84.49922
WAM	08-1264	Honduras	Coco	14.82353	-84.49922
WAM	08-1265	Honduras	Coco	14.82353	-84.49922
LSUMZ-F	4024	Honduras	Patuca	14.49572	-85.97054
LSUMZ-F	4025	Honduras	Patuca	14.49572	-85.97054
LSUMZ-F	4026	Honduras	Patuca	14.49572	-85.97054
LSUMZ-F	4286	Honduras	Patuca	14.79279	-85.19398
LSUMZ-F	4287	Honduras	Patuca	14.79279	-85.19398
LSUMZ-F	4315	Honduras	Patuca	14.566285	-85.2736
LSUMZ-F	4697	Honduras	Patuca	15.13459	-84.6547
LSUMZ-F	4740	Honduras	Patuca	14.64152	-85.3209
LSUMZ-F	4742	Honduras	Patuca	14.64152	-85.3209
WAM	08-0329	Honduras		15.13459	-84.6547
STRI		Panama	Rio Chagres	09 17'28.2"N	79 54'43"W
UMMZ	246298	Belize	Stann Creek	16.813611	-88.3775
WAM	08-2560	Honduras	Salado		
FMNH	179071	Costa Rica	Tortuguero		
FMNH	179072	Costa Rica	Tortuguero		
FMNH	179073	Costa Rica	Tortuguero		
FMNH	179074	Costa Rica	Tortuguero		

FMNH	179075	Costa Rica	Tortuguero		
FMNH	179168	Costa Rica	Matina	9.893783	-82.973267
FMNH	179169	Costa Rica	Matina	9.893783	-82.973267
FMNH	179170	Costa Rica	Matina	9.893783	-82.973267
FMNH	179171	Costa Rica	Matina	9.893783	-82.973267
FMNH	179175	Costa Rica	Matina	9.893783	-82.973267
FMNH	179209	Costa Rica	Matina	9.759812	-82.869601
FMNH	129603.1	Guatemala	Izabal	15.69644	-89.05877
FMNH	129603.2	Guatemala	Izabal	15.69644	-89.05877
FMNH	129603.3	Guatemala	Izabal	15.69644	-89.05877

Fishes were euthanized with an overdose of MS-222 prior to preservation. Tissue samples (muscle and/or fin clips) were preserved in 95% ethanol. Voucher specimens were subsequently preserved with 10% formalin, then stored in 70% ethanol and deposited in museum collections at Southeastern Louisiana University (SLU), LSU Museum of Natural Science (LSUMZ), and at the Field Museum of Natural History (FMNH).

DNA extraction of tissue samples was performed with the Qiagen DNeasy Tissue Kit following manufacturer protocol. The concentration of purified DNA samples was measured (ng/µL) with a Qubit fluorometer (Life Technologies, Inc.) as well as through visual examination using a 1% agarose gel. Only samples with DNA concentrations above 10ng/µL DNA were included in this study.

Mitochondrial sequencing sample preparation

Whole genomic DNA was isolated using the Qiagen DNeasy tissue kit and then used as a template for PCR. Each 25µL COI PCR reaction consisted of: 0.75µL of 25mM MgCl; 2.5µL of 10x

buffer; 1.0µL of dNTPs; 1.0µL of each 10 mM primer; 0.5 units of Taq; 2µL of DNA template; and 16–18 µL nuclease-free water. The primers used for COI amplification were BOL-F (forward) and BOL-R (reverse) from Ward et al. (2007). The thermocycler protocol for the COI gene was initial denaturation at 95 °C for 30 s; 25 cycles of 95 °C for 60 s, 52 °C for 30 s, 72 °C for 105 s; and a final extension at 72 °C for 240 s.

After amplification, samples were sequenced on a 3730 DNA Analyzer (Thermo Fisher Scientific). The per-sample 10µL protocol for 3730 sequencing was: 1µL Terminator Ready Reaction mix (Big Dyes); 3µL BigDye Seq Buffer (Dilution Buffer); 0.5µL primer; 2µL Template DNA, and 3.5µL water. The samples were run two times – once with the forward primer and once with the reverse primer. The thermocycler protocol was an initial denaturation of 96 °C for 1 min, 96°C for 10 sec, 50°C for 5sec, and 60°C for 4 min. After thermocycling, 2.5µL 125mM EDTA and 30µL 100% EtOH (32.5 µL) is added to each sample. Tubes are sealed and inverted to mix, then left at room temperature for up to 15 min to precipitate extension products. Samples were spun in refrigerated centrifuge at 2500g for 30 min at 4°C. The seal was then removed, and the tray was inverted onto a paper towel and secured with rubber bands. The tray was placed inverted into the centrifuge and spun 50g (up to 185 g) for 3 minutes. 30µl 70% EtOH was added to each pellet. Tubes were then resealed and inverted a few times to mix. The plate was spun at 2000-3000 g for 15 minutes at 4°C. The seals were removed, and the tray was inverted onto a paper towel and secured with rubber bands again. The tray was placed inverted into the centrifuge and spun 50g (up to 185 g) for 3 minutes to remove excess 70% EtOH. At this stage, samples are nearly ready to be resuspended for the 3730 DNA Analyzer

sequencing run. To resuspend samples and run on the 3730 DNA Analyzer, 10µl Hi-Di formamide was added to each tube.

COI sequence analysis

After sequencing, the resulting chromatograms were visually inspected in Geneious v2021.1.1. Samples were quality trimmed and an alignment was generated using the Muscle algorithm and default parameters, then exported as a NEXUS file. The NEXUS file was imported into MEGA (v.6) (Tamura et al., 2013). A parsimony analysis was performed with Bootstrap resampling for 100 replicates. Representative GenBank sequences from *Gambusia yucatana* and *Gambusia sexradiata* were included as additional ingroups, and *Gambusia affinis* was included as an outgroup (HQ564575, HQ564607, and HQ567415, respectively). MEGA was also used to analyze sequence divergence between recovered clades in the phylogenetic tree.

ddRAD sample preparation

To determine if the restriction endonucleases MspI and PstI were able to adequately digest samples, a genomic digest was performed with both MspI and PstI restriction endonucleases. DNA sample concentrations were optimized to 100ng/µl. Per well, 1ul MspI, 1µl PstI, 5µl NEBuffer, and the optimized volume of DNA were added. Nuclease-free water was then added until a total sample volume of 25µl was reached. The samples were incubated in a thermocycler protocol at 37°C for 60 min, inactivated at 80°C for 5 min, and held at 4°C. After running the thermocycler product on a 1% agarose gel to visualize digestion, samples were sent to the University of Wisconsin-Madison Biotechnology Center (UWMBC) for library prep and sequencing on a NovaSeq 6000 System (Illumina).

ddRAD assembly and filtering

The raw data files returned from UWMBC were run through `FastQC` v0.11.9 (Andrews 2010) to check the overall quality of the reads from the Illumina run. The FASTQ file output from the previous step became the input file for `ipyrad` (Eaton and Overcast 2020) pipeline for assembly and initial filtering. Reads that contained more than 5 bases with a low quality Phred score (<33) were excluded. Reads were then clustered based on an 85% similarity threshold and reads with less than 10x coverage were filtered out. A maximum of 5 ambiguous base calls and 5 heterozygous sites per read were allowed during filtering.

Population Structure

`VCFtools` (Danecek et al. 2011) was used to exclude individuals with more than 80% missing data, loci (SNPs) with a 60% call rate or lower, samples with less than 10x coverage, along with excluding all outgroup individuals to produce a VCF file of raw reads for 67 samples of *B. belizanus*, 32 samples of *G. nicaraguensis*, and 59 samples of *V. maculicauda*. The resulting VCF files were loaded into RStudio 1.4.1106 (RStudio Team 2021) and a text file was attached containing basic population information (which drainage basin each sample was collected from) to each sample, then converted to a `genlight` object using `vcfR` v1.10.0 (Knaus and Grünwald 2017).

Next, a principal component analysis (PCA) was generated from the `genlight` object, creating a two-dimensional graphic of observed genomic variation between samples. A PCA was performed using both the first and second principal components (PCs). Visualization of the resulting PCAs was done using `RColorBrewer` v.1.1-2 (Neuwirth and Brewer 2014) color-

blind friendly palette `Dark2`, `Set1`, and `Set2` within `ggplot2` (Wickham 2016) in `tidyverse` 1.3.0 (Wickham et al. 2019).

A discriminant analysis of principal components (DAPC) was created, following the methods of Jombart et al. (2010). DAPC is a multivariate *a priori* method that allows for the inference of population structure by determining the number of observed clusters (Pritchard et al. 2000; Jombart et al. 2010; Grünwald and Goss 2011). The data is partitioned into a between-group and within-group component in order to maximize the discrimination between groups, by first transforming the data into PCA and then by identifying clusters using discriminant analysis (Jombart and Ahmed 2011).

In order to generate a DAPC, we used `poppr` v2.9.1 (Kamvar et al. 2014) and its required packages `adegenet` v2.1.3 (Jombart 2008; Jombart and Ahmed 2011) and `ade4` (Dray and Dufour 2007), as well as `ape` 5.4-1 (Paradis and Schliep 2019), and used the `genlight` object produced in the initial filtering step. We then visualized the percent of variance explained by PCA as well as the discriminant analysis eigenvalues by following the methods of Jombart and Collins (2015), which indicated retaining 7 principal components and 3 discriminant functions for *V. maculicauda*, 7 principal components and 3 discriminant functions for *G. nicaraguensis*, and 4 principal components and 3 discriminant functions for *B. belizanus*.

Next, the DAPC object was transformed into the correct data frame format for visualization using `reshape2` (Wickham 2007) to convert the data, followed by `RColorBrewer` v.1.1-2 (Neuwirth and Brewer 2014) color-blind friendly palette `Dark2` and `ggplot2` (Wickham 2016) within the `tidyverse` 1.3.0 (Wickham et al. 2019) to visualize the

resulting DAPC analysis. To visualize the posterior assignment of each sample within the DAPC object, a composite stacked bar plot was created using the `compoplot` function within the `adegenet` v2.1.3 package, and visualized the resulting plot using `RColorBrewer` v.1.2.1 and `ggplot2`.

In addition, we ran the Bayesian clustering method `STRUCTURE`. Inferred populations (K) were evaluated from 1 to 11. Ten independent runs for each K were implemented with a burn-in period length of 10,000 iterations, followed by 100,000 Monte Carlo Markov Chains (MCMC) replicates. The most probable K value was determined using both likelihood and Delta K criteria (ΔK), and calculated using `Structure Harvester` (Earl & von Holdt, 2012). The most probable K value was then used to generate a barplot via the program `PopHelper` v2.1.1 (Francis, 2017). An unweighted Weir and Cockerham (1984) pairwise F_{ST} between clusters or populations was calculated. An unweighted Weir and Cockerham pairwise F_{ST} allows for the analysis of between-group variation present among the major clades recovered in previous analyses (Weir and Hill 2002; Weir and Goudet 2017). This was done using the `hierfstat` package (Goudet et al. 2005) v.0.5-7. Lastly, we conducted hierarchical AMOVAs, calculated by transforming the `genind` object into a `genclone` object using the `poppr` package, to test for genetic structure.

Results:

Mitochondrial analysis in *G. nicaraguensis* mitochondrial COI gene:

Including the outgroups, the aligned *G. nicaraguensis* COI mitochondrial gene sequences were truncated to 655bp long and had 35 variable characters. Of these variable characters, 30

were parsimony informative. Datasets were visualized using PopART for the haplotype network.

Phylogenetic analysis:

A generated parsimony tree of the *G. nicaraguensis* COI dataset resulted in four clades (Figure 2.1). Clade 1 was largely represented by samples collected from Roatan, clade 2 was represented by a single specimen from Tela (mainland Honduras), clade 3 was represented by samples from the mainland of Honduras and two specimens from Panama, and clade 4 by samples from the remainder of the sampled distribution in Costa Rica and Panama. Bootstrap support values were above 90 and indicated strong support for the individual clades, however support was lower for relationships among those clades.

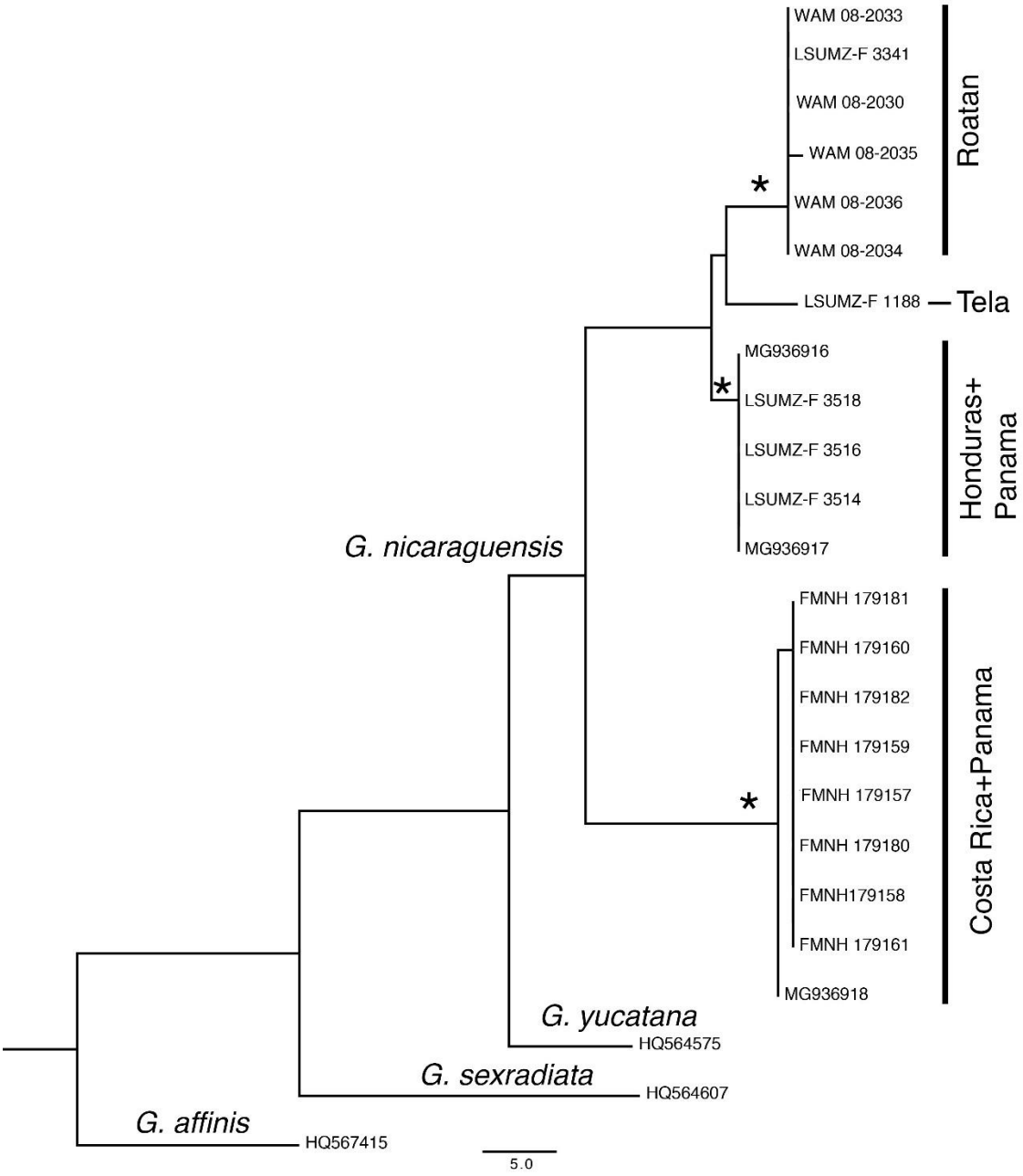


Figure 2.1: Parsimony tree based on mitochondrial gene COI for *G. nicaraguensis*.

Population structure analysis using SNP data:

G. nicaraguensis:

A total of 269,479 binary SNPs were recovered, with 30.28% missing data. In generating a PCA, the first 11 principal components (PCs) were saved as an object in RStudio, and a scatter plot was visualized from the first (47.808% variance explained) and second (8.362% variance explained) PCs (Figure 2.2).

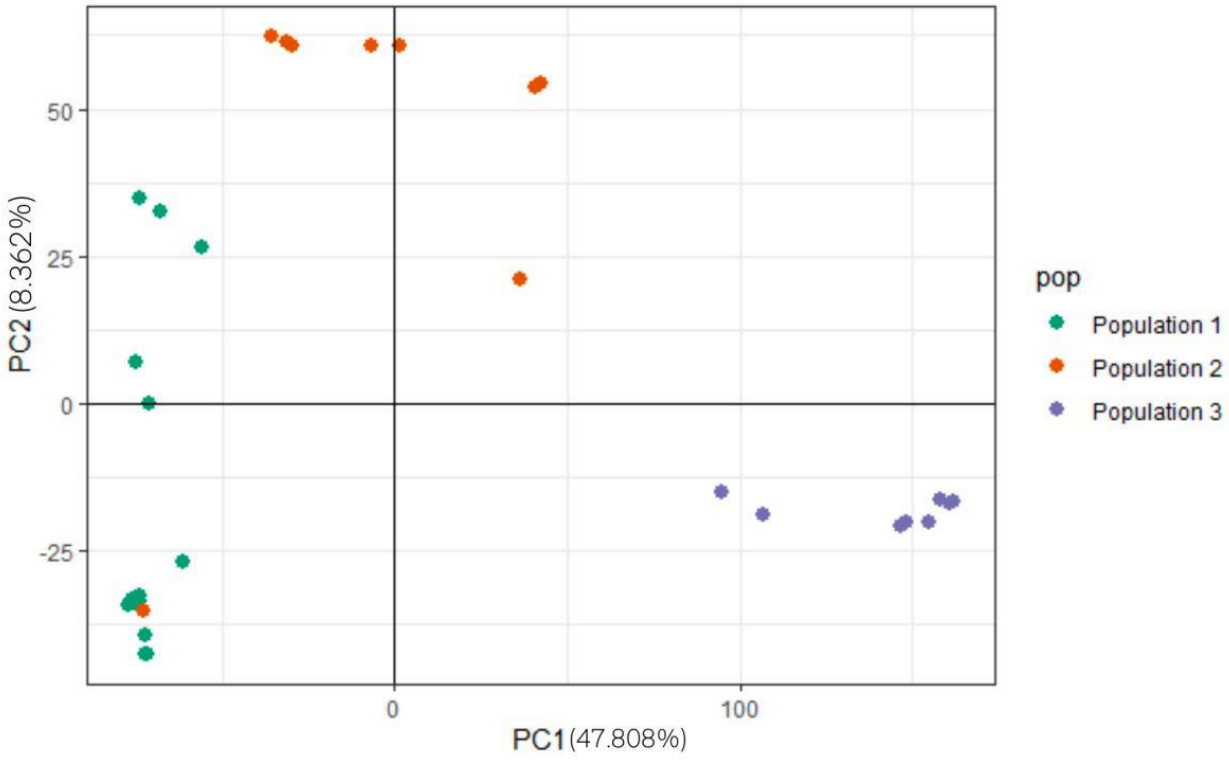


Figure 2.2: Principal component analysis of the processed RADSeq data for *G. nicaraguensis*. Individuals are labeled based on the drainage in which they were collected. The first and second principal components are used.

The DAPC was generated containing the first 7 PC's and 3 discriminant functions saved as an object in RStudio. The conserved variance was 70.0%. Principal components were saved based on the pattern observed in generated scree plots. The DAPC (Figure 2.3) shows three

groups of populations clustering based on the proximity of the drainage basins. The genetic structure of the Matina population (Population 3; purple) is the most unique, as reflected by both the DAPC and barplot characterizing it as a totally separate cluster.

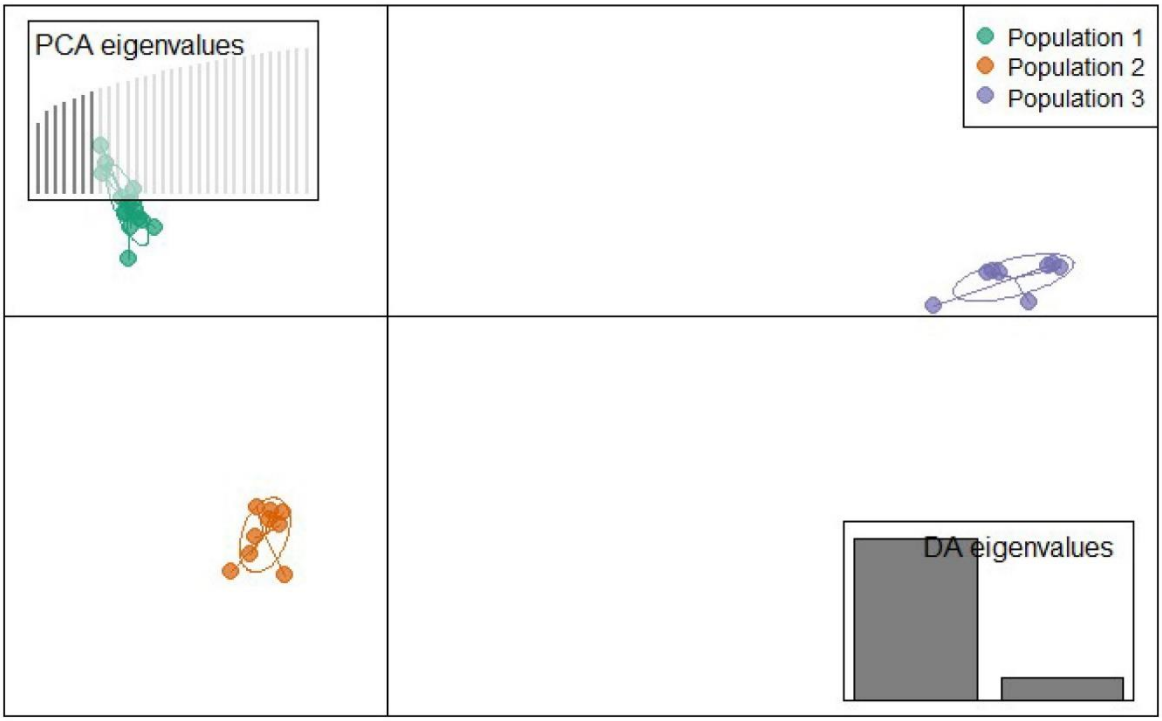


Figure 2.3: DAPC of *G. nicaraguensis*. PCA eigenvalues are in the top left corner, and DA eigenvalues are in the bottom right corner.

In order to find the optimal K-value, the number of clusters (K) was plotted against ΔK, which showed a sharp peak at K = 3, which is consistent with the DAPC analysis. A barplot of K=3 shows three groups (Figure 2.4).

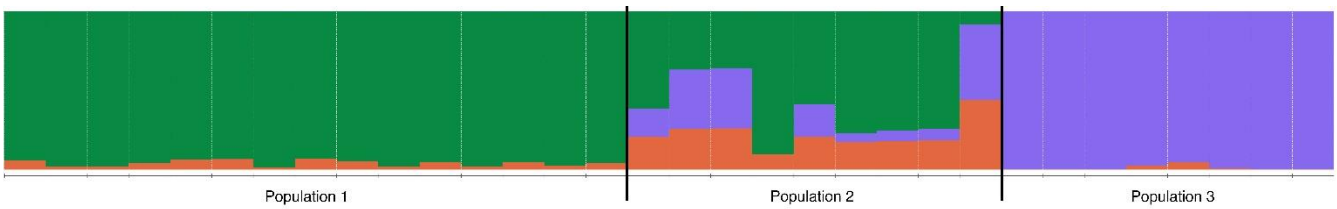


Figure 2.4: A barplot representing K = 3, representing the population structure of *G. nicaraguensis* with three assumed populations. Patterns of divergence are emphasized by black lines corresponding to breaks in populations.

First, there is a group containing individuals from Lean, Salado, Tela, Roatan, Lis-Lis, and Guaimoreto, represented as population 1. There is then an intermediate group containing the drainages from Prinzapolka, Wounta, and Karata, represented as population 2. Lastly there is a group containing individuals from Matina, represented as population 3. Population 2 contains notable admixture from populations 1 and 3. Interestingly, despite being located at the highest longitude closest to individuals from population 1, an individual from Motagua is shown to have admixture from all three populations. A map of the sampling distribution of *G. nicaraguensis* containing color coded dots that correspond to recovered populations can be seen in Figure 2.5.

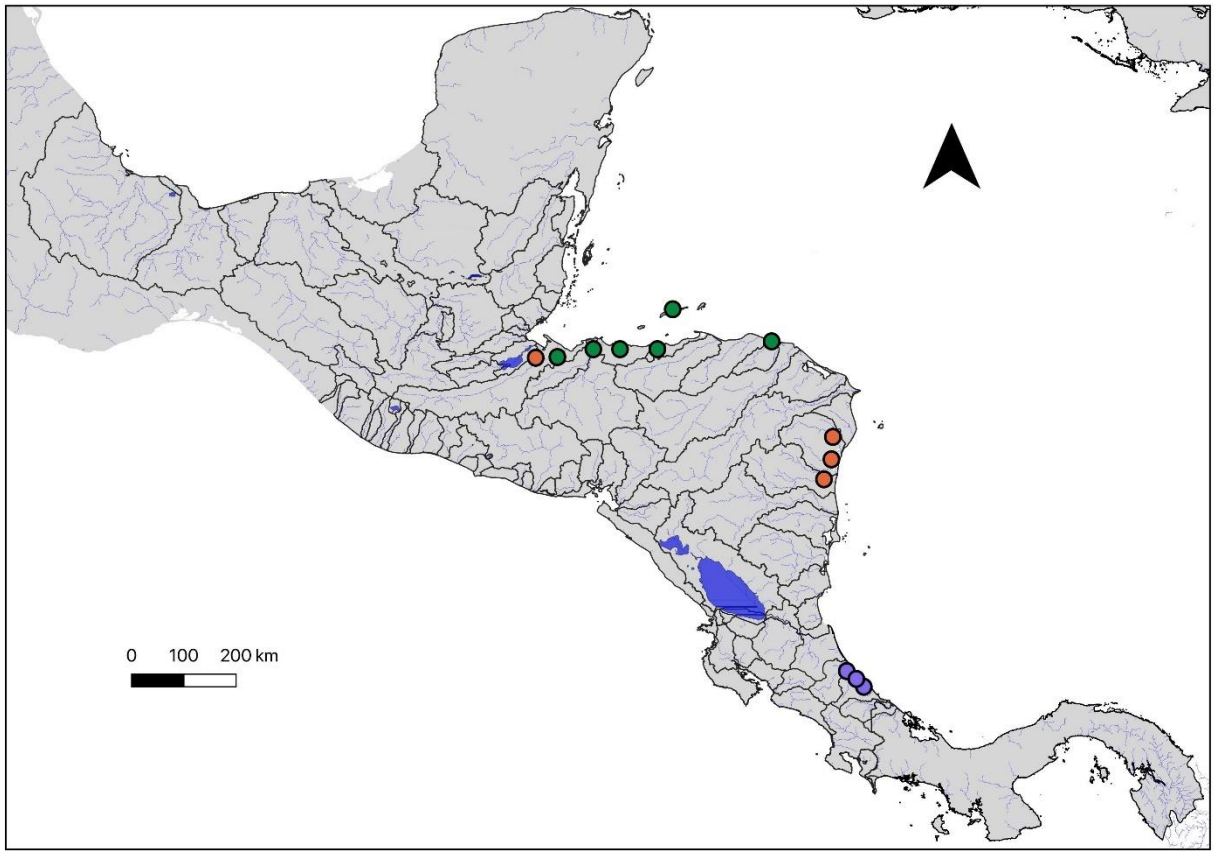


Figure 2.5: Sampling distribution of *G. nicaraguensis*. Colored dots represent sampling sites, with each color (green, orange, purple) corresponding to distinct populations recovered from this study.

Statistical analyses:

Pairwise Fst values of genomic SNPs were calculated for the inferred populations of *G. nicaraguensis*, following Weir and Cockerham (1984), with a value of 0.7543 between populations 1 and 3, a value of 0.2365 between populations 1 and 2, and a value of 0.5598 between populations 2 and 3. An analysis of molecular variance (AMOVA) showed that 69.178% of the genetic variation segregated between samples, and 30.822% of the variation segregated within samples ($p < 0.001$).

B. belizanus:

A total of 139,021 binary SNPs were uncovered, with 18.5% missing data. In generating a PCA, the first 11 principal components (PCs) were saved as an object in RStudio, and a scatter plot was visualized from the first (46.974% variance explained) and second (6.43% variance explained) PCs. (Figure 2.6).

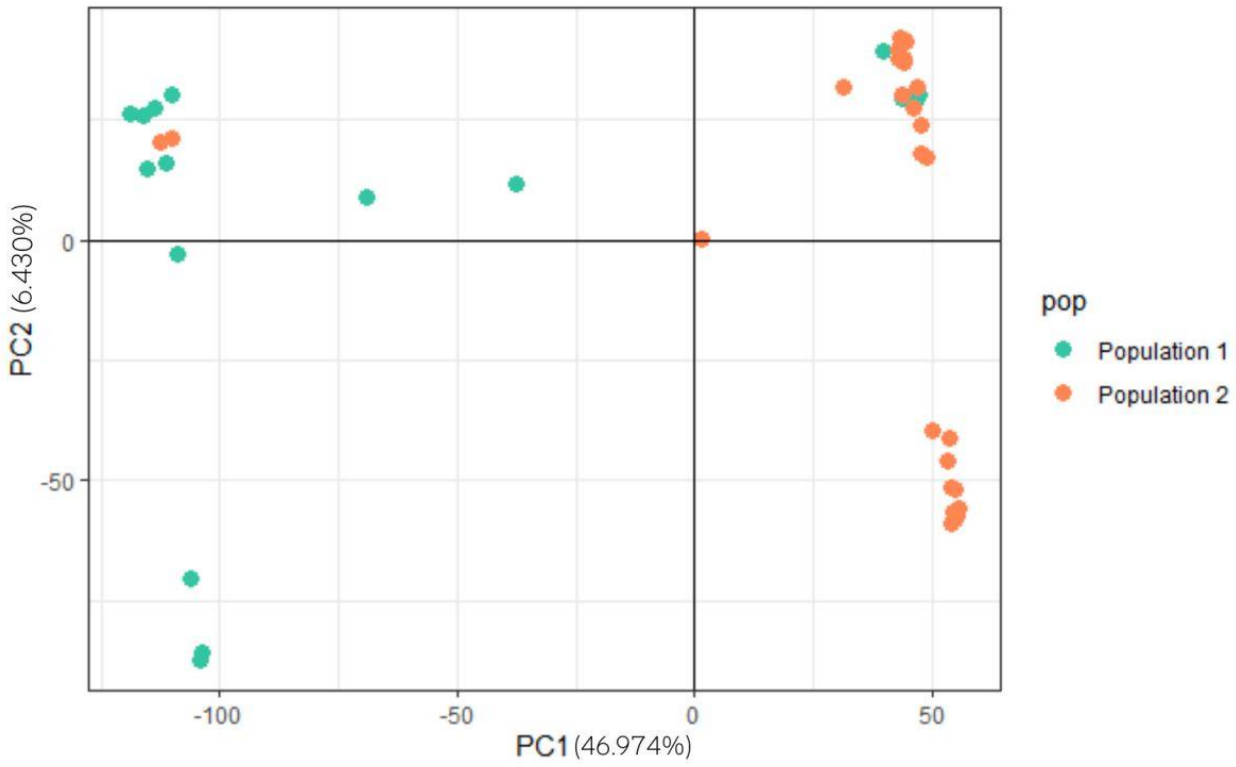


Figure 2.6: Principal component analysis of the processed RADSeq data for *B. belizanus*.

A DAPC containing the first 4 principal components and 3 discriminant functions conserved 68.8% of the variance. The DAPC shows population structure of *B. belizanus* samples containing admixture between each of the two populations (Figure 2.7).

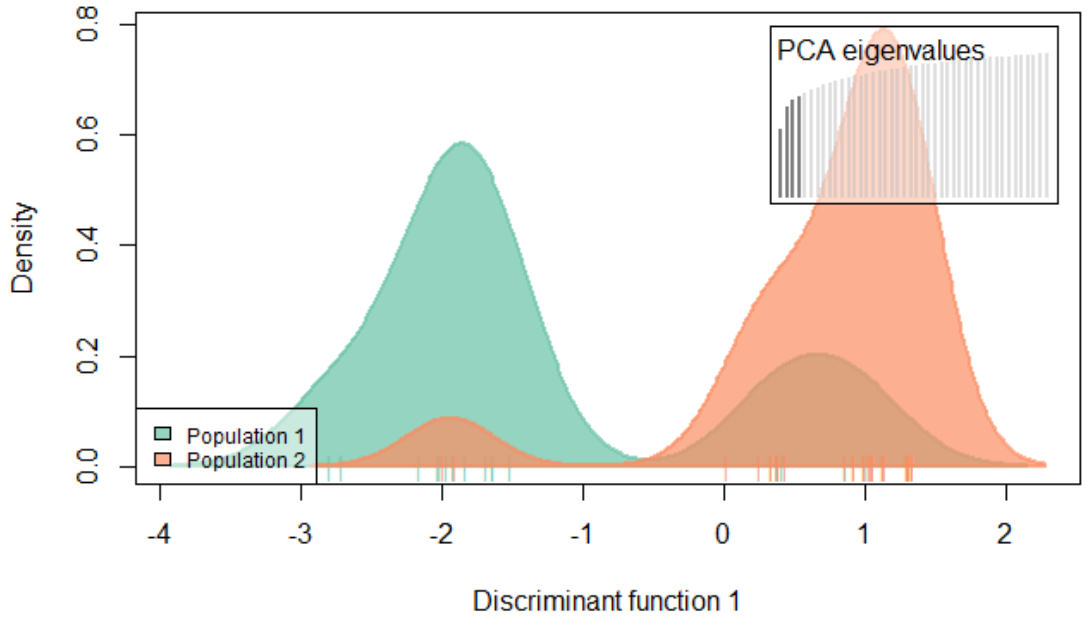


Figure 2.7: DAPC of *B. belizanus*. PCA eigenvalues are in the top right corner.

In order to find the optimal K-value, the number of clusters (K) was plotted against ΔK, which showed a sharp peak at K = 2. A barplot of K = 2 shows distinct population structure (Figure 2.8).

Figure 2.8: A barplot displaying K = 2, representing the population structure of *B. belizanus* with two assumed populations. Patterns of divergence are emphasized by black lines corresponding to breaks in populations.

Population 1 contains individuals from Champoton, Papaloapan, Yucatan, Hondo, Grijalva, Tonala, Sibun, and Belize drainages. It is after the Belize drainage that (population 2) appears, beginning from the Stann drainage and continuing southward to the end of the Tortuguero drainage, where the sampling ended. The DAPC showed concordance with the findings of `STRUCTURE`, as the programs identified two clusters with some admixture. A map of the sampling distribution of *B. belizanus* containing color coded dots that correspond to recovered populations can be seen in Figure 2.9.

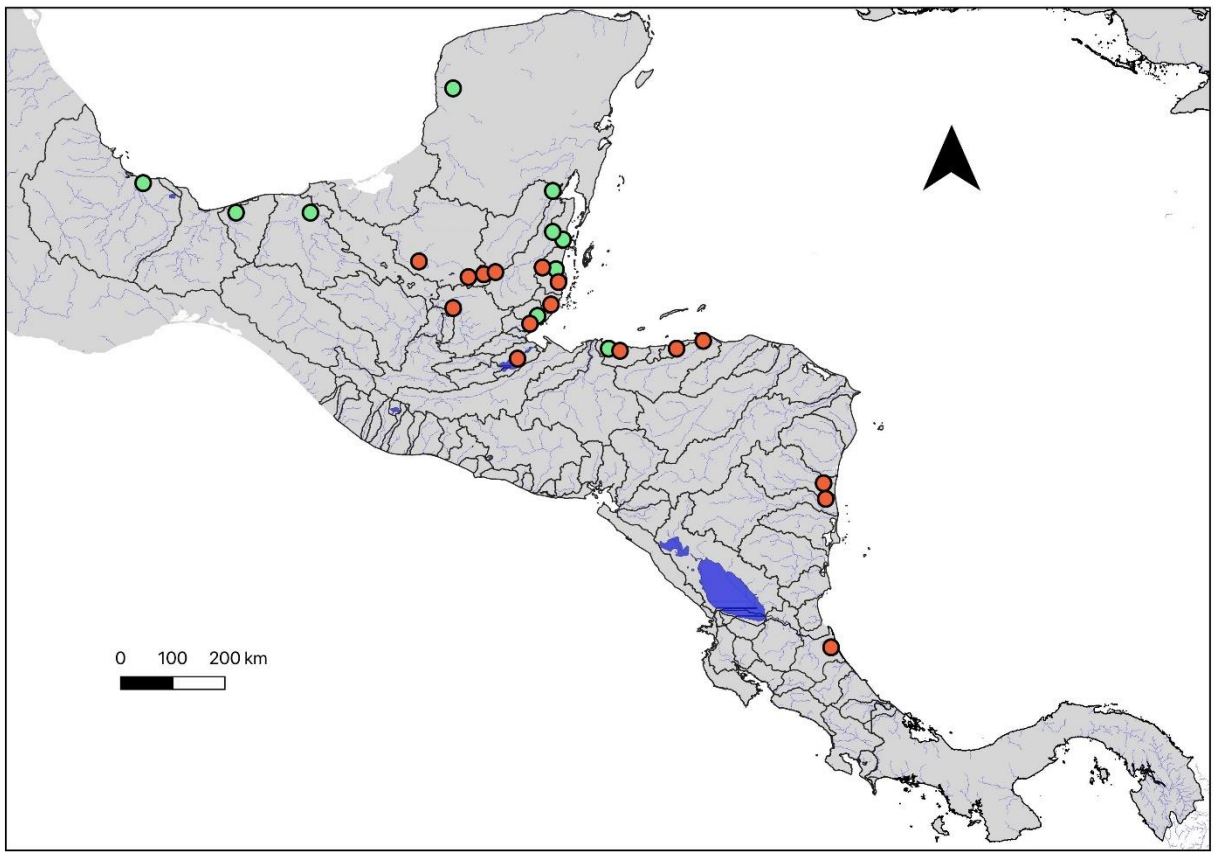


Figure 2.9: Sampling distribution of *B. belizanus*. Colored dots represent sampling sites, with each color corresponding to a distinct population recovered from this study.

Statistical analyses:

Pairwise F_{st} values were calculated for the drainage populations of *B. belizanus*, following Weir and Cockerham (1984), with a value of 0.3881 between the two recovered populations. An analysis of molecular variance (AMOVA) showed 36.12% of the genetic variation segregated between samples, and 63.88% of the variation segregated within samples ($p < 0.005$).

V. maculicauda:

A total of 191,122 binary SNPs were uncovered, with 13.7% missing data. In generating a PCA, the first 7 principal components were used, and 3 discriminant functions were saved as an object in RStudio. A PCA that retained the first (80.626% variance explained) and second (6.43% variance explained) PC was plotted from the aforementioned object (Figure 2.10).

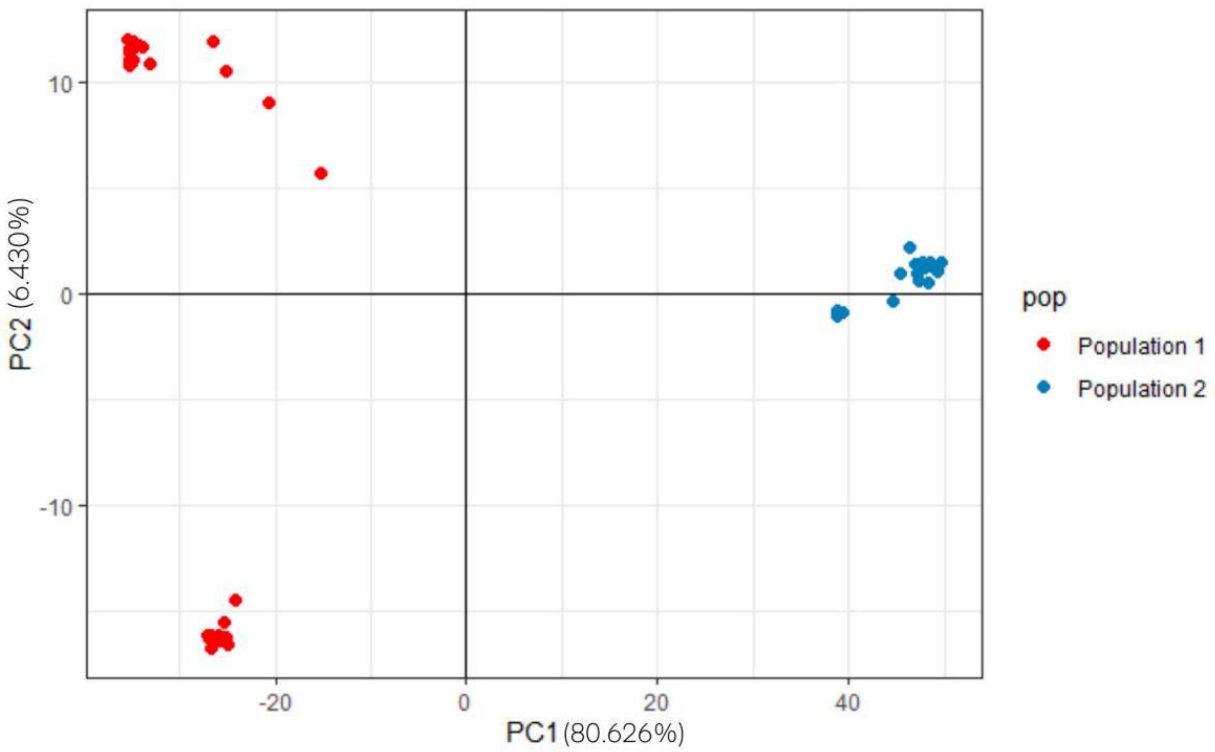


Figure 2.10: Principal component analysis of the processed RADSeq data for *V. maculicauda*. Individuals are labeled based on the drainage in which they were collected. The first and second principal components are used.

The DAPC generated (Figure 2.11) identified two clusters that were interspersed geographically.

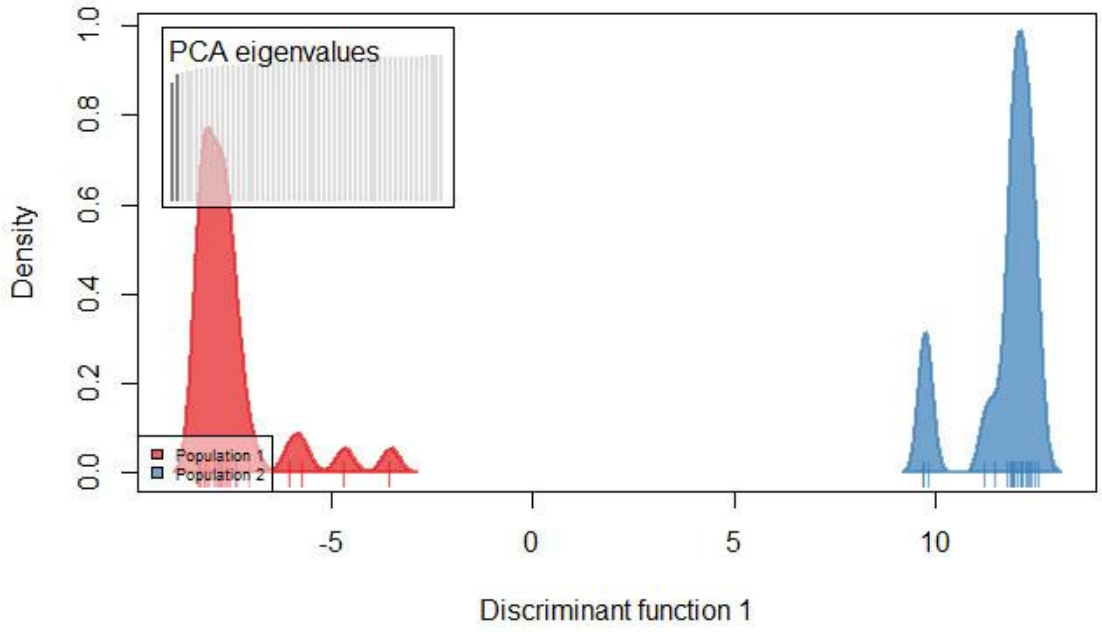


Figure 2.11: DAPC of *V. maculicauda*. PCA eigenvalues are in the top left corner. DA eigenvalues correspond to 87% of conserved variance.

The barplot produced by the STRUCTURE analysis gave an optimal value of K = 2 (Figure 2.12). Each of the populations denoted contains individuals found within the same rivers and drainages, i.e. that population structure was not concordant with geographic breaks. For example, individuals from both populations 1 and 2 are found within the Coco drainage basin in Honduras. Despite being completely interspersed geographically, the two populations show no genetic admixture.

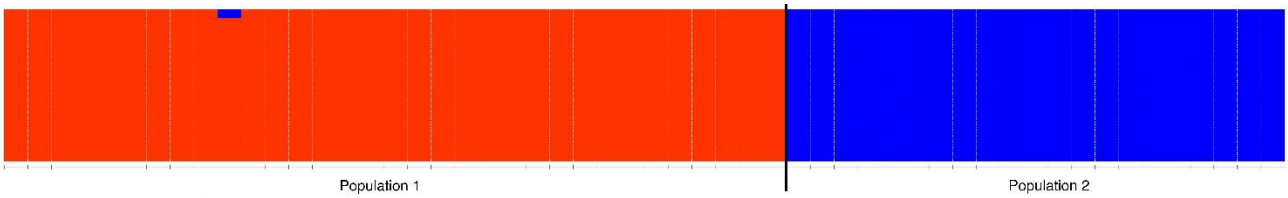


Figure 2.12: A barplot representing K = 2, showing the population structure of *V. maculicauda*. Samples are arranged in clusters. Patterns of divergence are emphasized by black lines corresponding to breaks in populations.

A map of the sampling distribution of *V. maculicauda* containing color coded dots that correspond to recovered populations can be seen in Figure 13.

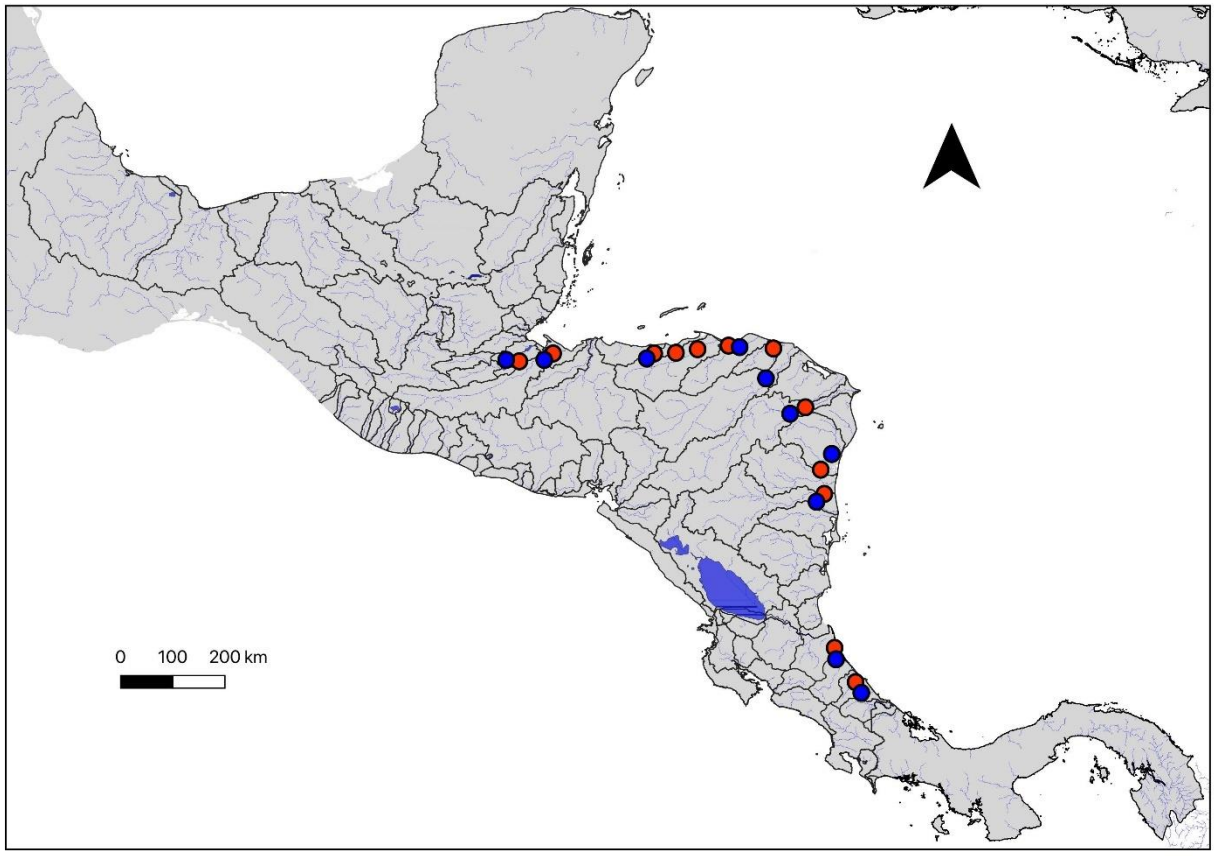


Figure 2.13: Sampling distribution of *V. maculicauda*. Colored dots represent sampling sites, with each color corresponding to a distinct population recovered from this study.

Statistical analyses:

A pairwise Fst value was calculated for the clustered populations of *V. maculicauda*, following Weir and Cockerham (1984), with a value of 0.9524 between population 1 and 2. An analysis of molecular variance (AMOVA) showed that 98.002% of the genetic variation segregated between samples, and 1.998% of the genetic variation segregated within samples ($p < 0.001$), the lowest value for within-sample variation of the three species.

Discussion:

The present study revealed notable genetic structure among Middle American sampled populations of *G. nicaraguensis*, *B. belizanus*, and *V. maculicauda*, demonstrating concordance between biogeographic boundaries with two of the three species. Additionally, these results are novel in that this is the first comparative biogeographic study of lowland secondary freshwater cichlid and live bearing fish species using RADSeq data. These data showed a consistent agreement across AMOVA, DAPC, and `STRUCTURE` analyses, and indicate that *G. nicaraguensis* has the most population structure of the three sampled species with three distinct recovered populations. These results also suggest that gene flow is variable across the species' genome and may be affected by different processes and/or is a result of recent population expansion. Receding ocean levels from the last glacial maximum, habitat discontinuities, recent population expansion, and riparian and/or coastal travel could all be

responsible for contemporary population structure in each of the three species, as complex topographies can influence dispersal. All three AMOVAs demonstrate significant between- and within-group variation throughout the sampled rivers and basins, and the DAPCs in each species agree with the results from the STRUCTURE. The composite plots show that populations of the two poeciliids exhibit some degree admixture with populations in other drainage systems. The cichlid showed strong genetic divergence among the two inferred populations but there was no geographic pattern to the divergence among populations. *Gambusia nicaraguensis* had a greater number of populations across its distribution than both *V. maculicauda* and *B. belizanus*.

Sea-level fluctuations may contribute to the observed patterns, as coastal waters in Middle America were subject to periods of isolation when sea level dropped as much as 130 m below current levels during the Pleistocene (McMahan et al., 2017). In some cases, rising and lowering of sea levels caused islands to appear off the coast, leading to geographic and biological isolation, while along the coast it may have led to population differentiation, as seen here in *G. nicaraguensis*. After the last glacial maximum, populations of *V. maculicauda* began to expand southward as new habitat became available (McMahan et al., 2017). This may also be the case with *G. nicaraguensis* and *B. belizanus*; however, more work on these fishes targeting the historical migration and dispersal of each species is needed before such conclusions can be drawn. Many of the sampled individuals in each species were taken from floodplains, which may also contribute to the observed patterns; for example, if two otherwise isolated rivers are subject to extreme rains and flooding, then individuals from different rivers would come into contact with each other during floods and thereby hybridize.

One challenge is to distinguish between divergence driven by selection and drift, so it is important to note that these processes may be co-occurring and could be acting on populations found throughout Middle America, given its complex geologic history and lowland environment (Montes et al., 2015). It is possible that physical barriers between the drainages restrict dispersal, allowing populations to diverge along geological breaks, with selection reinforcing those barriers.

The two poecilids STRUCTURE outputs both show more concordance with biogeographic breaks than the black belt cichlid, indicating that the patterns of population structure may be species specific in this region and relate to other factors besides the presence of geographic barriers like dispersal potential, though the biogeographic breaks between the two poeciliids were not at the same geographic points. Recent population expansion may explain the less distinct population structure in *V. maculicauda* and *B. belizanus*. An alternative may be that not necessarily gene flow, but rather recent geological connections between populations is the main driving force behind the observed results, especially with regards to *V. maculicauda*. Although the relative lack of structure could indicate recent divergence, the magnitude of genetic divergence between the two inferred populations suggests a more ancient split between them. The widespread distribution of *V. maculicauda* and its larger body size suggest that high levels of migration may have allowed the dispersal of both genetic populations throughout the region.

Prior research in two of these species using mitochondrial data (Marchio & Piller, 2013; McMahan et al., 2017) revealed very low divergence between sampled populations, and so the results in this study agree with this past work in that regard, since the number of distinct

populations detected was low. The phylograms produced by Marchio and Piller (2013) showed two clades of *B. belizanus* that break near the Rio Grande in southern Belize, with one clade containing all samples north of the Rio Grande, and another clade containing samples to the south. The more fine-scaled RADSeq data presented in this study is roughly in agreement, though the break in the plot generated from STRUCTURE shows the clade separation to be north of the Rio Grande, around Stann Creek in Belize. This region around Stann Creek and Punta Gorda is of interest, as the STRUCTURE plot shows that admixture may be occurring.

Samples of *V. maculicauda* showed no concordance along biogeographic boundaries; indeed, the recovered plot from STRUCTURE shows that populations of *V. maculicauda* are co-occurring throughout its distribution with no detectable hybridization. The presence of more than one population contrasts with the mitochondrial results reported by McMahan et al. (2017), which showed minimal genetic divergence across its distribution. The differentiation of *V. maculicauda* with no admixture suggests that these two populations are distinct and are not interbreeding. The magnitude of genetic divergence between these populations suggests that they have been genetically isolated for some time, despite being sympatric. Future research into the causes of this divergence may answer targeted questions relating to dispersal and historical divergence, particularly if a coalescence-based approach is taken to find when and where these populations separated. Questions regarding the cytogenetics of sampled *V. maculicauda* populations may also prove fruitful. It may be the case that these two recovered populations are karyomorphs (cannot hybridize with one another), which may explain this possibly syntopic species pair. Genetic divergence related to karyotype evolution has been

reported previously for Neotropical fishes (Aguiar de Oliveira et al., 2015; Bertollo et al., 2000), though our results may be the first reported case of Middle American cichlid karyomorphs.

Mitochondrial and genomic data generated in this study showed four and three clades of *G. nicaraguensis* samples, respectively. A large region of the Caribbean slope of Honduras containing individuals from the Motagua river to Guaimoreto admixture is seen from this population through population 2. An intermediate region of containing individuals from the Prinzapolka river to Karatá in Nicaragua (population 2) contains genetic admixture from all three populations. Given the admixture of all three populations within population 2, it is perhaps unsurprising that this geographic region is the Mosquitia floodplain, a large drainage system where seasonal flooding may be introducing populations to one another. Of the three discovered populations, the population in Costa Rica (population 3) is the most distinct, and represents the southernmost sampling of the species. Future work in *G. nicaraguensis* should include RADSeq data for samples from populations in Panama and Honduras, as a general increase in samples would be a benefit to questions related to its biogeographic history.

The use of RADSeq markers can be powerful tools in understanding the biogeographical history of species. These fine-scale units can help to uncover ecological, geological, and evolutionary patterns that are overlooked at more coarse scales, as with microsatellites (Bohn et al., 2013), as well as underlying drivers of evolution, such as gene flow, genetic drift, and selection (Loiselle et al., 1995). Programs like `STRUCTURE` are designed to infer how many groups of individuals a given dataset contains and to show which samples belong to which group, but these programs are not without limitations. As highlighted by Porras-Hurtado et al. (2013), these programs struggle with the detection of clusters that are strongly under sampled,

detecting ancestral admixture that is shared by all individuals of a cluster, and handling individuals that are highly inbred or are missing large amounts of data. High quality data and larger samples can overcome some of these limitations, and when used in conjunction with other analyses (morphological, behavioral, genetic, etc.) can be informative and powerful tools in answering questions of population subdivision and gene flow (Marchio & Piller, 2013, McMahan et al., 2017). Additional work to better understand the ability of secondary freshwater fishes to use the coast for dispersal (and the frequency with which this occurs) would also be beneficial in illuminating the biogeographic history of targeted species, and of the region in general.

While sampling gaps do exist -- difficulties in sampling (both physical and geopolitical) combined with financial costs associated with population-scale studies are significant challenges -- the present study included a robust coverage of the Caribbean slope of Middle America for all species, with total geographic coverage extending from the Yucatán in Mexico to the Matina drainage in Costa Rica. In this study, we have found evidence of significant population structure in all three species examined, highlighting the power of genomic methods to detect genetic divergence among closely related populations.

References

Andrews, S. (2010). FastQC: A Quality Control Tool for High Throughput Sequence Data [Online]. Available online at: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

Avise, J., Arnold, J., Ball, R., Bermingham, E., Lamb, T., Neigel, J., . . . Saunders, N. (1987). Intraspecific Phylogeography: the mitochondrial DNA bridge between population genetics and systematics. *Annual Review of Ecology and Systematics*, 18, 489-522.

Baird, N., Etter, P., Atwood, T., Currey, M., Shiver, A., Lewis, Z., . . . Johnson, E. (2008). Rapid SNP discovery and genetic mapping using sequenced RAD markers. *PLoS ONE, 3*(10).

Bermingham, E., & Avise, J. (1986). Molecular zoogeography of freshwater fishes in the southeastern United States. *Genetics*. 113(4), 939-65.

Bermingham, E., & Martin, A. (1998). Comparative mtDNA phylogeography of neotropical freshwater fishes: testing shared history to infer the evolutionary landscape of lower Central America. *Molecular Ecology*, 7, 499-517.

Bertollo, L., Born, G., Dergam, J., Fenocchio, A., & Moreira-Filho, M. (2000). A biodiversity approach in the neotropical Erythinidae fish, *Hoplias malabaricus*. Karyotopic survery, geographic distribution of cytotypes and cytotaxonomic considerations. *Chromosome Research*, 8, 603-613.

Bohn, S., Barraza, E., McMahan, C.D., Matamoros, W., & B. Kreiser. (2013) Cross amplification of microsatellite loci developed for alligator gar (*Atractosteus spatula*) in tropical gar (*Atractosteus tropicus*). *Revista Mexicana de Biodiversidad,* 84: 1349-1351.

Bouckaert R., Vaughan T., Barido-Sottani, J., Duchêne S., Fourment M., Gavryushkina A., et al. (2019) BEAST 2.5: An advanced software platform for Bayesian evolutionary analysis. *PLoS computational biology*, *15*(4), e1006650.

Brewer, Cynthia A., (2014). http://www.ColorBrewer.org, accessed 2020

Brown, J., & Lomolino, M. (1998). *Biogeography* (2nd Ed.). Sunderland, Massachusettes (Sinauer Associates, Inc. Publishers). ISBN 0-87893-073-6.

Bussing, W. (1998). *Peces de las aguas continentales de Costa Rica* (2nd Ed.). (Editorial de la Universidad de Costa Rica, San Jose). ISBN 978-9977674896.

Chakrabarty, P. (2006). Systematics and historical biogeography of Greater Antillean Cichlidae. *Molecular Phylogenetics and Evolution, 39*(3), 619-627.

Chakrabarty, P., Faircloth, B., Alda, F., Ludt, W., McMahan, C., Near, T., . . . Alfaro, M. (2017). Phylogenomic systematics of ostariophysan fishes: ultraconserved elements support the surprising non-monophyly of Characiformes. *Systematic Biology, 66*(6), 881-895.

Chapin III, F., Zavaleta, E., Eviner, V. *et al.* (2000). Consequences of changing biodiversity. *Nature* 405, 234–242.

Danecek, P., Auton, A., Abecasis, G., Albers, C.A., Banks, E., DePristo, M.A., Handsaker, R.E., Lunter, G., Marth, G.T., Sherry, S.T., McVean, G., Durbin, R. (2011). 1000 Genomes Project Analysis Group. The variant call format and VCFtools. *Bioinformatics*. (15), 2156-2158.
Dray, S. & Dufour, A.B. (2007). The ade4 Package: Implementing the Duality Diagram for Ecologists. J*ournal of Statistical Software*, 1-20.

Drummond, A., Ho, S., Phillips, M. & Rambaut, A. (2006). Relaxed Phylogenetics and Dating with Confidence. *PLoS Biology*, *4*(88).

Drummond, A. & Rambaut, A. (2007). BEAST: Bayesian evolutionary analysis by sampling trees. *BMC Evolutionary Biology*, 7:214.

Earl, D., von Holdt, B. (2015). PopART: Full-feature software for haplotype network construction. *Methods Ecol. Evol*. *6*(9):1110–1116.

Eaton D. & Overcast I. (2020). ipyrad: Interactive assembly and analysis of RADseq datasets. *Bioinformatics*, *36*(8): 2592-2594.

Emerson, B., Hewitt, G. (2005). Phylogeography. *Current Biology*. 15, 367-371.

Family Cichlidae - Cichlids. (2012). Retrieved from https://www.fishbase.se/summary/FamilySummary.php?ID=349

Francis, R. (2017). POPHELPER: an R package and web app to analyse and visualize population structure. *Molecular Ecology Resources*, *17*(1), 27-32.

Goudet, J. (2005). HIERFSTAT, a package for R to compute and test hierarchical F-statistics. *Molecular Ecology Notes*. 5. 184-186.

Grünwald N. & Goss E. (2011). Evolution and population genetics of exotic and re-emerging pathogens: novel tools and approaches. *Annu. Rev. Phytopathology*, 49, 249-267.

Günther, A. (1864). *A catologue of the fishes in the British Museum* (Vol. 6 ed.) London, England. The Order of the Trustees.

Harrison, E., Trexler, J., Collins, T., Vazquez-Domínguez, E., Razo-Mendivil, U., Matamoros, W., & Barrientos, C. (2014). Genetic evidence for multiple sources of the non-native fish *Cichlasoma urophthalmus* (Günther; Mayan Cichlids) in southern Florida. *PLoS ONE, 9*(9).

Jombart T. (2008) Adegenet: a R package for the multivariate analysis of genetic markers. *Bioinformatics*. 24, 1403 – 1405.

Jombart, T., Ahmed, I. (2011). adegenet 1.3-1: new tools for the analysis of genome-wide SNP data. *Bioinformatics*, *27*(21), 3070–3071.

Jombart, T., & Collins, C. (2015). A tutorial for discriminant analysis of principal components (dapc) using adegenet 2.0.0. Available at: http://adegenet.r-forge.r-project.org/files/tutorial-dapc-pdf

Jombart, T, Devillard, S, Balloux, F. (2010) Discriminant analysis of principal components: a new method for the analysis of genetically structured populations. *BMC Genetics*, *11*(94).

Kamvar, Z., Tabima, J., Grünwald, N. (2014) Poppr: an R package for genetic analysis of populations with clonal, partially clonal, and/or sexual reproduction. PeerJ., *4*(2)e281.

Knaus, B., Grünwald, N. (2017). VCFR: a package to manipulate and visualize variant call format data in R. *Molecular Ecology Resources*, 17(1), 44–53. ISSN 757.

Kner, R. (1860). Uber *Belonesox belizanus*, nov. gen. et. spec., aus der Familie der Cyprinodonten. *Sitzungsberichte Akademie der Wissenschaften Wien. 40*, 419-422.

Kullander, S. (1998). A phylogeny and classification of the South American Cichlidae (Teleostei: Perciformes). *EdiPUCRS*, 461-498.

Leigh, J., Bryant, D. (2015). PopART: Full-feature software for haplotype network construction. *Methods Ecol. Evol.*, 6(9):1110–1116.

Lischer, H., and Excoffier, L. (2012) PGDSpider: An automated data conversion tool for connecting population genetics and genomics programs. *Bioinformatics*, 28: 298-299.

Loiselle, B., Sork, V., Nason, J., & Graham, C. (1995). Spatial Genetic Structure of a Tropical Understory Shrub, *Psychotria officinalis* (Rubiaceae). *American Journal of Botany, 82*(11), 1420-1425.

Lucinda, P. (2003). *Poeciliidae, Livebearers* in *Reis, R., Kullander, S., & Ferraris, C. (ed):* Check list of the freshwater fishes of South and Central America*.* Porte Alegre*. EdiPUCRS*, 555-581.

Malabarba, L. & Malabarba, M. (2019). Phylogeny and classification of Neotropical fish, Chapter 1. *Biology and Physiology of freshwater Neotropical fish*. Academic Press.

Malabarba, L., Reis R., Vari R., Lucena Z., & Lucena, C. (1998). Phylogeny and classification of neotropical fish*. EdiPUCRS*, Porto Alegre, 461-469.

Manokaran, N. (1992). An Overview of Biodiversity in Malaysia. *Journal of Tropical Forest Science*, (5)2, 271–290.

Marchio, E., & Piller, K. (2013). Cryptic diversity in a widespread live-bearing fish (Poeciliidae: *Belonesox*). *Biological Journal of the Linnean Society, 109*(4), 848-860.

Matamoros, W., McMahan, C., Chakrabarty, P., Albert, J., & Schaefer, J. (2014). Derivation of the freshwater fish fauna of Central America revisited: Myers's hypothesis in the twenty-first century. *Cladistics, 31*(2), 177-188.

McMahan, C., Ginger, L., Cage, M., David, K., Chakrabarty, P., Johnston, M., & Matamoros, W. (2017). Pleistocene to holocene expansion of the black-belt cichlid in Central America, Vieja maculicauda (Teleostei: Cichlidae). *PLoS ONE, 12*(5).

Meier, J., Marques, D., Mwaiko, S., Wagner, C., Excoffier, L., & Seehausen, O. (2017). Ancient hybridization fuels rapid cichlid fish adaptive radiations. *Nature Communications, 8*.

Miller, R. (1966). Geographical distribution of Central American freshwater fishes. *Copeia* (4), 773-802.

Montes, C., Cardona, A., Jaramillo, C., Pardo, A., Silva, J.C., Valencia, V., Ayala, C., Pérez-Angel, L.C., Rodriguez, L.A. (2015). Middle Miocene closure of the Central American Seaway. *Science*, 348(6231), 226-229.

Myers, G. (1966). Derivation of the freshwater fish fauna of Central America*. Copeia* (4), 766-773.

Myers, N., Mittermeier, R., Mittermeier, C. *et al.* (2000). Biodiversity hotspots for conservation priorities. *Nature,* 403, 853–858.

de Oliveira, E., Bertollo, L., Yano, C. *et al*. Comparative cytogenetics in the genus *Hoplias* (Characiformes, Erythrinidae) highlights contrasting karyotype evolution among congeneric species. Molecular Cytogenetetics *8*(56), 1-10.

Paradis, E. & Schliep K. (2019). ape 5.0: an environment for modern phylogenetics and evolutionary analyses in R. *Bioinformatics*, *35*(3), 526–528.

Peterson, B., Weber, J., Kay, E., Fisher, H., & Hoekstra, H. (2012). Double digest RADseq: An inexpensive method for de novo SNP discovery and genotyping in model and non-model species. *PLoS ONE, 7*(5).

Poland, J., Brown, P., Sorrells, M., & Jannink, J. (2012). Development of high-density genetic maps for barley and wheat using a novel two-enzyme genotyping-by-sequencing approach. *PLoS ONE, 7*(2).

Porras-Hurtado, L., Ruiz, Y., Santos, C., Phillips, C., Carracedo, A., & Lareu, M. (2013). An overview of STRUCTURE: applications, parameter settings, and supporting software. Frontiers in Genetics, *4*(98), 1-13.

Pritchard, J. K., Stephens, M., & Donnelly, P. (2000). Inference of population structure using multilocus genotype data. *Genetics*, 155, 945–959

Purcell, S., Neale, B., Todd-Brown, K., *et al*. (2007). PLINK: a toolset for whole-genome association and population-based linkage analysis. *American Journal of Human Genetics*, 81.

Rambaut, A. (2018). FigTree v1.4.4 (http://tree.bio.ed.ac.uk/software/figtree/).

Sparks, J., & Smith, W. (2005). Freshwater fishes, dispersal ability, and nonevidence: "Gondwana life rafts" to the rescue. *Systematic Biology, 54(*1*)*, 158-165.

Tamura, K., Stecher, G., Peterson, D., Filipski, A., & Kumar, S. (2013). MEGA6: Molecular Evolutionary Genetics Analysis version 6.0. *Molecular Biology and Evolution*, 30, 2725-2729.

Webb, S. (2006). The Great American Biotic Interchange: Patterns and Processes. *Annals of the Missouri Botanical Garden*, *93*(2), 245-257.

Weir, B. & Cockerham, C. (1984). ESTIMATING F-STATISTICS FOR THE ANALYSIS OF POPULATION STRUCTURE. *Evolution*, *38*, 1358-1370.

Weir, B., Goudet, J. (2017). A Unified Characterization of Population Structure and Relatedness. *GENETICS*, 206(4), 2085-2103.

Weir, B., Hill, W. (2002). Estimating F-statistics. *Annu. Rev. Genet.*, 36:721-750.

Wickham, H. (2007). Reshaping Data with the reshape Package. Journal of Statistical Software [Online], 21.12 (2007): 1 - 20.

Wickham H (2016). ggplot2: Elegant Graphics for Data Analysis. *Springer-Verlag*.

Winker, K. (2011). Middle America, not Mesoamerica, is the Accurate Term for Biogeography. *The Condor, 113*(1), 5-6.

Supplemental Files

G. nicaraguensis files:

A non-model based UPGMA tree of *G. nicaraguensis* was generated, with individuals colored based on what drainage they were sampled from. This UPGMA tree reveals two large distinct groups and two smaller groups, with some admixture seen (Figure S1). A PCA was generated, with eigenvalues displayed as a bar graph (Figure S2).

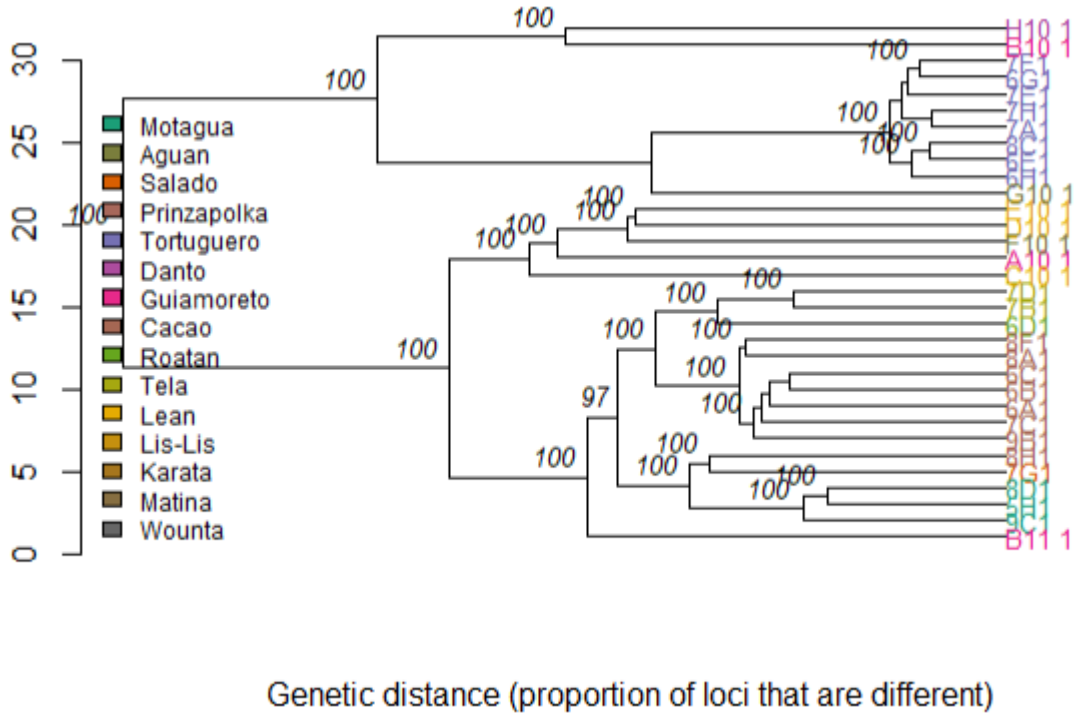


Figure S1: UPGMA tree of *G. nicaraguensis*. Two large distinct groups and two smaller groups can be seen, with some admixture throughout.

A PCA was generated, with eigenvalues displayed as a bar graph (Figure S2).

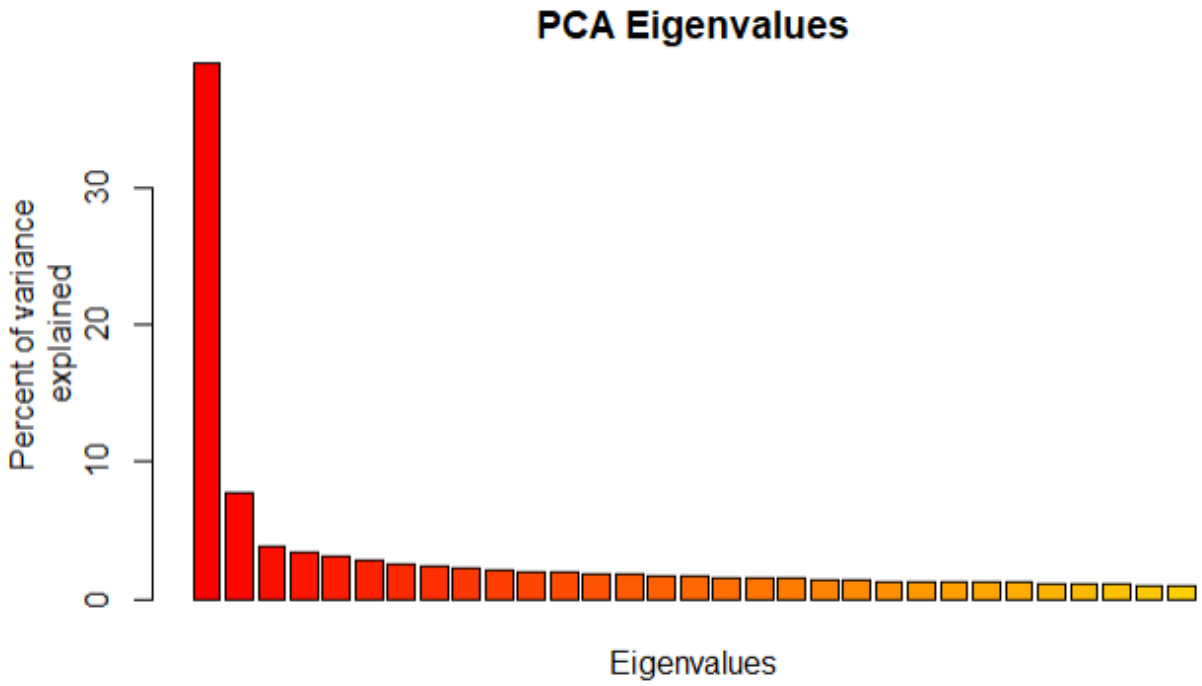


Figure S2: A barplot of principal component analysis eigenvalues for *G. nicaraguensis*.

A sharp decrease in Delta K was observed with the increase of K using a subsampling of 25 iterations of STRUCTURE (Figure S3). This decrease stopped at K = 4, with a notable increase of Delta K at K = 5. The optimal K-value indicates that three populations showed the highest probability for population clustering. In addition, there was a small peak observed at K = 6 before the gradual decrease in Delta K for the rest of the K values presented, which might indicate another informative instance of population structure. Running Structure Harvester with the complete set of STRUCTURE runs (n=100) showed a clear peak at K = 4, with all other values near zero. Therefore, the STRUCTURE results at both K = 3 and K = 4 were subject to the following population genetics analyses. These results can be viewed as an Evanno table in Figure S4.

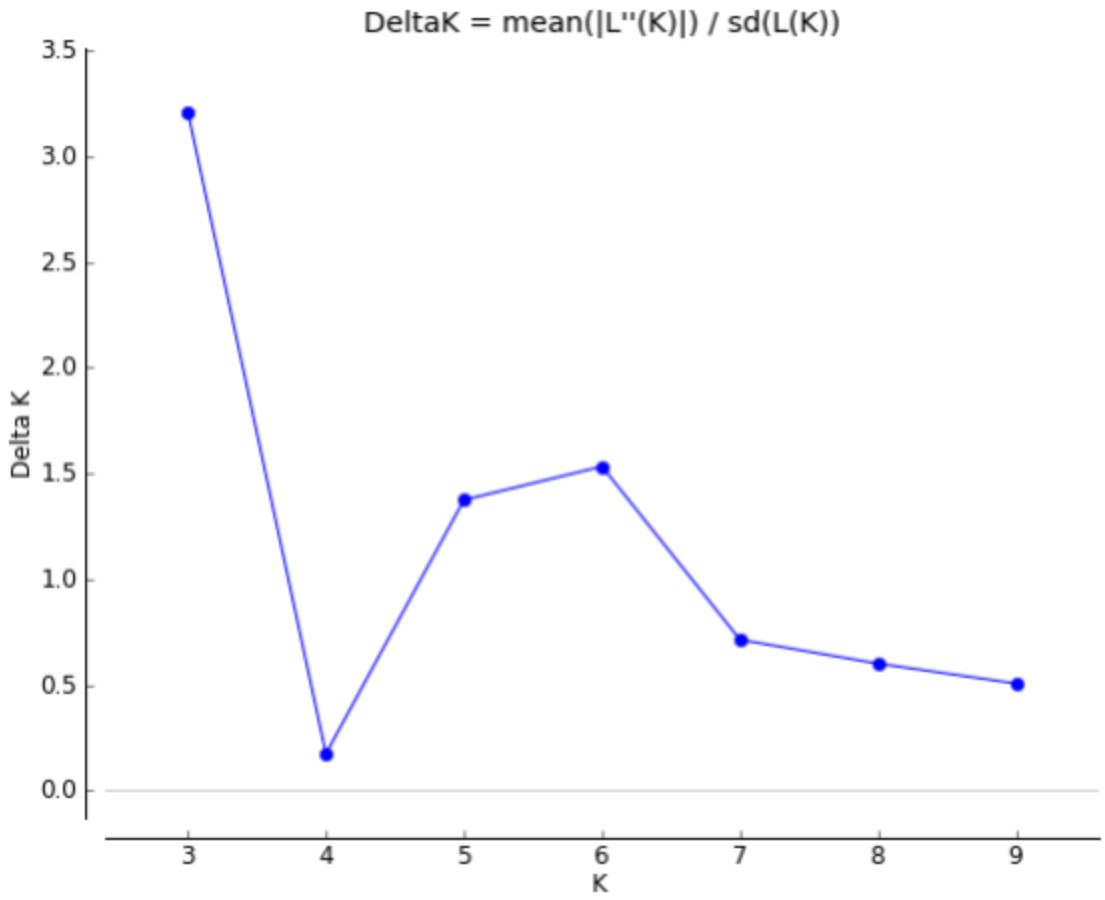


Figure S3: A line graph showing delta K values for *G. nicaraguensis*. The optimal value corresponds to K = 4.

K	Reps	Mean LnP(K)	Stdev LnP(K)	Ln'(K)	|Ln''(K)|	Delta K
2	24	-165653.900000	1231.045794	—	—	—
3	25	-118860.568000	30047.979091	46793.332000	96360.841273	3.206899
4	22	-168428.077273	201743.963722	-49567.509273	35100.333455	0.173985
5	10	-253095.920000	340047.812323	-84667.842727	467268.787273	1.374127
6	10	-805032.550000	505044.764570	-551936.630000	773857.340000	1.532255
7	10	-583111.840000	618641.742822	221920.710000	441196.680000	0.713170
8	10	-802387.810000	809490.595885	-219275.970000	485711.210000	0.600021
9	10	-535952.570000	484133.753121	266435.240000	244947.540000	0.505950
10	10	-514464.870000	291709.596232	21487.700000	—	—

Figure S4: An Evanno table for *G. nicaraguensis*. Peaks in Delta K at K = 3 and K = 6 were found. Columns correspond to (from left to right): K, the number of reps per K, the mean log probability of K, the standard deviation of the log probability of K, the log of K, the absolute value log of K, and delta K.

A PCA was generated containing the first and third PC's (Figure S5).

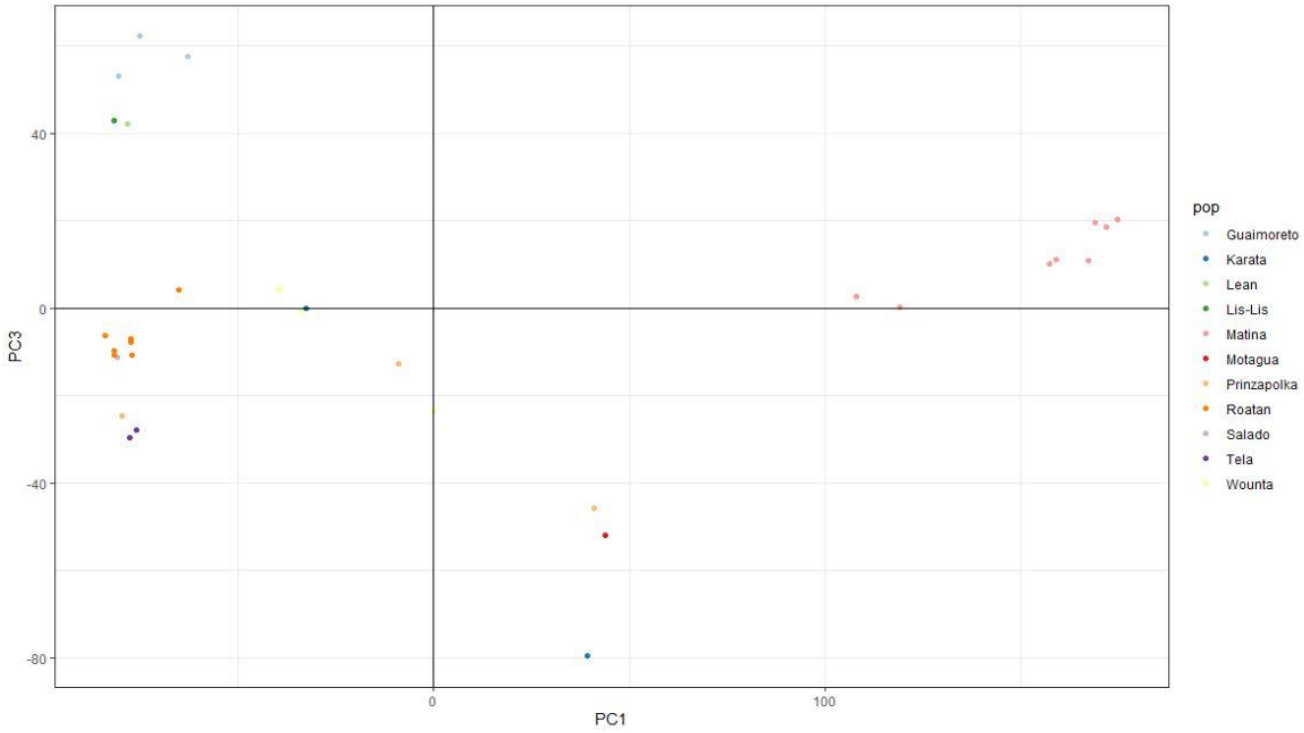


Figure S5: Principal component analysis of the processed RADSeq data for *G. nicaraguensis*. Individuals are labeled based on the drainage in which they were collected. The first and third principal components are used.

B. belizanus files:

A non-model based UPGMA tree of *B. belizanus* was generated, with individuals colored based on what drainage they were sampled from. This UPGMA tree did not reveal distinct groups (Figure S6). A PCA was generated, with eigenvalues displayed as a bar graph (Figure S7).

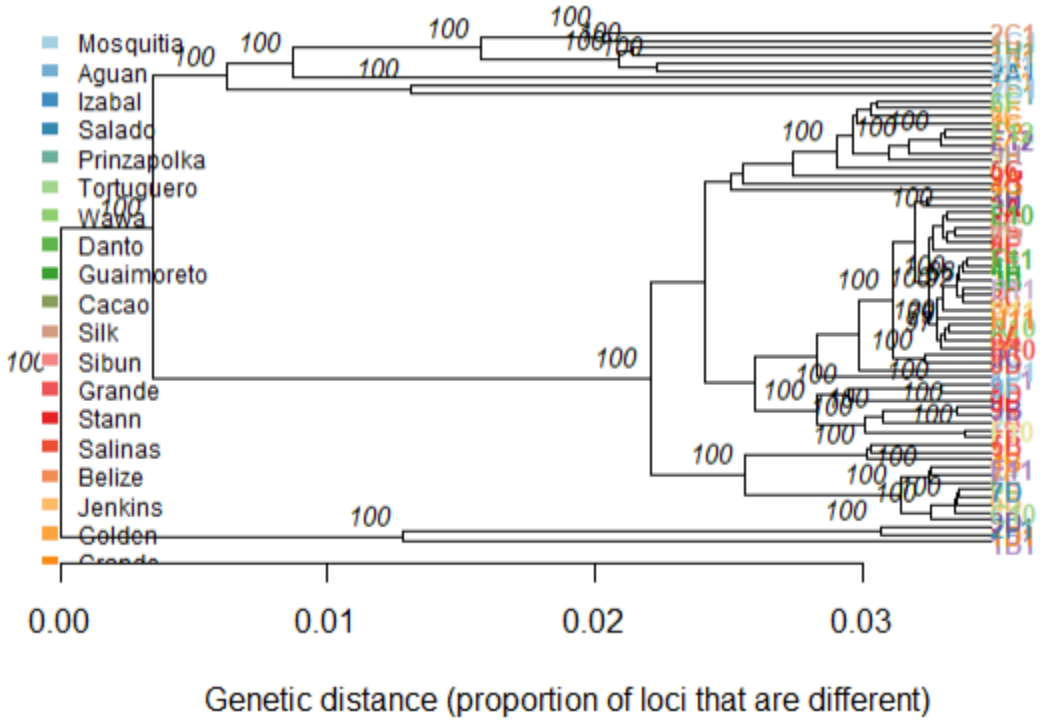


Figure S6: UPGMA tree of *B. belizanus*. No distinct groupings of drainages or individuals can be seen.

A PCA was generated, with eigenvalues displayed as a bar graph (Figure S7).

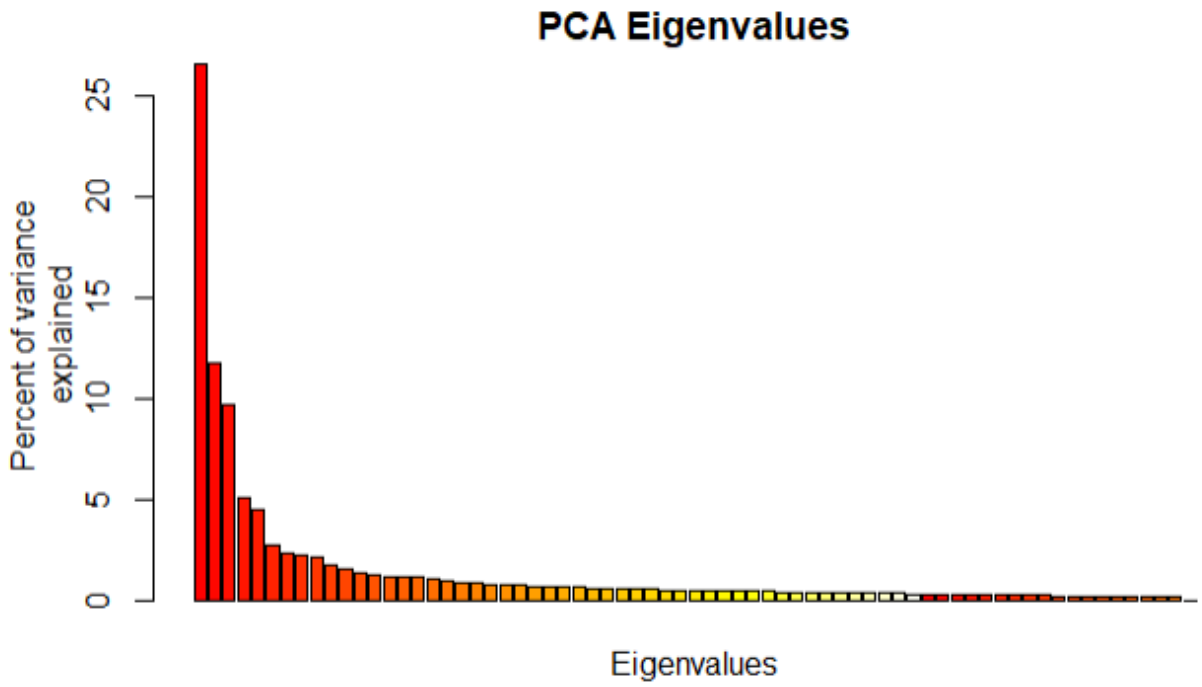


Figure S7: A barplot of principal component analysis eigenvalues for *B. belizanus*.

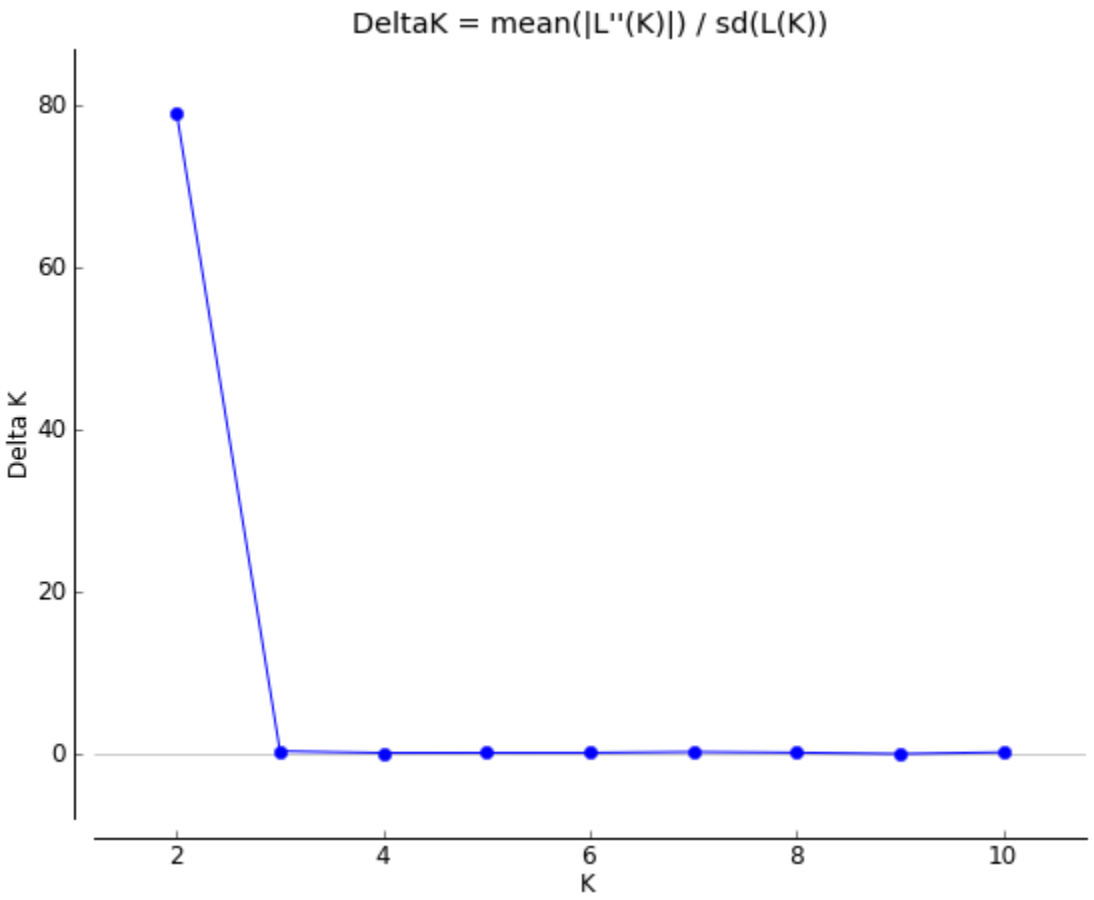


Figure S8: A line graph showing delta K values for *B. belizanus*. The optimal value corresponds to K = 2.

An initial peak in Delta K was observed with the increase of K. This peak corresponds to a K = 2, indicating that two populations showed the highest probability for clustering. A sharp decrease in Delta K occurred after K = 2 and remained near zero from K = 3 to K = 10. These results can be viewed as an Evanno table in Figure S9.

K	Reps	Mean LnP(K)	Stdev LnP(K)	Ln'(K)	|Ln''(K)|	Delta K
1	100	-14904.803000	241.906571	—	—	—
2	100	-1279.577000	169.438319	13625.226000	13378.943000	78.960551
3	100	-1033.294000	397.999633	246.283000	139.270000	0.349925
4	100	-926.281000	455.424566	107.013000	64.255000	0.141088
5	100	-883.523000	444.327589	42.758000	67.393000	0.151674
6	100	-908.158000	450.292848	-24.635000	69.877000	0.155181
7	100	-862.916000	457.827428	45.242000	122.763000	0.268143
8	100	-940.437000	453.122549	-77.521000	77.516000	0.171071
9	100	-940.442000	460.397826	-0.005000	10.778000	0.023410
10	100	-951.225000	448.778546	-10.783000	98.214000	0.218847
11	100	-863.794000	447.419634	87.431000	—	—

Figure S9: An Evanno table for *B. belizanus*. The optimal value is highlighted in yellow. Columns correspond to (from left to right): K, the number of reps per K, the mean log probability of K, the standard deviation of the log probability of K, the log of K, the absolute value log of K, and delta K.

A PCA was generated containing the first and third principal components (Figure S10).

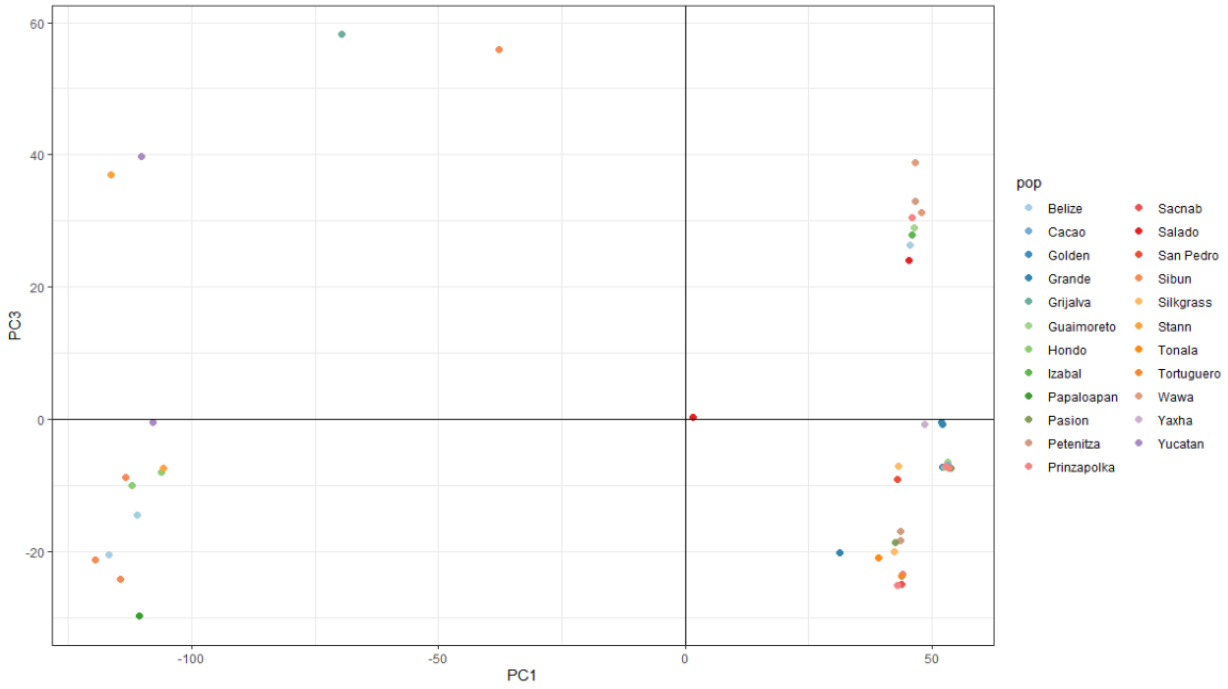


Figure S10: Principal component analysis of the processed RADSeq data for *B. belizanus*. Individuals are labeled based on the drainage in which they were collected. The first and third principal components are used.

V. maculicauda files:

A non-model based UPGMA tree of *V. maculicauda* was generated, with individuals colored based on what drainage they were sampled from. This UPGMA tree reveals two large distinct groups smaller groups, with admixture seen (Figure S11).

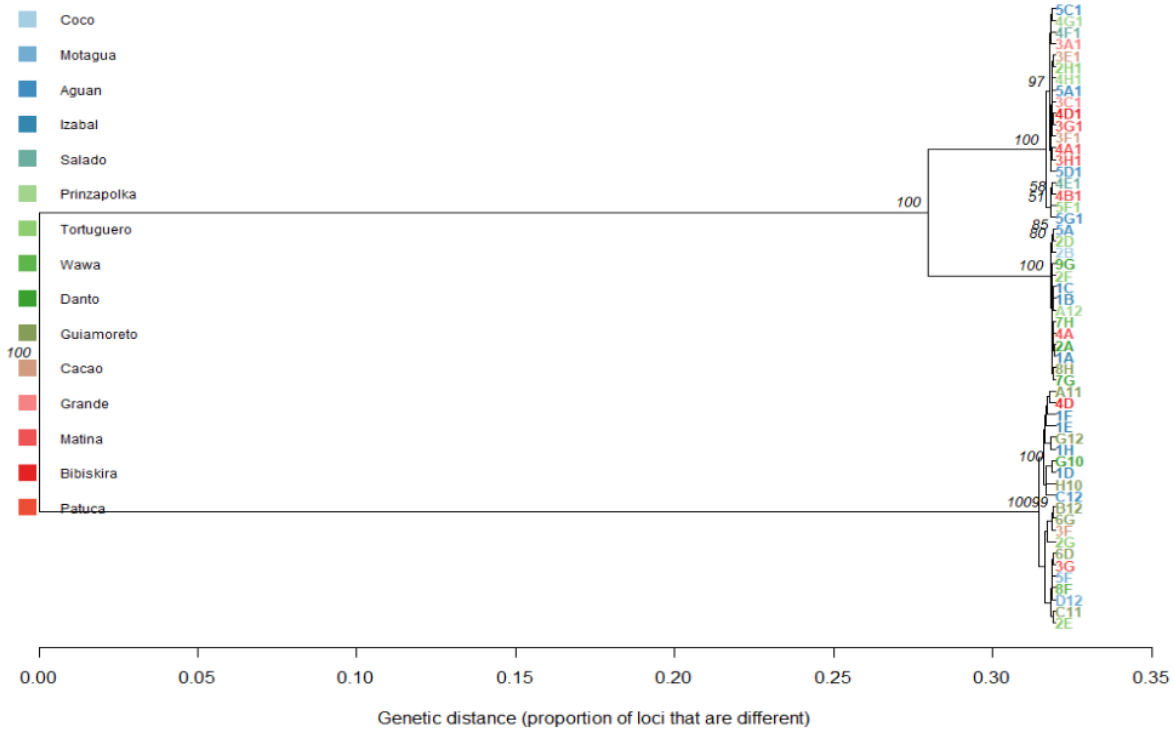


Figure S11: UPGMA tree of *V. maculicauda*.

The first 7 principal components were used (Figure S12) and 3 discriminant functions were saved as an object in RStudio.

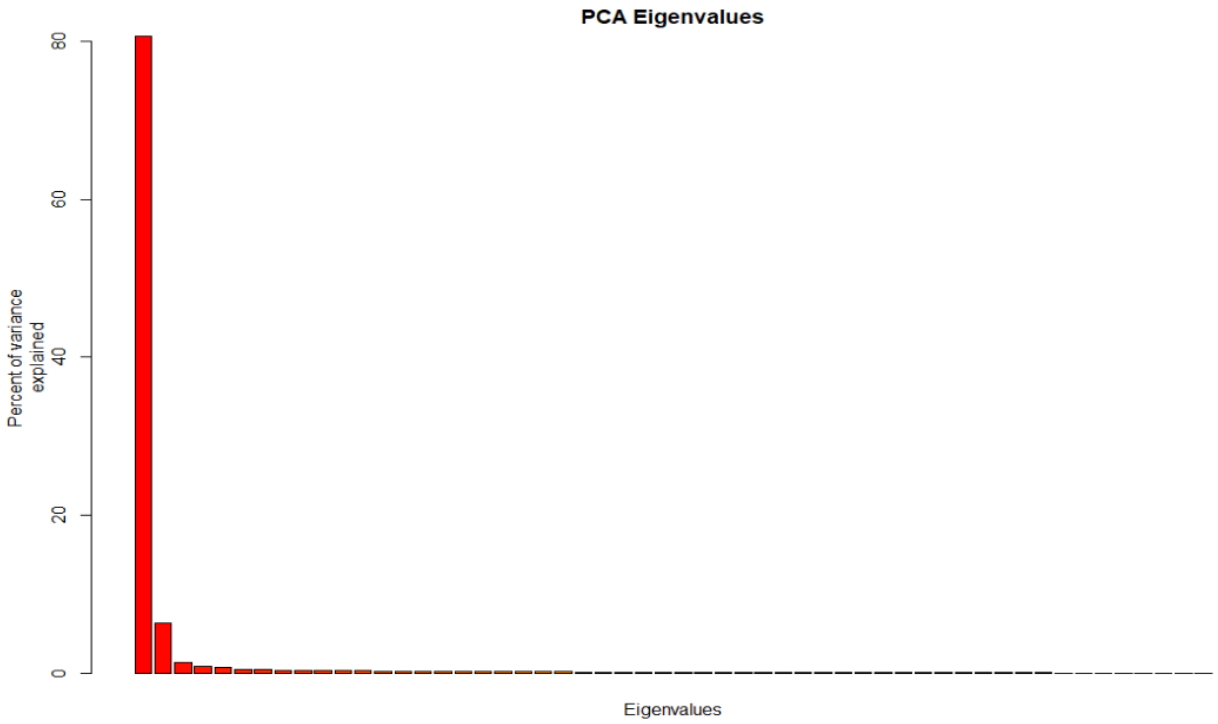


Figure S12: A barplot of principal component analysis eigenvalues for *V. maculicauda*.

The K-value was used to estimate the number of clusters of the accessions based on the SNP data. In order to find the optimal K-value, the number of clusters (K) was plotted against ΔK, which showed a sharp peak at K = 2 (Figure S13). A sharp decrease was observed in ΔK with the increase of K as it approached K =3, followed by a less pronounced decrease at K = 3 to K = 4. Following this, the value for ΔK remained near zero for the remainder of the graphed values. The optimal K-value indicates that two populations showed the highest probability for population clustering (Figure S14). Since there is a clear optimal value for K, only the STRUCTURE result of K = 2 was subject to further analysis.

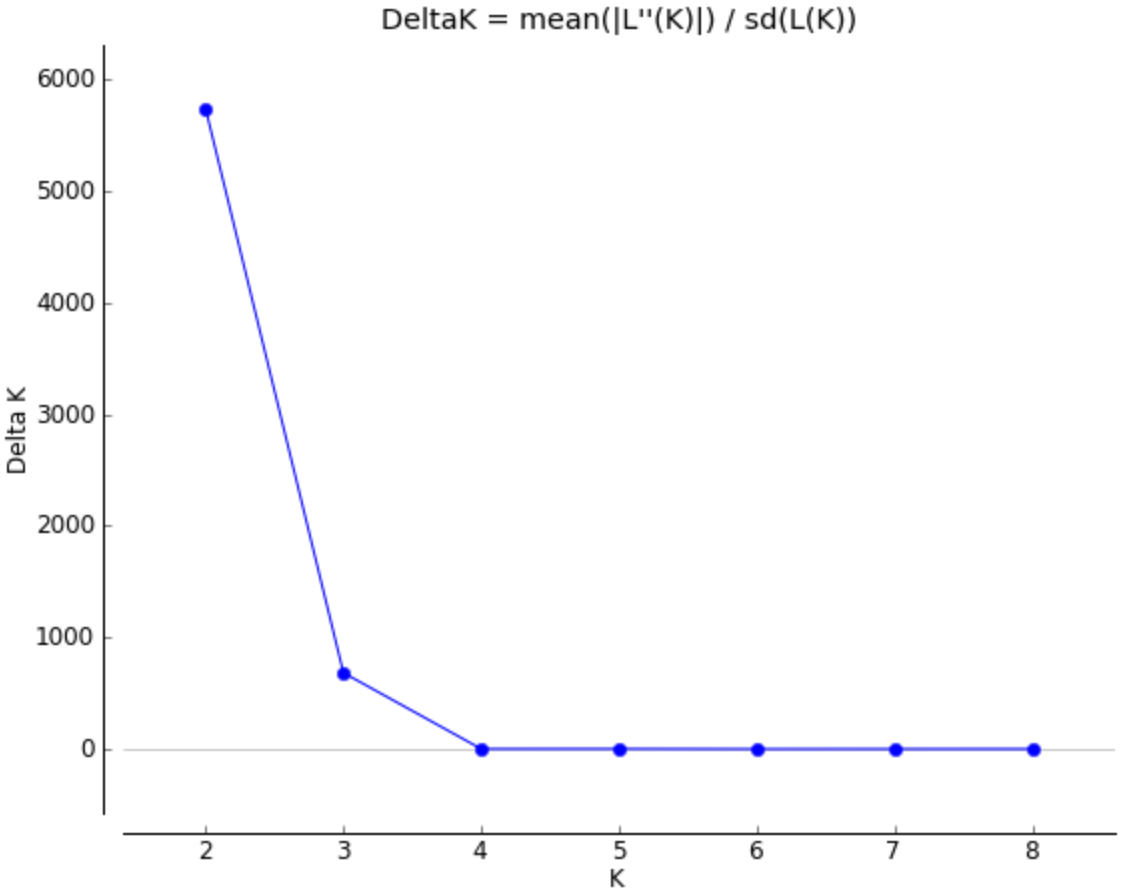


Figure S13: A line graph showing delta K values for *V. maculicauda*.

K	Reps	Mean LnP(K)	Stdev LnP(K)	Ln'(K)	|Ln''(K)|	Delta K
1	10	-936430.370000	60.807840	—	—	—
2	10	-420010.190000	82.068284	516420.180000	470986.980000	5738.964648
3	10	-374576.990000	5709.693079	45433.200000	3901087.520000	683.239443
4	10	-4230231.310000	6559272.227592	-3855654.320000	6603722.410000	1.006777
5	10	-1482163.220000	2140969.066131	2748068.090000	2902313.570000	1.355607
6	10	-1636408.700000	3831471.619883	-154245.480000	357163.210000	0.093218
7	10	-2147817.390000	5189105.628275	-511408.690000	2005244.550000	0.386434
8	10	-4664470.630000	7596358.910669	-2516653.240000	4847174.770000	0.638092
9	5	-2333949.100000	2819203.592188	2330521.530000	—	—

Figure S14: An Evanno table for *V. maculicauda*. The optimal value for K has been highlighted in yellow. Columns correspond to (from left to right): K, the number of reps per K, the mean log probability of K, the standard deviation of the log probability of K, the log of K, the absolute value log of K, and delta K.

A principal component analysis was generated containing the first and third principal components (Figure S15).

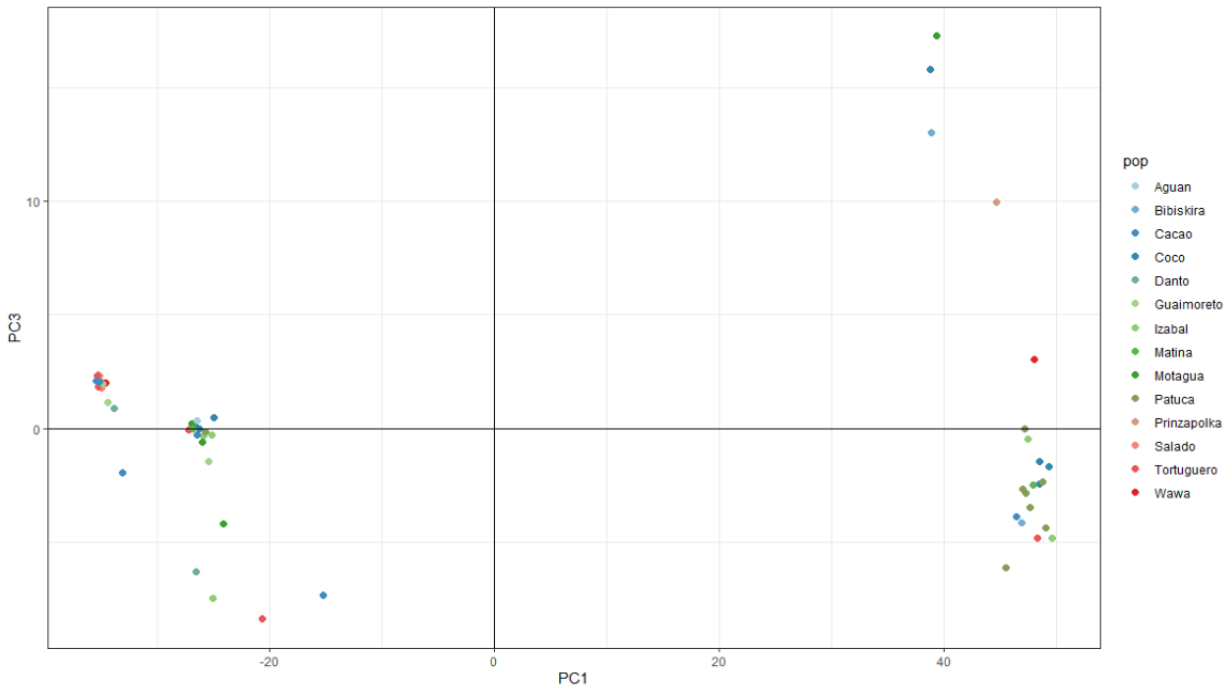


Figure S15: Principal component analysis of the processed RADSeq data for *V. maculicauda*. Individuals are labeled based on the drainage in which they were collected. The first and third principal components are used.

Bioinformatics scripts and guides:

This section contains all scripts and guides used for bioinformatic analyses of the RADSeq data used in the completion of this thesis. Subheadings will indicate what platform each script corresponds to, and the scripts will be presented in a format that present and future readers can copy and paste directly into the relevant platform and have a functioning bioinformatics script. While each of the three focal species required the use of scripts to process and analyze, only one copy of each script has been added to this section. This is because the only variables that change from each of the species are the input VCF files, and the transformed data stemming from that VCF file.

A very basic understanding of R, the command line, and the use of notebooks and scripts is *strongly* suggested *before* attempting to work with these scripts. Navigating through a terminal, understanding the layout of RStudio, and creating and executing notebook commands are just a few of the essential skills that are absolutely required prior to analyzing bioinformatic datasets. Courses, textbooks, and online lessons are all an effective means to this end.

Python/CLI:

ADMIXTURE:

Below is a command line guide for the inference of population structure and individual ancestries. ADMIXTURE is a clustering software like STRUCTURE, and requires unliked SNPs that have been transformed into .bed format via the Plink software. It is worth noting that ADMIXTURE runs *much* more quicky than STRUCTURE, and can be used in conjunction with or as a replacement to STRUCTURE as an additional or alternate source of population structure analysis.

In [1]:

First, make a directory for ADMIXTURE files:

```
mkdir Admixture

cd Admixture
```

In [2]:

Transform your input SNP file to plink format. Make sure this VCF file is in the Admixture directory:

```
plink --vcf yourvcffile.vcf --make-bed --out
youroutfile --allow-extra-chr --max-alleles 2
```

In [3]:

ADMIXTURE does not accept chromosome names that are not human chromosomes. You can get around this by exchanging the first column with zero:

```
awk '{$1=0;print $0}' youroutfile.bim >
youroutfile.bim.tmp
mv youroutfile.bim.tmp youroutfile.bim
```

In [4]:

Run admixture. This will produce two files. The first is a .Q file that contains the clustering assignments for individuals, and the second is .P, which contains allele frequencies for each population. This output file tests for K = 2. You can adjust this value for however many K you wish to test. You can do this by hand easily enough by just replacing the twos (highlighted below in green) with the desired value.

```
admixture --cv $youroutfile.bed 2 > log2.out
```

StructureHarvester:

Below is a command line script for determining the optimal value for K to use in visualizing genomic data. The term “script” is being used loosely – it is more of a guide that contains a single line of code. Nonetheless it can be very valuable, and is *essential* to know if you are working with very large SNP files that are not supported on the website version. For that reason, I have included it here.

After obtaining output files from STRUCTURE, you will have numerous data files containing different set values for K, as well as repetitions of that value for K. This is how you determine what value for K you should select for viewing in Distruct, CLUMPAK, PopHelper, etc.

In [1]:

Download StructureHarvester from http://alumni.soe.ucsc.edu/~dearl/software/struct_harvest/ and make sure the script is saved. The script should be `structureHarvester.py`.

In [2]:

Put your results folder (we will name this "Results" from STRUCTURE in the structureHarvester folder.

In [3]:

At your command prompt, type:

```
python structureHarvester.py --dir=Results --
out=Output --evanno
```

Then hit enter. This will create an output folder named "Output" that will contain two files. The first file is named evanno.txt and the second file is named summary.txt.

In [4]:

Open evanno.txt in Microsoft Excel. You will make a line graph with Delta K as your Y-axis and K as your X-axis.

VCFtools:

Below is a guide to using VCFtools. While executed via CLI, it has been written as a notebook to simplify reading and understanding the steps. This program is almost certainly going to be necessary when processing SNP data. Generally, you want to have samples that contain high coverage and therefore want to remove samples with large amounts of missing data.

VCFtools can do much more than just what I am writing below; as such, time should be spent reading the manual, as it can provide useful statistics and formatting for future analyses. Each step has been formatted to contain 1) an explanation of what is being done at each stage, and 2) the line(s) of code necessary to execute this section correctly.

In [1]:

First, download and install VCFtools, and make sure your VCF file is in your working directory. The following code gives you a list of all your individual samples and tells you how much data each sample is missing. This is viewed in the far-right column, which gives you the percentage that is missing, e.g. 0.89 would be 89% missing data, and should probably be filtered out:

```
vcftools --vcf yourdata.vcf --missing-indv

cat out.imiss
```

In [2]:

Next, create a plain text document with the names of all of the samples in your out.imiss file that had a higher percentage missing data than you want. Then, copy and paste this list into lowdepth.indv and save it:

```
nano lowdepth.indv
```

In [3]:

Next, specify the number of missing loci (I chose 80%). Anything that has more missing loci than that will get dropped. Remember to be careful that `max_missing` is defined to be between 0 and 1, where 0 allows sites that are completely missing and 1 indicates no missing data allowed -- so it is counterintuitive. If you use `--max_missing 0.8` you will retain only sites with less than 20% missing data. After filtering, you will have a new VCF file, which can be converted into other formats using PGDSpider. The output from this is named "oneout":

```
vcftools --vcf samepathtofile.vcf --remove
lowdepth.indv --max-missing 0.8 --recode --out
oneout
```

PGDSpider:

The following guide is for the Windows desktop version of PGDSpider2 rather than from the command line. This guide will assume an input file in VCF format exists, and will also assume that the desired output filetype is STRUCTURE.

In the PGDSpider window that launches when you open the application, click:

- Data input file → File format: VCF → Select input file

This will open Windows Explorer, where you can select your input VCF file. Next:

- Data output file → File format: STRUCTURE → Select output file

This will once again open Windows Explorer, where you can select the destination and name of your output file. Next:

- Convert

A new window will pop up in PGDSpider2 asking for additional parameters or writer questions regarding your input and output file. Default is usually sufficient for your input file, but for your output STRUCTURE file, do the following:

- STRUCTURE (optional) → data type: SNP

Rstudio:

Population genetics notebook script:

Below is a script used to analyze population structure using RStudio. The script is annotated to explain what is being done during each step. The indented portions of this script are actual code, while the undented descriptions are meant to be pasted into the textboxes of the notebook.

```
In [1]:

title: "Population Genetics Script"

Adapted from Grunwald lab website

Make sure the packages being called by the library() functions
have been installed. Use the `install.packages()` function to do
this if you haven't already. Remember to set your working
directory!

        library(vcfR)

        library(poppr)

        library(ape)

        library(RColorBrewer)

        library(igraph)

        library(ggplot2)
```

```
library(parallel)

library(pegas)

library(devtools)

library(ade4)

library(adegenet)

library(LEA)

library(mapplots)

library(irtoys)

library(hierfstat)

library(magrittr)

library(mmod)

library(treemap)

library(adegenet)

library(cluster)

library(factoextra)

library(ggplot2)

library(tidyr)
```

In [2]:

Below: Here we will read your VCF file, make a genind object, make a genlight object, make prepare to make a UPGMA tree.

```
yourdata.vcf <- read.vcfR("yourdata.vcf")

yourdata.vcf

yourdata.data <- read.table("yourdata.txt", sep
="\t", header = TRUE)

all(colnames(yourdata.vcf@gt)[-1] ==
yourdata.data$ID)

gl.yourdata <- vcfR2genlight(yourdata.vcf,
n.cores = 4)

gind.yourdata <- vcfR2genind(gnic.vcf)

gind.yourdata

ploidy(gl.yourdata) <- 2

pop(gl.yourdata) <- yourdata.data$Country

gl.yourdata

nrow(gl. yourdata)

gl. yourdata.dist <- dist(gl. yourdata)

gl. yourdata.dist <- poppr::bitwise.dist(gl.
yourdata)
```

In [3]:

Now we will build a genetic distance tree that represents the genetic relatedness of the samples. The similarity between samples and groups of samples is represented by the branch length. In most trees, the branch length is represented by the number of substitutions per site for a cluster or a sample. When samples are very similar, they are grouped by short branches. The longer the branch, the higher the number of substitutions and the higher the genetic distance is between samples or clusters.

We will reconstruct a distance tree based on the UPGMA algorithm, with 100 bootstrap replicates to assess branch support:

```
tree <- aboot(gl. yourdata, tree = "upgma",
distance = bitwise.dist, sample = 100, showtree =
F, cutoff = 50, quiet = T)

cols <- brewer.pal(n = nPop(gl. yourdata), name =
"Paired")

cols <- colorRampPalette(cols)(30)

plot.phylo(tree, cex = 0.8, font = 2, adj = 0,
tip.color =  cols[pop(gl. yourdata)])

nodelabels(tree$node.label, adj = c(1.3, -0.5),
frame = "n", cex = 0.8,font = 3, xpd = TRUE)
```

```
        legend('topleft', legend = c("Motagua","Aguan",
        "Salado", "Prinzapolka", "Tortuguero", "Danto",
        "Guiamoreto", "Cacao", "Roatan", "Tela", "Lean",
        "Lis-Lis", "Karata", "Matina", "Wounta"), fill =
        cols, border = FALSE, bty = "n", cex = 0.75)
        axis(side = 1)
        title(xlab = "Genetic distance (proportion of
        loci that are different)")
```

In [4]:

Principal components analysis

A principal component analysis (PCA) converts the observed SNP data into a set of values of linearly uncorrelated variables called principal components that summarize the variation between samples. We can perform a PCA on our genlight object by using the glPCA function.

```
        yourdata.pca <- glPca(gl. yourdata, nf = 11)
        yourdata.pca
        barplot(100*
        yourdata.pca$eig/sum(yourdata.pca$eig), col =
        heat.colors(50), main="PCA Eigenvalues")
```

```
        title(ylab="Percent of variance\nexplained", line
        = 2)

        title(xlab="Eigenvalues", line = 1)
```

In [5]:

The barplot indicates that we will need to only retain the first 10 PCAs, which cumulatively explain explain 66.6 percent of the variance of the data.

To view the results of the PCA we can use the package ggplot2. We need to convert the data frame that contains the principal components (rubi.pca$scores) into the new object gnic.pca.scores. In addition, we will add the population values as a new column in our gnic.pca.scores object, in order to be able to color samples by population.

ggplot2 will plot the PCA, color the samples by population, and create ellipses that include 95% of the data for each the population:

```
        yourdata.pca.scores <-
        as.data.frame(yourdata.pca$scores)

        yourdata.pca.scores$pop <- pop(gl. yourdata)
```

```
set.seed(9)

p <- ggplot(yourdata.pca.scores, aes(x=PC1,
y=PC2, colour=pop))

p <- p + geom_point(size=1.5)

p <- p + stat_ellipse(level = 0.95, size = 1)

p <- p + scale_color_manual(values = cols)

p <- p + geom_hline(yintercept = 0)

p <- p + geom_vline(xintercept = 0)

p <- p + theme_bw()

p
```

In [6]:

Discriminant Analysis of Principal Components (DAPC)

The DAPC is a multivariate statistical approach that uses populations defined a priori to maximize the variance among populations in the sample by partitioning it into between-population and within-population components. DAPC thus maximizes the discrimination between groups. DAPC is explained in depth in the DAPC chapter on Part II of this tutorial and in the DAPC adegenet vignette.

DAPC requires a genlight object with populations defined a priori. We already have this genlight object from the

above steps. Usually, we use the number of principal components and discriminant axes that maximize the variance between populations; but our objective here is to calculate the population assignments based on the results of the PCA. We will use the same parameters as in the PCA to make the results comparable between both methods. These parameters (n.pca=10 and n.da=7) will be used to reconstruct the DAPC, obtain the assignment of the samples to each population, and suggest admixture on a geographical basis. By making n.pca and n.da equal NULL, you can take an exploratory approach.

Now to confirm that the DAPC is like the PCA we can plot the data in a scatter plot.

```
yourdata.dapc <- dapc(gl. yourdata, n.pca = 5,
n.da = 3)

yourdata.dapc

scatter(yourdata.dapc, col = cols, cex = 2,
legend = TRUE, clabel = F, posi.leg =
"bottomleft", scree.pca = TRUE,

posi.pca = "topleft", cleg = 0.75)
```

In [7]:

Cross-Validation

We can help determine the optimal number of Principal Component axes to retain by using cross-validation. This function performs stratified cross-validation of DAPC using a varying numbers of PCs.

```
xval.yourdata <- xvalDapc(tab(gl. yourdata,
NA.method = "mean"), pop(gl. yourdata))

xval.yourdata
```

In [8]:

To visualize the results of the DAPC we can plot the data in a bar plot (this is basically a Distruct/STRUCTURE plot).

```
compoplot(yourdata.dapc,col = cols, posi = 'top')
```

In [9]:

We can organize our scatter plot based on locale too:

```
p1 <- ggplot(dapc.results, aes(x=Sample,
y=Posterior_membership_probability,
fill=Assigned_Pop))

p1 <- p1 + geom_bar(stat='identity')

p1 <- p1 + scale_fill_manual(values = cols)

p1 <- p1 + facet_grid(~Original_Pop, scales =
"free")
```

```
p1 <- p1 + theme(axis.text.x = element_text(angle
= 90, hjust = 1, size = 10))
p1
```

In [10]:

If it helps, you can view DAPC results in a table:

```
dapc.results <-
as.data.frame(yourdata.dapc$posterior)
dapc.results$pop <- pop(gl. yourdata)
dapc.results$indNames <- rownames(dapc.results)
dapc.results <- pivot_longer(dapc.results, -
c(pop, indNames))
head(dapc.results, n = 20)
colnames(dapc.results) <-
c("Original_Pop","Sample","Assigned_Pop","Posteri
or_membership_probability")
```

In [11]:

Pairwise genetic differentiation across populations (vcfR):

Now that you've got data/results for your population genetics, we can do calculate some statistics. Here we will calculate measures of genetic differentiation across

all population pairs. You define your populations to use first as a factor, and then make an object that measured pairwise genetic differentiation across those populations we just defined. We will specify 'nei' for our method, which stands for 'Nei's standard genetic distance'. This distance has the nice property that if the rate of genetic change (amino acid substitution) is constant per year or generation then Nei's standard genetic distance (D) increases in proportion to divergence time. This measure assumes that genetic differences are caused by mutation and genetic drift We can then make a table of the results:

```
pops <- as.factor(c("Coco","Motagua","Aguan",
"Izabal", "Salado", "Prinzapolka", "Tortuguero",
"Wawa", "Danto", "Guiamoreto", "Cacao"))

yourdata.diff <-
pairwise_genetic_diff(yourdata.vcf, pops, method
= 'nei')

colMeans(yourdata.diff[,c(4:ncol(yourdata.diff))]
, na.rm = TRUE)

knitr::kable(head(yourdata.diff[,1:15]))

knitr::kable(head(yourdata.diff[,16:19]))
```

```
knitr::kable(round(colMeans(yourdata.diff[,c(3:9,
16,19)], na.rm = TRUE), digits = 3))
```

In [12]:

Let’s do another, but instead of 'nei', we will specify 'jost' for our method. This stands for Jost's D, another way of measuring genetic differentiation:

```
yourdata.diff1 <-
pairwise_genetic_diff(yourdata.vcf, pops, method
= 'jost')

colMeans(yourdata.diff1[,c(4:ncol(yourdata.diff1)
)], na.rm = TRUE)

knitr::kable(head(yourdata.diff1[,1:15]))

knitr::kable(head(yourdata.diff1[,16:19]))

knitr::kable(round(colMeans(yourdata.diff1[,c(3:9
,16,19)], na.rm = TRUE), digits = 3))
```

In [13]:

Other stats:

This next function is pretty cool. We can use basic.stats() from the hierfstat` package to estimate individual counts, allelic frequencies, observed heterozygosities and genetic diversities per locus and population. It also Estimates mean observed

heterozygosities, mean gene diversities within population Hs, Gene diversities overall Ht and corrected Htp, and Dst, Dstp. Finally, estimates Fst and Fstp as well as Fis following Nei (1987) per locus and overall loci.

```
strata(gind.yourdata) <- yourdatapop.data

setPop(gind.yourdata) <- ~Country

gind.yourdata

bs.yourdata <- basic.stats(gind.yourdata)

bs.yourdata
```

Haplotype network notebook script:

Below is a script used to generate information for analyzing genetic data for the purpose of haplotype network generation. The function of this script is to take genetic data (in `.fasta` format) and to group individuals for the purpose of haplotype network construction.

In [1]:

Make sure the packages being called by the `library()` functions have been installed. Use the `install.packages()` function to do this if you haven't already. Remember to set your working directory!!

This script is meant to be exploratory in nature with regards to analyzing your dataset. It is very quick, and when used with other programs (which I indicate later), you can get great figures to put in your Results section.

First, we are going to read the DNA sequence and assign it to an object. Since I am working on *G. nicaraguensis* here, my object will be called `gnic`. This input file is going to be in fasta format.

```
library(ape)
gnic <- read.dna("Gnic-CO1.fasta", format =
"fasta")
gnic
```

In [2]:

With this data file, we have 27 sequences that are all the same length (655bp).

Next, we are going to use the `pegas` package to convert the DNA sequence into a haplotype.

```
library(pegas)

gnic.hap <- haplotype(gnic)

gnic.hap
```

In [3]:

For this data set, it should be finished processing almost immediately. We can see that there are 17 haplotypes, with: 1 sample belonging to haplotype 1, 1 sample belonging to haplotype 2, 1 sample belonging to haplotype 3, 1 sample belonging to haplotype 4, 1 sample belonging to haplotype 5, 1 sample belonging to haplotype 6, 6 samples belonging to haplotype 7, 1 sample belonging to haplotype 8, 1 sample belonging to haplotype 9, 1 sample belonging to haplotype 10, 1 sample belonging to haplotype 11, 1 sample belonging to haplotype 12, 2 samples belonging to haplotype 13, 5 sample belonging to haplotype 14, 1 sample belonging to haplotype 15, 1 sample belonging to haplotype 16, and 1 sample belonging to haplotype 17.

You can go on to plot the network below, but I think the thing to do is to work with PopART (http://popart.otago.ac.nz/index.shtml) from this point on. I don't think RStudio makes good graphics for haplotype networks, but PopART does. With that said, there is a lot of good info you can get from this script, namely what you see above.

```
gnic.net <- haploNet(gnic.hap)

plot(gnic.net, size = attr(gnic.net, "freq"),

fast = FALSE)
```

Jupyter notebook:

Below is a script that was executed via Jupyter notebook. Each unique chunk of code is indicated numerically within brackets. The purpose of this script is to take the output file (in `.snps.hdf5` format) and use it in a population structure analysis. This can be used to coincide with other similar programs to check your work. Modified from the ipyrad website.

```
In [1]:
        #conda install ipyrad -c bioconda
        #conda install structure clumpp -c ipyrad
        #conda install toyplot -c eaton-lab
        import ipyrad.analysis as ipa
        import toyplot
        #Now indicate the path to your .snps.hdf5
        database file.
        data =
        "/home/FM/ktaube/201105_AHLVWJDSXY/trimmed/master
        s-p1-vmac1-trim_outfiles/masters-p1-vmac1-
        trim.snps.hdf5"
```

```
        # group individuals into populations. You MUST
        know your data. What samples belong to what
        populations is essential. It takes time to do,
        and even more time to do thoroughly.


        imap = {
            "crica": ["2G", "4H", "8H", "9G", "G10",
        "A12", "B12", "G12"],
            "guat": ["1E", "2F", "5A", "5F", "6D"],
            "hond": ["1A", "1B", "1C", "1D", "1F", "1H",
        "2A", "2B", "3F", "4D", "7G", "7H", "8B", "8F",
        "H10", "A11", "C11", "G11"],
            "nica": ["2D", "2E", "4A", "6G", "G12"],
        }
In [2]:
        #This will require that 50% of samples have data
        in each group
        minmap = {i: 0.5 for i in imap}
        #init analysis object with input data and
        (optional) parameter options. We will do a
        structure run here with different populations,
        specified by `kpop`, and will repeat it 3 times
        by using `nreps`.
        struct = ipa.structure(
            name="test",
            data=data,
            imap=imap,
            minmap=minmap,
            mincov=0.9,
        )


        struct.mainparams.burnin = 5000
        struct.mainparams.numreps = 10000
```

```
        struct.run(nreps=3, kpop=[2, 3, 4, 5, 6, 7],
        auto=True)
```

In [3]:

```
        #Make a table of [2]. The table will include
        nreps, lnPK, lnPPK, deltaK, estLnProbMean, and
        estLnProbStdev.

        etable = struct.get_evanno_table([2, 3, 4, 5, 6,
        7])

        etable
```

In [4]:

```
        #Now we retrieve a canvas object and set the size
        of our graph. We are going to put the table made
        in [3] into a graphical format.


        canvas = toyplot.Canvas(width=400, height=300)


        #We plot the mean log probability of the models
        in red. When you view the graph, look for where
        the red line is the lowest. This will give you a
        good idea of how many populations (indicated in
        literature as K) you may have.

        axes = canvas.cartesian(ylabel="estLnProbMean")

        axes.plot(etable.estLnProbMean * -1,
        color="darkred", marker="o")


        axes.y.spine.style = {"stroke": "darkred"}

        #We plot delta K with its own scale bar of left
        side and in blue.


        axes = axes.share("x", ylabel="deltaK",
        ymax=etable.deltaK.max() + etable.deltaK.max() *
        .25)

        axes.plot(etable.deltaK, color="steelblue",
        marker="o");

        axes.y.spine.style = {"stroke": "steelblue"}
```

```
        #Now we set X labels.
        axes.x.ticks.locator =
        toyplot.locator.Explicit(range(len(etable.index))
        , etable.index)
        axes.x.label.text = "K (N ancestral populations)"
```

In [5]:

```
        #From here we start the process of analyzing
        results. We will go from [5] to [7] to do this.
        k = 3
        table = struct.get_clumpp_table(k)
```

In [6]:

```
        #We sort our list by columns, as we would when
        making a population structure plot.
        table.sort_values(by=list(range(k)),
        inplace=True)
        #OR, we can sort our graph by a list of names,
        here taken from imap which we set up earlier.
        import itertools
        onames = list(itertools.chain(*imap.values()))
        table = table.loc[onames]
```

In [7]:

```
        # Finally, we can build our barplot! This figure
        probably won't be put into a journal, BUT it is
        an awesome tool to use with other scripts to get
        good results.
        canvas = toyplot.Canvas(width=500, height=250)
        axes = canvas.cartesian(bounds=("10%", "90%",
        "10%", "45%"))
        axes.bars(table)
        #The last thing is to add labels to the x-axis.
```

```
ticklabels = [i for i in table.index.tolist()]
axes.x.ticks.locator =
toyplot.locator.Explicit(labels=ticklabels)
axes.x.ticks.labels.angle = -60
axes.x.ticks.show = True
axes.x.ticks.labels.offset = 10
axes.x.ticks.labels.style = {"font-size": "12px"}
```

Supplementary lab protocols

Below are five laboratory protocols used throughout this thesis, with each protocol step indicated with a bullet point in chronological order. These protocols were used extensively throughout the time spent working towards completion of this thesis, and are included below to 1) acknowledge their repeated importance throughout this project, and 2) to assist in aiding future students that may seek guidance on how to conduct these particular methods effectively. They are written with brevity in mind, so that they can be easily referred to when working at a lab bench or in writing.

Protocol 1: Qubit Broad Range DNA Analysis

- Add 200µL AccuGreen Broad Range dsDNA Quantitation Solution (master mix) into a tube (solution is light sensitive, must be protected) for each sample +1 (if testing 10 tubes, have enough for 11)
- For the samples, add 198µL master mix to each assay tube, followed by 2µL of the DNA sample
- Vortex all assay tubes (including standards). Let all samples sit in the dark for 2 minutes
- To use Qubit, touch the touch screen to wake up the machine
- From the home screen, pick your assay (DNA), and then the sensitivity (Broad Range)

- Select “Yes” when asked if you want to read new standards. Lift black lid, and place Standard 1 into the chamber, close the lid, and push “Read”. When finished, repeat with Standard 2
- Read samples. After first sample is assayed, push the “Calculate Stock Conc.” button, then use the digital dial to indicate the volume of sample you put into your assays (2µL). This will give you the concentration of your sample in whatever units you like
- To save data to a flash drive, after reading all your samples, push the “Data” button on the bottom of the screen. Then push the flash drive button on the left side of the screen (has green light in the corner). Once it is saved, the flash drive may be removed from the machine. Files are saved in .csv format, which can be opened in Excel

Protocol 2: Qubit High Sensitivity DNA Analysis

- Add 199µL AccuGreen Buffer, 1X into a tube (solution is light sensitive, must be protected) for each sample +1 (if testing 10 tubes, have enough for 11)
- Add 1µL AccuGreen Dye, 200x into the same tube. Vortex the tube. This is the master mix
- To make standards, add 190µL master mix with 10 µL of AccuGreen Standard 1 (0ng/µL) into an assay tube and AccuGreen Standard 2 (10ng/µL) into an assay tube

- For the samples, add 198µL master mix to each assay tube, followed by 2µL of the DNA sample
- Vortex all assay tubes (including standards). Let all samples sit in the dark for 2 minutes.
- To use Qubit, touch the touch screen to wake up the machine
- From the home screen, pick your assay (DNA), and then the sensitivity (Broad Range)
- Select “Yes” when asked if you want to read new standards. Lift black lid, and place Standard 1 into the chamber, close the lid, and push “Read”. When finished, repeat with Standard 2
- Read samples. After the first sample is assayed, push the “Calculate Stock Conc.” button, then use the digital dial to indicate the volume of sample you put into your assays (2µL). This will give you the concentration of your sample in whatever units you like
- To save data to a flash drive, after reading all your samples, push the “Data” button on the bottom of the screen. Then push the flash drive button on the left side of the screen (has green light in the corner). Once it is saved, the flash drive may be removed from the machine. Files are saved in .csv format, which can be opened in Excel

Protocol 3: Qiagen DNeasy Tissue Extraction Protocol

- Cut tissue and place in a 1.5mL microcentrifuge tube.
- Add 180µL Buffer ATL
- Add 20µL proteinase K
- Mix by vortexing and incubate at 56 degrees C until completely lysed (2 hours-overnight). Vortex completely before moving to step 5.
- Add 200µL Buffer AL. Mix thoroughly by vortexing.
- Add 200µL EtOH (96-100%). Mix thoroughly by vortexing.
- Pipet the mixture into a DNeasy Mini spin column placed in a 2mL collection tube. Centrifuge at 8000rpm for 1 minute. Discard flow-through and collection tube.
- Place the spin column in a new 2mL collection tube. Add 500µL Buffer AW1. Centrifuge for 1 minute at 8000rpm. Discard flow-through and collection tube.
- Place the spin column in a new 2mL collection tube. Add 500µL Buffer AW2. Centrifuge for 3 minutes at 14,000rpm. Discard flow-through and collection tube.
- Transfer the spin column to a new 1.5 or 2mL microcentrifuge tube.
- Elute the DNA by adding 200µL Buffer AE to the center of the spin column membrane. Incubate for 1 minute at room temperature. Centrifuge for 1 minute at 8000rpm. Can also use 100µL Buffer AE and spin into two separately labeled microcentrifuge tubes.

Protocol 4: Cycle Sequencing Initial Purification Protocol (Step 1 of 2)

In each reaction tube, the following reagents and volumes should be prepared:

Terminator Ready Reaction mix (Big Dyes)	1 µl
BigDye Seq Buffer (Dilution Buffer)	3 µl
Primer	0.5 µl
Template	2 µl
Water	3.5 µl
Total	10 µl

It is important to consider first how much total volume will be needed

- Add dilution buffer so total volume of dilution buffer and Big Dyes is 4 µl. 0.5 µl of 10mM primer is usually more than enough. Make master mix, and use electronic pipette to aliquot into 96-well plate (plates are easier to manipulate during the cleaning steps). Remember to run samples two times -- once with the forward primer and once with the reverse primer.
- Initial denaturation of 96° C for 1 min (25 cycles):

  96° C for 10 sec

  50° C for 5 sec

  60° C for 4 min

- Refrigerate at 4° C until ready to continue.

Protocol 5: Cycle Sequencing 3730 Protocol (Step 2 of 2)

- Make a master mix of EtOH/EDTA solution. If your final sequencing volume was 10 µl, then for each sample, add:
- 2.5 µl 125mM EDTA
- 30 µl 100% EtOH
- Add 32.5 µl of EtOH/EDTA solution to 10µl of cycle sequencing product
- Seal tubes and invert a few times to mix
- Leave at room temperature up to 15 minutes to precipitate extension products
- Spin in refrigerated centrifuge 2500g for 30 minutes at 4oC (program #1 on Eppendorf refrigerated centrifuges). Be sure to balance racks and tubes (Same number of tubes in each rack (balancing tubes can be empty))
- Remove seal and invert tray onto paper towel. Secure 2-3 paper towels over tubes with rubber bands.
- Place tray inverted into centrifuge and spin 50 g (up to 185 g) for 3 minutes

- Add 30µl 70% EtOH to each pellet. ABI recommends making a fresh stock of 70% each time you do this. For each sample, add:

  21 µl non-denatured 100% ethanol

  9 µl Water
- Seal tubes and invert a few times to mix
- Spin plates 2000-3000 g for 15 minutes at 4oC
- Repeat steps 6 and 7 to remove 70% EtOH
- Samples are ready to be resuspended for 3730 run. They can also be covered in aluminum foil and stored at 4° C
- To run on 3730, add 10µl Hi-Di formamide to each tube.

Supplementary thesis files:

Below are a series of files meant to describe the parameters of ipyrad, the steps of ipyrad, and accessing Jupyter Notebooks via cluster. Additionally, several results files have been added to this section, which, while informative, would not typically be seen in a published paper. It is *essential* that a working understanding of ipyrad is necessary to both operate the program and troubleshoot, as is running scripts via cluster using Jupyter Notebooks.

The parameters of ipyrad:

Each of the steps below are listed in order, and contains the step number, the step title, and beneath it a brief description of what is happening in the program. The most important two steps are `clust_threshold` in steps 3 and 6 and `min_samples_locus` in step 7. They are the ones that are probably most misused or mis-specified that have the biggest impact on downstream analysis (ipyrad manual, 2019). Knowing what steps are used in the ipyrad program runs is essential for troubleshooting and understanding what is actually being done with the raw data. Therefore, the steps used in ipyrad have been added to this thesis for the purpose of giving readers a better understanding of the program, and how it contributes to the software pipeline. The following steps have been taken and edited from the ipyrad manual available online, version 0.9.55.

Step 0: `assembly_name`

The assembly name is used as the prefix for all output files. It should be a unique identifier for the assembly, meaning the set of parameters you are using for the current data set. When I assemble multiple data with different parameter combinations I usually either name them consecutively (e.g., data1, data2), or with names indicating their parameter combinations (e.g., data_clust90, data_clust85). The Assembly name cannot be changed after created with the `-n` flag, but a new Assembly with a different name can be created by branching the Assembly (see branching workflow).

Step 1: `project_dir`

A project directory can be used to group together multiple related assemblies. A good name for `project_dir` will generally be the name of the organism being studied. The `project_dir` path should generally not be changed after an analysis is initiated, unless the entire directory is moved to a different location/machine.

Step 2: `raw_fastq_path`

This is a path to the location of raw (non-demultiplexed) fastq data files. If your data are already demultiplexed then this should be left blank. The input files can be `gzip` compressed (i.e., have name-endings with `.gz`). If you enter a path for raw data files then you should also enter a path to a

barcodes file. To select multiple files, or all files in a directory, use a wildcard character (*).

Step 3: **`barcodes_path`**

This is the path to the location of the barcodes file and is used in Step 1 for demultiplexing, and can also be used in Step 2 to improve the detection of adapter/primer sequences that should be filtered out. If your data are already demultiplexed the barcodes path can be left blank.

Step 4: **`sorted_fastq_path`**

This is a path to the location of sorted fastq data. If your data are already demultiplexed then this is the location from which data will be loaded when you run Step 1. A wildcard character can be used to select multiple files in directory.

Step 5: **`assembly_method`**

There are four assembly methods options in ipyrad: `denovo`, `reference`, `denovo+reference`, and `denovo-reference`. The latter three all require a reference sequence file (parameter 6) in fasta format.

Step 6: `reference_sequence`

The reference sequence file should be in fasta format. It does not need to be a complete nuclear genome, but could also be any other type of data that you wish to map RAD data to; for example: plastome or transcriptome data.

Step 7: `datatype`

There are many forms of restriction-site associated DNA library preparation methods and thus many differently named data types. Currently, ipyrad categorizes these into six data types. This step can be ignored.

Step 8: `restriction_overhang`

The restriction overhang is used during demultiplexing (Step 1) and to detect and filter out adapters/primers (in Step 2), if the `filter_adapters` parameter is turned on. Identifying the correct sequence to enter for the `restriction_overhang` can be tricky. You do *not* enter the restriction recognition sequence, but rather the portion of this sequence that is left attached to the sequenced read after digestion.

Step 9: `max_low_qual_bases`

During Step 2 bases are trimmed from the 3' end of reads when the quality score is consistently below 20 (which can be modified by modifying `phred_Qscore_offset`). However, your reads may still contain some number of ambiguous (N) sites that were not trimmed based on quality scores, and these will affect the efficiency and accuracy of clustering downstream. This parameter sets the upper limit on the number of Ns allowed in reads. The default value for `max_low_qual_bases` is 5. I would generally recommend against increasing this value greatly.

Step 10: `phred_Qscore_offset`

Bases are trimmed from the 3' end of reads if their quality scores is below this 20. The default offset for quality scores is 33. Some older data use a qscore offset of 64, but this is increasingly rare. You can toggle the offset number to change the threshold for trimming. For example, reducing the offset from 33 to 23 is equivalent to changing the minimum quality score from 20 to 10, which is approximately 95% probability of a correct base call.

Step 11: `mindepth_statistical`

This is the minimum depth at which statistical base calls will be made during step 5 consensus base calling. By default, this is set to 6, which for most

reasonable error rates estimates is approximately the minimum depth at which a heterozygous base call can be distinguished from a sequencing error.

Affected steps: 4 and 5.

Step 12: `mindepth_majrule`

This is the minimum depth at which majority rule base calls are made during Step 5 consensus base calling. By default, this is set to the same value as `mindepth_statistical`, such that only statistical base calls are made. This value must be <= `mindepth_statistical`. If lower, then sites with coverage >= `mindepth_majrule` and < `mindepth_statistical` will make majority rule calls. If your data set has low sequencing depth, then lowering `mindepth_majrule` can be an effective way to increase the amount of usable information in your data set. However, you should be aware the majority rule consensus base calls will underestimate heterozygosity.

Step 13: `maxdepth`

Sequencing coverage is often highly uneven among due to differences in the rate at which fragments are amplified during library preparation, the extent to which varies across different library prep methods. Moreover, repetitive regions of the genome may appear highly similar and thus cluster as high

depth clusters. Setting a `maxdepth` helps to remove the latter problem, but at the expense of potentially removing good clusters that simply were sequenced to high depth. The default `maxdepth` is set quite high (10,000), but you may change it as you see fit.

Step 14: `clust_threshold`

This the level of sequence similarity at which two sequences are identified as being homologous, and thus cluster together. The value should be entered as a decimal (e.g., 0.90). We do not recommend using values higher than 0.95, as homologous sequences may not cluster together at such high threshold due to the presence of Ns, indels, sequencing errors, or polymorphisms.

Step 15: `max_barcode_mismatch`

The maximum number of allowed mismatches between the barcodes in the barcodes file and those found in the sequenced reads. The default value is 0. Barcodes usually differ by a minimum of 2 bases, so it is not recommend using a value >2.

Step 16: `filter_adapters`

It is important to remove Illumina adapters from your data if present. Depending on the fidelity of the size selection procedure implemented during library preparation there is often at least some small proportion of sequences in which the read length is longer than the actual DNA fragment, such that the primer/adapter sequence ends up in the read. This occurs more commonly in double-digest (GBS, ddRAD) data sets that use a common cutter, and can be especially problematic for GBS data sets, in which short fragments are sequenced from either end. The `filter_adapters` parameter has three settings (0, 1, or 2). If 0, then reads are only removed if they contain more Ns than allowed by the `max_low_qual_bases` parameter. If 1, then reads are trimmed to the first base which has a Qscore < 20 (on either read for paired data), and removed if there are too many Ns. If 2, then reads are searched for the common Illumina adapter, plus the reverse complement of the second cut site (if present), plus the barcode (if present), and this part of the read is trimmed.

Step 17: `filter_min_trim_len`

During Step 2 if `filter_adapters` is > 0 reads may be trimmed to a shorter length if they are either low quality or contain Illumina adapter sequences. By default, ipyrad will keep trimmed reads down to a minimum length of 35bp. If you want to set a higher limit you can do so here.

Step 18: `max_alleles_consens`

This is the maximum number of unique alleles allowed in (individual) consensus reads after accounting for sequencing errors. The default value is 2, which is fitting for diploids. At this setting any locus which has a sample with more than 2 alleles detected will be excluded/filtered out. If set to `max_alleles_consens` = 1 (haploid) then error-rate and heterozygosity are estimated with H fixed to 0.0 in step 4, and base calls are made with the estimated error rate, and any consensus reads with more than 1 allele present are excluded. If `max_alleles_consens` is set > 2 then more alleles are allowed, however, heterozygous base calls are still made under the assumption of diploidy i.e., hetero allele frequency=50%.

Step 19: `max_Ns_consens`

The maximum fraction of uncalled bases allowed in consens seqs. If a base call cannot be made confidently (statistically) then it is called as ambiguous (N). You do not want to allow too many Ns in consensus reads or it will affect their ability to cluster with consensus reads from other Samples, and it may represent a poor alignment. Default is 0.05.

Step 20: `max_Hs_consens`

The maximum fraction of heterozygous bases allowed in consensus seqs. This filter helps to remove poor alignments which will tend to have an excess of Hs. The default value is 0.05.

Step 21: `min_samples_locus`

The minimum number of Samples that must have data at a given locus for it to be retained in the final data set. If you enter a number equal to the full number of samples in your data set, then it will return only loci that have data shared across all samples. If you enter a lower value (like 4) it will return a sparser matrix, including any loci for which at least four samples contain data. This parameter is overridden if a `min_samples` values are entered in the `popfile`. The default value is 4.

Step 22: `max_SNPs_locus`

Maximum number of SNPs allowed in a final locus. This can remove potential effects of poor alignments in repetitive regions in a final data set by excluding loci with more than N SNPs. Setting lower values is likely only helpful for extra filtering of very messy data sets. The default value is 0.2.

Step 23: `max_Indels_locus`

The maximum number of Indels (insertions or deletions of bases in the genome of an organism) allowed in a final locus. This helps to filter out poor final alignments, particularly for paired-end data. The default is 8.

Step 24: `max_shared_Hs_locus`

Maximum number (or proportion) of shared polymorphic sites in a locus. This option is used to detect potential paralogs, as a shared heterozygous site across many samples likely represents clustering of paralogs with a fixed difference rather than a true heterozygous site. The default is 0.5.

Step 25: `trim_reads`

Sometimes you can look at your fastq data files and see that there was a problem with the sequencing such that the cut site which should occur at the beginning of your reads is either offset by one or more bases, or contains many errors. You can trim off N bases from the beginning or end of R1 and R2 reads during step 2 by setting the number of bases here. This could similarly be used to trim all reads to a uniform length (though uniform read lengths are not required in ipyrad).

Step 26: `trim_loci`

Trim N bases from the edges of final aligned loci. This can be useful in denovo data sets, where the 3’ edge of reads is less well aligned than the 5’ edge, and thus error rates are sometimes higher at the ends of reads.

Step 27: `output_formats`

Here you decide what kinds of output datafiles you want. Disk space is cheap, so all formats can be made.

Step 28: `pop_assign_file`

Population assignment file for creating population output files, or assigning min_samples_locus value to each population. Enter a path to the file.

These 29 steps within the parameter file are what the ipyrad program follows in order to process SNP data. Once prepared, ipyrad processes the data with seven main steps and one preliminary step, each defined and described below. Sample code is provided where necessary below the step description for the purpose of aiding future readers working with this program.

The eight steps of ipyrad:

Below are the seven steps of ipyrad with a precursor step added (Step 0). Depending on the size of the input file, ipyrad can be completed in minutes, to over 24 hours to complete.

Step 0: accessing your data from the desktop/webserver

Always start an ipyrad assembly by using the `-n <filename>` argument to create a new named Assembly. Use the name relevant to your project. When you connect to a server make sure your .txt barcode files and `.gz/.fastq.gz` files are there (WinSCP is the program to use).

Code for step 0:

```
ipyrad -n masters-thesis
```

This will create a file in the current directory called params-masters-thesis.txt. The params file lists on each line one parameter followed by a ## mark, then the name of the parameter, and then a short description of its purpose.

Step 1: demultiplex the raw data files

Start assembling the data with ipyrad. Step 1 reads in the barcodes file and the raw data. It scans through the raw data and sorts each read based on the mapping of samples to barcodes. At the end of this step we'll have a new directory in our `project_dir` called `masters-thesis_fastqs/`. Inside this directory will be individual `fastq.gz` files for each sample.

Code for step 1:

```
ipyrad -p params-masters-thesis.txt -s 1 -r
```

There are 4 main parts to this step: (1) It creates a new Assembly called params-masters-thesis, since this is our first time running any steps for the named assembly; (2) It launches a number of parallel engines, by default this is the number of available CPUs on your machine; (3) It performs the step functions, in this case it sorts the data and writes the outputs; and (4) It saves the Assembly.

Another piece of information to look at here is the number of raw reads demultiplexed for each sample. Fortunately, ipyrad tracks the state of all your steps in your current assembly.

-s is the step(s). In ipyrad it can be a single number, or many at once up to 7.

-r tracks the state of all your steps in your current assembly, so at any time you can ask for results by invoking the -r flag. It basically tells the program to report (hence the `r`) the output of the step(s) you entered.

Step 2: filter reads

This step filters reads based on quality scores, and can be used to detect Illumina adapters in your reads, which is a common concern with any NGS data set, and especially so for homebrew type library preparations. Here the filter is set to the default value of 0 (zero), meaning it filters only based on quality scores of base calls,

and does not search for adapters. This is a good option if your data are already pre-filtered. The resulting filtered files from

Step 2 files are written to a new directory called `masters-thesis_edits/`.

Code:

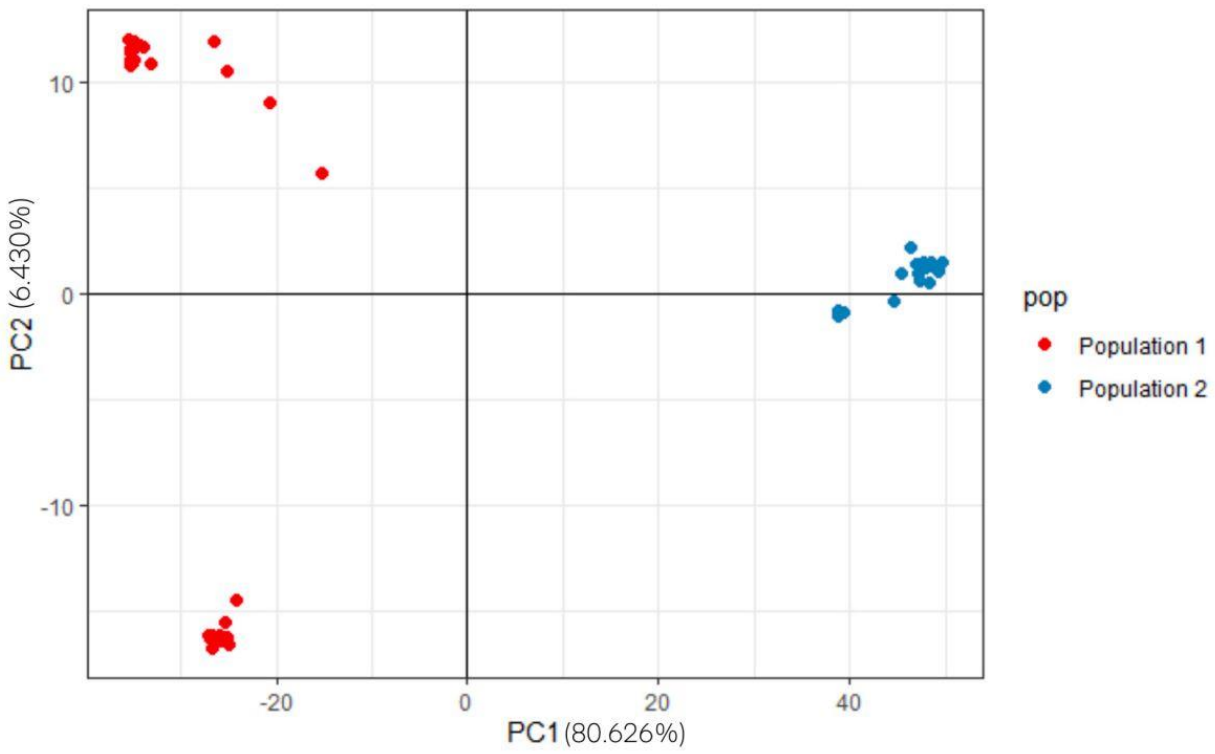
```
ipyrad -p params-masters-thesis.txt -s 2 -r
```

This result will show you the number of raw reads and the number of reads that passed quality filtering. For raw fastq paths with more than one lane (R1 and R2), use:

```
*.fastq
```

This should be at the end of your parameters. This will select all the fastq files. Can also be:

```
*.fastq.gz
```



Step 3: clustering within samples

Steps 3 and 6 are the “clustering” steps. These are by far the most intensive steps and on real data you should expect them to take quite a bit longer than the other steps.

Step 3 de-replicates and then clusters reads within each sample by the set clustering threshold and then writes the clusters to new files in a directory called masters-thesis_clust_0.85/. Intuitively we are trying to identify all the reads that map to the same locus within each sample. The clustering threshold specifies the minimum percentage of sequence similarity below which we will consider two reads to have come from different loci. The true name of this output directory will be dictated by the value you set for the clust_threshold parameter in the params file.

The aligned clusters found during this step are now located in ./masters-thesis_clust_0.85/. You can get a feel for what this looks like by examining a portion of one of the files using the command:

```
gunzip -c masters-thesis_clust_0.85/1A.clustS.gz
| head -n 28
```

Reads that are sufficiently similar (based on the above sequence similarity threshold) are grouped together in clusters separated by "//". It can be difficult to tell if this a homozygote with lots of sequencing errors, or a heterozygote with few reads for one of the alleles.

Step 4: joint estimation of heterozygosity and error rate

Step 4 jointly estimates sequencing error rate and heterozygosity to disentangle which reads are "real" and which are sequencing error. We need to know which reads are "real" because in diploid organisms there are a maximum of 2 alleles at any given locus. If we look at the raw data and there are 5 or ten different "alleles", and 2 of them are very high frequency, and the rest are singletons then this gives us evidence that the 2 high frequency alleles are good reads and the rest are probably not.

Code:

```
ipyrad -p params-masters-thesis.txt -s 4 -r
```

This step does not produce new output files, only a stats file with the estimated heterozygosity and error rate parameters.

Step 5: consensus base calls

Step 5 uses the inferred error rate and heterozygosity to call the consensus of sequences within each cluster. Here we are identifying what we believe to be the real haplotypes at each locus within each sample.

Code:

```
ipyrad -p params-masters-thesis.txt -s 5 -r
```

And here the important information is the number of `reads_consens`. This is the number of "good" reads within each sample that we'll send on to the next step. As you'll see in examples with empirical data, this is often a step where many reads are filtered out of the data set. If no reads were filtered, then the number of `reads_consens` should be equal to the number of `clusters_hidepth`.

This step creates a new directory called `./masters-thesis_consens` to store the consensus sequences for each sample. We can use the head command to look at the output.

Code:

```
gunzip -c masters-thesis_consens/1A_0.consens.gz
| head
```

You can see that all loci within each sample have been reduced to one consensus sequence. Heterozygous sites are represented by IUPAC ambiguity codes.

Step 6: cluster across samples

Step 6 clusters consensus sequences across samples. Now that we have good estimates for haplotypes within samples, we can try to identify similar sequences at each locus between samples. We use the same clustering threshold as Step 3 to identify sequences between samples that are probably sampled from the same locus, based on sequence similarity.

Code:

```
ipyrad -p params-masters-thesis.txt -s 6 -r
```

This step differs from previous steps in that we are no longer applying a function to each Sample individually, but instead we apply it to all Samples collectively. The end result is a map telling us which loci cluster together from which Samples. This output is stored as an HDF5 database (`masters-thesis_test.hdf5`), which is not

easily human readable. It contains the clustered sequence data, depth information, phased alleles, and other metadata. There is no simple way to summarize the outcome of step 6, so the output is uninteresting.

Step 7: filter and write output files

The final step is to filter the data and write output files in many convenient file formats. First, filters are applied for maximum number of indels per locus, max heterozygosity per locus, max number of SNPs per locus, and minimum number of samples per locus. All these filters are configurable in the params file and you are encouraged to explore different settings, but the defaults are quite good and quite conservative.

Code:

```
ipyrad -p params-masters-thesis.txt -s 7 -r
```

A new directory will be created called `masters-thesis_outfiles`. This directory contains all the output files specified in the params file. The default is to create all supported output files which include PHYLIP(.phy), NEXUS(.nex), EIGENSTRAT's genotype format(.geno), STRUCTURE(.str), as well as many others.

Accessing Jupyter Notebook via server:

Before using a Jupyter notebook through a server, you will obviously need to have Jupyter installed. This can be done using the following code:

```
conda install -c conda-forge jupyterlab
```

As stated earlier, it is essential to take the time necessary to understand what this software is, how it works, and to troubleshoot any issues that may present themselves.

Once installed, enter in the following code to activate the notebook:

```
ipython notebook --no-browser --port=8889
```

Then, open a new Ubuntu window without connecting to your server, and type

```
ssh -N -f -L localhost:8888:localhost:8889
youremail@phoebe.fm.pri
```

And then it will ask you for your password, which you then will enter. Next you will open a web browser (like Chrome or Firefox) and type in the following into the address bar:

```
localhost:8888
```

Which then will bring you to a login page where you will need to enter a password or token. To enter a token, use the value to the right of the `?token=` in the address bar as the token value, and then log in.