Περιεχόμενα

Σιμυλατινγ ανδ Πρεδιστινγ ανσιεντ ποπυλατιον δεμογραπηιςς υσινγ ὃμπυτατιοναλ ανδ Μασηινε Λεαρνινγ μετηοδς

Ιοαννις Πατραμανις

31 Οκτωβρίου 2019

0.1 Περίληψη

Το παρόν κείμενο αποτελεί την διπλωματική εργασία του Ιωάννη Πατραμάνη για το μεταπτυχιακό πρόγραμμα Πληροφορικής. Στην εργασία αυτή διερευνάται ,τόσο χρησιμοποιώντας υπολογιστικές μέθοδοι και μεθόδους μηχανικής μάθησης όσο και προσομοιώσεων, το φαινόμενο του οργανωμένου αποικισμού, γνωστή ως, "Δεύτερη φάση του Ελληνικού αποικισμού".

0.2 Summary

In this document lies the thesis of Ioannis Patramanis for the Bioinformatics Master program. In this work we used both computational methods and machine learning along with simulations in order to explore the phenomenon of the organised colonization known as "the second phase of Greek Colonization".

0.3 Acknowledgments

This thesis is the product of my 2 years in the Master of Bioinformatics. I would like to congratulate all of the people that worked and are working to make this master possible and allow students like me a free Master Diploma, in an age that being educated at the highest level and participating in research is still a privilege not enjoyed by everyone. Although there have been obvious difficulties in setting up this Master program, I believe that the people that finish have the necessary knowledge and experience and meet the international standards to call themselves bioinformaticians.

Having said that I would like to give my special thanks to the two people that supported this thesis by supervising it: Dr Dimitris Kafetzopoulos and Dr Pavlos Pavlidis. The two of them combined their strengths and granted my wish to work in and learn the field of human population genetics, an indeed rare opportunity. I wish of them to continue working together in the future and with more people, students and projects in this field. I believe our country has still a lot of results to yield, especially in the ancient genomes department and these two labs are primed to lead the way.

After about 7 years in Herakleion I would also like to thank all the people that have kept me company through the years and granted me probably some of the best memories of my life, even though most of them will never read this.

Finally a special thanks is reserved for Mr. Dimitris Patramanis and Mrs. Stavroula Kastriki, my parents as well as Eva Patramani, my sister for their endless support all those years here in this island.

Chapter 1

Introduction

In this document you will find a detailed description of my Master Thesis. This thesis was conducted under both Dimitris Kafetzopoulos, head of the Ancient DNA Lab of IMBB, FORTH and Pavlos Pavlidis, head of Evolab of ICS, FORTH. This work is inseparably connected to the "Apoikia" project of the aDNA lab. All actions taken have been with this project in mind. Having said that, the pipeline created here is of an exploratory manner and can be redesigned to answer a number of different questions. All code written during my master and this thesis can be found here and is free to use as it is, or as a template for your work.

Our overall aim is to create a pipeline for demographic inference from genome-wide shotgun sequencing data and to assess the predictability of certain demographic models. Our initial aim is to simulate genotypes under certain models, in an attempt to mimic a specific historical event. This event can include migrations and admixture between populations, lineage splits, population bottlenecks or exponential growths of population size. We then test our ability to predict the parameters of these simulations as well as to distinguishing between different models that produce similar results, using computational and machine learning methods.

The particular scenario in question is the creation of the ancient Corinthian colony of Ambracia. For reasons that will be discussed latter down the line, this scenario offers a unique opportunity to stress test some existing methods for demographic inference that are not so frequently used in human population genetics as well as some novel modification to them. Should our method provide descent results, it will pave the way for the use of these approaches in actual data.

Because of the multidisciplinary theme of this work, we will first have to explain some basic concepts from each field, so that no matter the background, anyone can understand the aim and purpose of this thesis: We will begin with an introduction to the filed of population genetics, focusing on human population studies. Following that is a brief overview of the data, tools and methods used in these fields, paying special attention to machine learning and one particular class of computational methods, center piece of this thesis: the Approximate Bayesian Computations. Finally there will be a summary of the historical and archaeological background regarding Ambracia and the Apoikia project before moving on to the main thesis.

Figure 1.1: "The Greek Colony, Marseille " oil painting by Pierre Puvis de Chavannes

1.1 A hundred-plus years of Population Genetics

1.1.1 Birth and early stages

Population genetics is a sub field of evolutionary biology, studying the genetic diversity and structure within and between populations and the processes governing them. [37, 38] After the publication of the Hardy–Weinberg principle in 1908, which can be regarded as the "year of birth" of this discipline, the seminal work performed by Ronald A. Fisher[28], John B. S. Haldane [43], Sewall G. Wright [75, 85] as well as and the important articles by Richard C. Lewontin [55] firmly established population genetics as an integral part of theoretical and evolutionary biology [33]. These founding personalities irreversibly rooted the field in mathematics and statistics, where its core still lies.

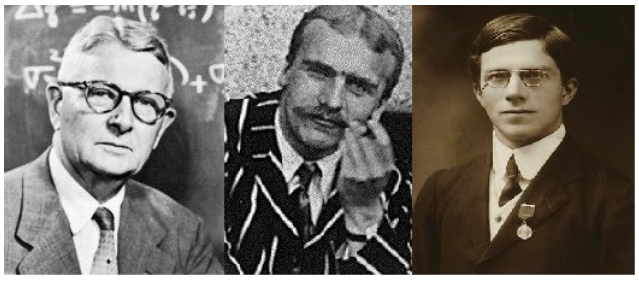


Figure 1.2: From left to right: Sewall G. Wright, John B. S. Haldane, Ronald A. Fisher

1.1.2 Selection vs Neutrality

In 1964 Motoo Kimura and James Crow suggested that most evolutionary changes at the molecular level, and most of the variation within and between species, are due to random genetic drift of mutant alleles that are selectively neutral [36]. This theory that was also suggested ,independently ,by two American biologists Jack Lester King and Thomas Hughes Jukes [44] and that would later be described in detail by Kimura in his 1983 monograph titled "The Neutral Theory of Molecular Evolution" [48] reignited an already existing debate of neutralist - selectionist (random genetic drift versus selection in evolution had also been vigorously debated in the 1940s) [9] and changed biology forever since. Today when searching to identify selection on a given locus ,one uses the neutral theory as a null hypothesis.

1.1.3 The Coalescent theory

Coalescent theory, a natural extension of the neutral theory was developed by multiple teams in the 1980's [62]. Coalescent is a model of how gene variants sampled from a population may have originated from a common ancestor. Given a set of parameters that effect it, two genes have a certain chance in each generation to coalesce into a common ancestral gene. In the simplest case, coalescent theory assumes no recombination, no natural selection, and no gene flow , no overlapping generations or population structure but has since been developed to include almost any scenario possible.

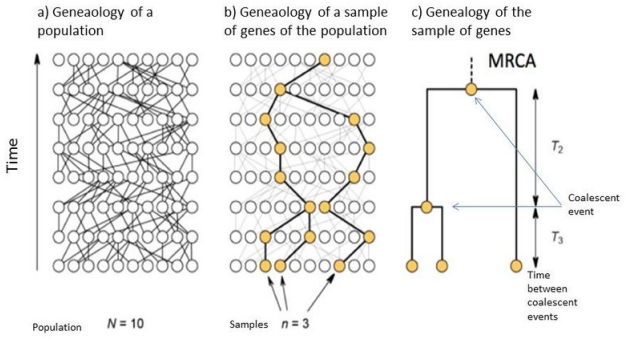


Figure 1.3: Visualisation of an example of coalescent theory. Three samples are selected from a population with a constant effective population size of 10, their Most Recent Common Ancestor lies 6 generations in the past.

1.1.4 The history and geography of human genes

When talking specifically about **human** population genetics it is impossible not to mention Luigi Luca Cavalli-Sforza. Even from the 1960's he pioneered statistical methods for estimating tree topologies [7] and wrote his concerns about representing human populations with such tree like structures, which he though inefficient when migration and admixture were heavily involved, as in the case of our species. He and his colleges where the first to apply Principal Component Analysis on genetic data [15] a method that is considered the baseline of a genetic analysis today. In the 1990's and early 2000's he among others pushed for the creation of database capturing the genetic diversity among humans, named the Human Genome Diversity Project [16], a database that

is still utilised today in many population genetics publications. He was one of earliest people to work with molecular data in a an anthropological framework, in conjunction with historical and linguistics data and by doing so, opened the way for the study of human history through the study of loci.

1.2 Population Data

1.2.1 Human Genetic Variation

All humans differ genetically. It has been shown that even in mono-zygotic twins, differences can be identified in their Copy Number Variation (CNV) Profiles [86]. Most of the genetic variation between humans are single base alterations. The 1000 genomes project sequenced and analyzed around 2500 individual whole genomes, from different populations around the world, and discovered that SNPs account for about 95 percent of the genetic variation and insertions and deletions around 4 percent. The remaining 1 percent can be attributed to what is regarded as 'structural differences' [81].

Single Nucleotide Polymorphisms (SNPs) are therefore the most reliable way to infer genetic affinity between two individuals, as they are present in every human, are inheritable and accumulate at a predictable rate in a population. In 2018 the dbSNP database announced moving from build 150 to 151 and from 324 to 660 million recorded SNPs, more than double. Build 150, in 2017 also contained almost double the number of SNPs from its previous version. Besides their applications in forensics [21] , genome wide associations studies [27], pharmacogenetics [87] and personalized medicine ([30], SNPs are the primary data for population genetics studies.

Both the existence or not of a SNP in a population, as well as its frequency in a population are valuable sources of information for the genetic history of that population. Two populations with a very recent common genetic origin will probably have very similar frequencies for most SNPs. Populations that are products of admixture tend to have SNP frequencies similar to their ancestral populations. Other parameters however, such as the effective population size and isolation, can also play a pivotal role on the genetic composition of a population. By analyzing thousands or even millions of SNPs at the same time, from different populations, it is possible to infer migration patterns and other past demographic events.

When using SNP data however it is of great importance to know the location and nature of the SNPs used in an analysis. A integral part of modern evolutionary biology is locating segments of DNA where selection, positive or negative, has taken place. When studying the genetic history of populations however one should avoid these loci and focus on neutral ones, as they can better represent a population's history. Genetically linked loci are also usually omitted in preference to ones that are independent of each other, as they can skewer some of the analyses. Finally it is important to distinguish autosomal SNPs from the ones present on the Y chromosome and the mitochondrion. The later ones, because of their unique nature regarding inheritance, recombination and mutation rate both require distinct handling when being analysed but also offer a different view on a population's history [49].

Figure 1.4: On average, genetic similarity between any two humans is 99.9 percent.There is about 2–3 times more genetic diversity within the wild chimpanzee population than in the entire human gene pool.

1.3 Modern Population Genetics

1.3.1 From Population Genetics to Population Genomics

In the last 10 years we have seen a dramatic decrease in the cost of sequencing and a parallel increase in the number of available genomes [64]. Even before the official completion of the 1000 genomes project [20], studies on worldwide human populations were being conducted using genome-wide data [56, 11]. These works led the transition of the field from population 'genetics' to population 'genomics': hundreds of samples and thousands of loci are now the new standard. Thanks to the decrease of the costs that we mentioned however, even a small to medium sized laboratory has the capability to acquire genome-wide data for multiple samples.

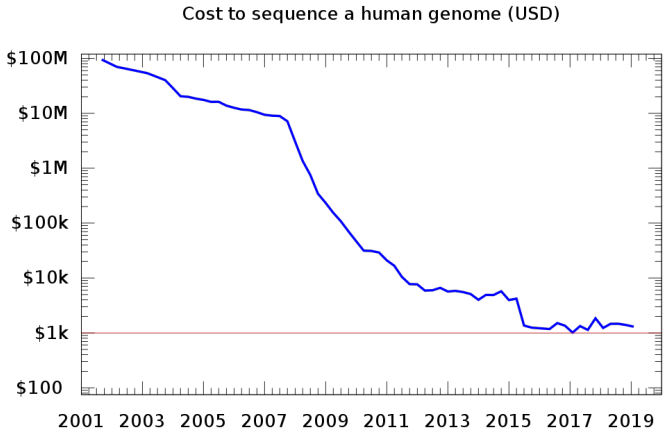


Figure 1.5: Cost of whole genome sequencing through the past years

1.3.2 SNP Arrays - Next Generation Sequencing

In the current day and age there are two main sources of SNP data being used by teams around the world for studying populations. Even with the overall decrease in the use of micro arrays in biology, SNP-panel micro arrays are still very popular in the field [78, 26, 74]. These are usually designed by companies using the preexisting knowledge of population diversity, contain thousands-even millions of SNPs at a relative low price per sample and are able to capture rough differences between contemporary groups of people. Just like in any other field of biology however, Next Generation Sequencing is quickly rising in usage. In population genetics the advantages are clear: more SNPs which offer a deeper and truer view on the structure and history of a group, as well as new, undetected SNPs which can lead to all sorts of discoveries[3]. This of course comes at a higher price per sample and many laboratories have to choose between higher sample size or higher depth for their samples.

1.3.3 Data Collection from temporary populations

One final note on modern population genetics, is on the changes in the sampling process. As data sets are getting larger, who gets to be sequenced is becoming a more meticulous selection process. Most studies today apply strict criteria when obtaining samples to represent a certain population: The individuals are usually required to have their grandparents all born in the same region of interests, to not have a known relation with any of the other samples of the study, be genetically healthy as well as any other cultural criterion the study demands (e.g. speaking a certain language).

1.4 Ancient Population Genetics - Archaic Humans

1.4.1 Ancient DNA

Ancient DNA refers to the DNA obtained from left over materials of organisms that lived hundreds or even thousands of years ago. These materials include bones, teeth, hair, feces as well a non-organic material that have into contact with an organism. The first attempts to extract and analyse aDNA were performed before the PCR era. In in 1984, [39] managed to recover DNA using bacterial cloning from dried muscle of quagga, an extinct subspecies of plains zebra (*Equus quagga*). However, due to extremely poor DNA preservation, analyses of aDNA were limited until an effective technology for DNA amplification, like PCR made very small amounts of DNA accessible for study [42]. High-throughput sequencing has however allowed the sequencing of larger ancient DNA fragments from across the entire length of a genome, not only increasing the available material to work with, but allowing a two-fold approach to authentication through analysis of postmortem damage and detection of secondary contaminating individuals

[77]. Some of the greatest achievements of the field include: the capturing of prior human genetic diversity, including their microbiome and their pathogens, [54], sequenceing extinct organisms [84] , including ancient homini and revealing secrets about our own evolution as a species[31, 60, 73].

1.4.2 Archeogenomics

The word archeogenomics is a combination of the words archeo meaning ancient and genomics the field studying genomes. It is commonly used to describe the emerging field of studying ancient genomes. Today hundreds of ancient human genomes from multiple locations and time points have been sequenced and analysed. By sequencing DNA from human remains such as bones and teeth we are able to reconstruct the populations of the past and infer our history, assisted always by other fields, such as history, anthropology, archaeology and linguistics. Recently using ancient DNA we have been able to reconstruct the genetic history of Europe by identifying 3 very diverse ancestral populations of modern Europeans[51], discovered unknown migrations to the Pacific Islands [71] and recorded the peopling of the Americas [63].

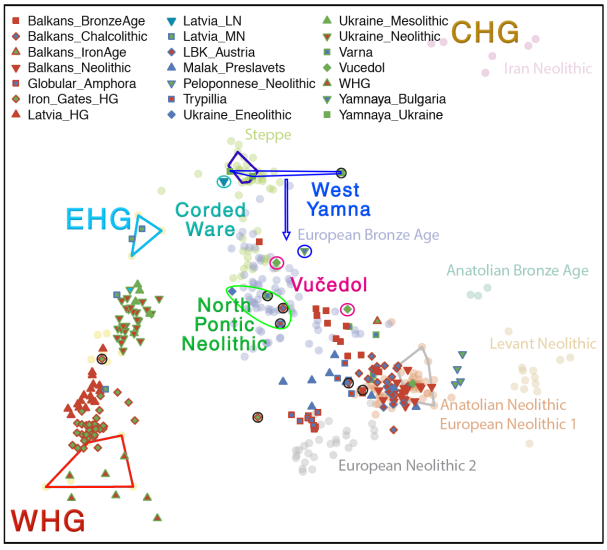


Figure 1.6: Principal Component Analysis of Ancient Samples from Europe from different time points. Notice the ancient populations forming a triangle around the modern Europeans : Hunter gatherer (HG) groups, Neolithic Farmers, Steppe Pastoralists

1.4.3 Capturing ancient polymorphism

In addition to brute force shotgun sequencing, new methods of capturing ancient genomic diversity in humans have been made popular. Targeted enrichment of the regions required to address a particular biological question is a method commonly used, which ranges from a limited number of loci, to several millions genome-wide markers [32, 29] and even the entire genomes [14]. When describing a new population, a common method is to enrich for several hundreds of thousands of loci that are known to diverge in frequencies between populations. The logic is similar to one we described above for modern genomes: either a more expensive and rather stochastic approach, but with the potential for discovery, using shotgun sequencing or a cheaper, guided and more conserved approach using enrichment of known polymorphic sites. One major drawback of ancient population genetics is however the lack of confidence, corresponding to the low quality of data. Most methods can overcome missing or erroneous data often by increasing the quantity of SNPs. However some of the most modern tools that require phased, 'haplotypic' data [23] are used very infrequently [59] as the quality of aDNA is considered too low for these kinds of analyses.

1.5 Tools and Metrics in Population Genetics data

In order to understand and visualize the differences and similarities between different population groups, a number of methods have been developed over the years. These range from simple distance metrics to complex haplotipic algorithms. As we have mentioned, most data on populations is comprised of Single Nucleotide Polymorphisms, thus the majority of tools and metrics of the field are centered around the analysis of SNPs.

1.5.1 Fst Distances and Visualization

One of the oldest metrics to infer genetic relations between populations are the Fixation Indices also known as F-statist[**?**]. Among these statistics Fst is the most commonly used. Fst, also known as Population Inbreeding Coefficient was developed as a metric to infer population substructure by measuring the distance between two subgroups within the original population. It is a fairly simple metric that measures population differentiation due to genetic structure and thus is commonly used to calculate how similar two populations are: the lower the Fst between them, the more closely related they are. The formula of Fst for 2 populations is given bellow.

$$Fst = \frac{\pi between - \pi within}{\pi between}$$

Where pi is the average number of pairwise differences between two individuals.

Very similar to the Fst are: Fis, which calculates the inbreeding coefficient for an individual and Fit which calculates an individuals inbreeding coefficient relative to the population.

1.5.2 Inbreeding Estimation

One of the most interesting measurements for a populations is its inbreeding levels. These can be calculated by a number of different metrics such as the Fst between random sub-sampled populations within the original, by measuring the Runs of Homozygosity per individual(total length of homozygous tracks on the genome) or by the number of Identical by Descent haplotype segments shared with other individuals. In the past few years, many studies have been made on these measurements as well as other signatures of inbreeding [17, 69, 4]

1.5.3 Haplotype Sharing Methods

When using the term "haplotype" in human population genetics one can be referencing one of two things: either the non-recombining and non-Mendelian inherited segments of DNA such as mitochondrial DNA and the Y chromosome or a group of mutations that because of their close genetic distance are inherited together. When inspecting two closely related individuals one can find many haplotypes of the second type that are identical in their polymorphic pattern. These are what we call Identical by Descent segments (IBDs). These segments help us identify admixture events that happened close to the present (in evolutionary scale) and their information doesn't always agree with the whole genome frequencies information. One example of this is in [26] where Crete shows very low affinity to North East populations when looking at allele frequency data and high affinity to them when looking at IBD sharing data.

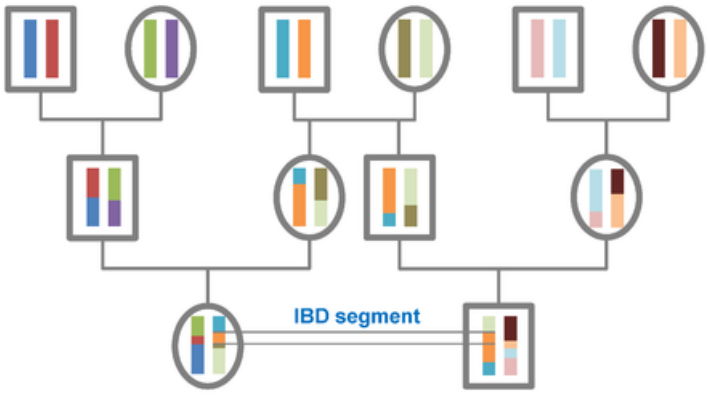


Figure 1.7: Visual representation of Identical by descent segments

1.5.4 Principal Component Analysis

Principal Component Analysis in a procedure aiming to transform a set of observed variables into another set of uncorrelated variables called principal components. Since the first component describes the maximum variance and each succeeding component a reduced amount from its previous one, PCA is a simple and effective way to perform dimensionality reduction. By plotting the

initial components one can describe the overall structure of the data. Originally invented by Karl Pearson in 1901 [68], it has become extremely popular in the recent years for data exploration. As we mentioned before, it was first utilised by Cavalli-Sforza and Edwards [**?**] by using each population as a sample and their frequencies for certain SNPs as features. In modern analyses individuals and not populations are being used as samples and the presence or not of the alternative SNP (heterozygous or homozygous) as features.

1.5.5 F-statistics

The term F-statistics can be used for two different but related groups of measurements. The traditional F-statistics which are based on the Fixation Index we described before and the more recent ones, also known as F-statistics described first in [**?**]. This second batch of F-statistics are related to the first one, but attempt to measure scenarios of admixture as well as relations between groups of individuals. One of the most commonly used of these statistics is "F3" which, using 3 groups and their allele frequencies, models one of the three as an admixture of the other 2. F3 uses an "F2" measurement which is very similar to the Fst. The formula for both F2 and F3 are given bellow. In order to test whether population three is a product of admixture between population one and population two, first we measure the F2 values between all pairs.

$$F2(p1, p2) = \pi between - \frac{\pi within p1 + \pi within p2}{2}$$

Where pi is the average number of pairwise differences between two individuals and p1 is short for population one. Now using these F2 values we can calculate the F3 for test population three using population one and two as "parental" populations.

$$F3(p1, p2, p3) = \frac{F2(p3, p2) + F2(p3, p1) - F2(p1, p2)}{2}$$

When F3 is a negative value there is evidence of admixture since, population three has a close relation to both population one and tw (low F2 values) and population one and two have a distant relation (high F2 between them).

In addition to F3, F4 as well as a D-statistic belong to the same group of metrics and are commonly used. However these require 4 populations to be utilised. These F statistics are very common in ancient DNA where haplotipic data is very rare and genotypes are commonly missing, since they can be used with frequency data

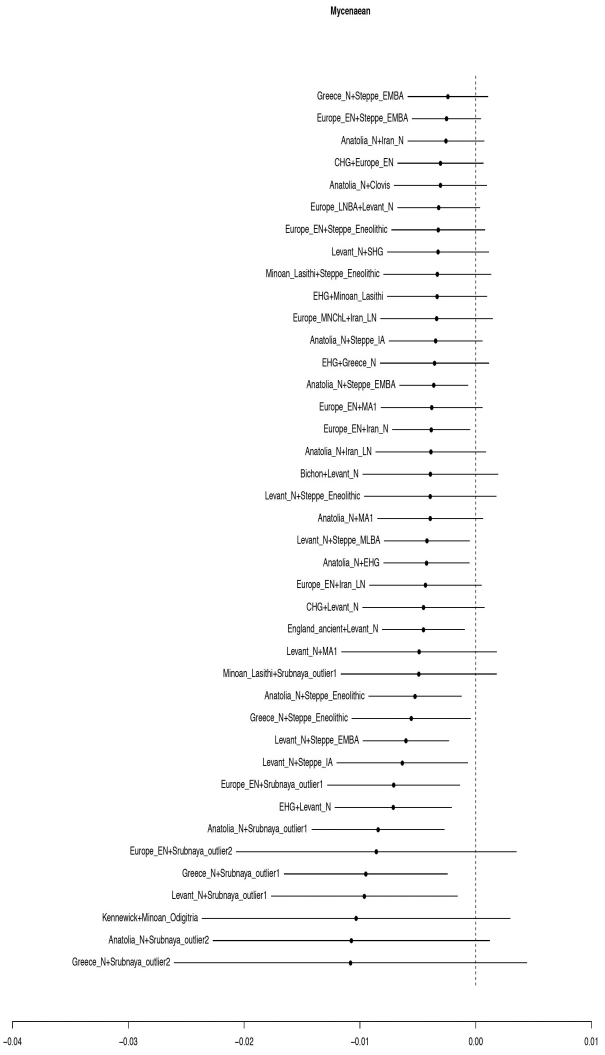


Figure 1.8: Example of the use of F3 from [52]. Here pairs of populations are used as models of parental population for the Mycenaean samples with Neolithic Greece and Srubnaya being the best fit.

1.6 Machine Learning and Computational tools on Population Genetics Data

Over the last decade Machine Learning (ML) has revolutionized our world. Entire fields have reinvented themselves from its use such as speech recognition, natural language processing, image recognition and bioinformatics [13, 58, 6]. In populations genetics however the field has lagged behind sticking mostly to traditional computational methods and only somewhat ML methods such as the PCA. Only recently have groups of the field experimented with these new set of tools: visualization using tSNE [50] on genome wide data [57], classification algorithms for natural selection [67] and even uMAP [53] on medieval ancient DNA genomes [59].

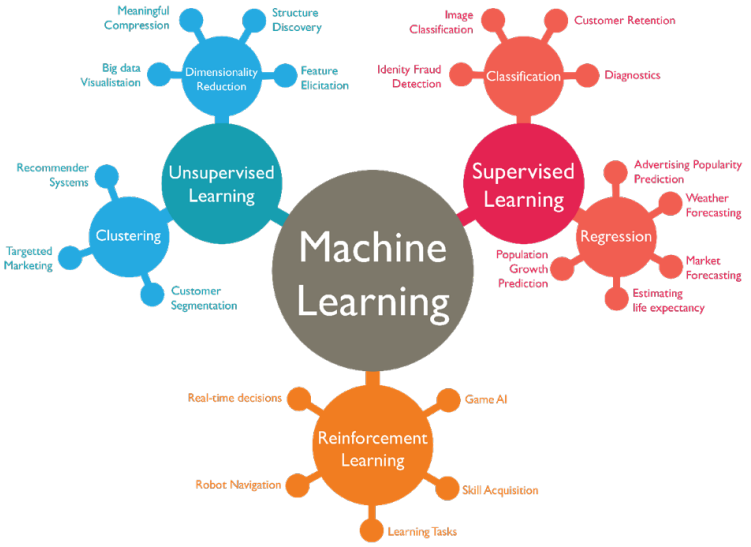


Figure 1.9: Visual overview of Machine Learning with examples

The basic concept of machine learning is to use a training data set to "learn" and a test data set to make predictions on (Classification,Regression) or to be transformed (PCA,tSNE,uMAP for visualization).

Machine Learning is usually divided into three groups: Supervised, Unsupervised and Semi-supervised learning. An example of the first group is a typical classification method, where two labeled classes of data are used for learning, class A and class B and a new data is then to be assigned to either one of the two classes.

An example of unsupervised learning would be a typical PCA where the data are inserted label-less and labeled afterwards when plotting. The algorithm has no knowledge to which class each sample belongs to when it runs, the only knowledge comes the data itself.

A third group, semi-supervised also known as reinforcement learning is comprised of methods where both labeled and unlabeled data are used in the training. One of most basic concept of reinforcement learning is that of trial and error as well as that of the reward. The algorithm attempts to map its way around the most ideal path using these two concepts.

1.6.1 A Note on ML tools

There are various machine learning algorithms available right now and each passing year a couple of brand new ones or some modified versions of an existing model are released. These range from basic decision trees to multilayered artificial neural networks, which evolved to what is now commonly known as deep learning. Depending on what task should be accomplished, what type and what amount of data is available a different ML method can prove itself the best for the job. There is no objectively best algorithm and when one is searching for the best tool for a particular task they ought to try as many as their time frame and workload allows.

Furthermore most ML tools run using some parameters. These parameter require what we call "tuning": running the same algorithm with different parameters and examining which combination outputs the best results.

Finally when talking about ML it is important for someone to understand the concept of "overfitting" and "underfitting". Lets say for example that we train an ML algorithm on some of our data. Paying attention to the notes above we chose the best software, model and parameters for our data and achieve very low error rates on our predictions. If we then use the same algorithm on different data and get much higher error rates, then what we have done is called overfitting: we have tailored our algorithm for our particular data which however are not the same with everyone's else and thus our method struggles with a different data set. The reverse problem is called underfitting: creating a model that performs the same for all data, but not in the optimal way.

Figure 1.10: Example of what overfitting and underfitting looks like

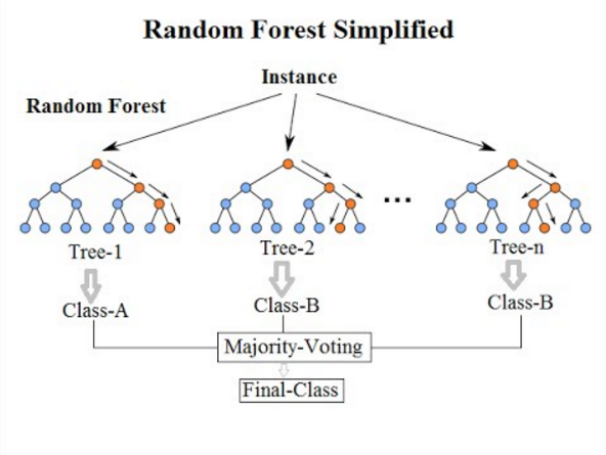


Figure 1.11: Example of Random Forests

1.6.2 Classifiers to be used

As we mentioned earlier there are hundreds of available ML models to use. For the purpose of this study we will use two classes of them: the Random Forests Classifiers and an SVM classifiers.

1.6.3 Random Forests

Random Forests (RF) are classifiers that work well for classification problems as they are able to exploit both high and low 'informative' features and deal with the problem of overfitting. The original classification algorithm which inspired RF was the Decision Trees method. Based on the values each of the features may take, 'decision' nodes are created resulting in a tree structure. Upon reaching a leaf of this tree, a decision is achieved for the label of the input data. The features with lower entropy (the most informative) appear closer to the root of the tree. However, a single tree might be heavily biased and as a result the algorithm may overfit. The solution to the overfitting problem is RF, a classifier that consists of several different decision trees whose outcomes are combined, usually by averaging the results, to predict the class of the input.

1.6.4 Support Vector Machines

SVM is a machine learning algorithm proposed by [22]. SVMs attempt to split the dataset into two classes via using a hyperplane that separates those classes. The goal is to find the ideal hyperplane which best separates those classes. It uses specific data points of each class to determine the position of the hyperplane. These points are called the Support Vectors. The distance between the hyperplane and the closest support vector from each class is calculated as 'the margin'. SVMs attempt to maximize this margin in order to maximize the probability of correctly classifying new data. Due to the ability of SVMs to reach higher dimensions, they do not suffer from the 'curse of dimensionality', making them a suitable algorithm for classifying samples with a multitude of features.

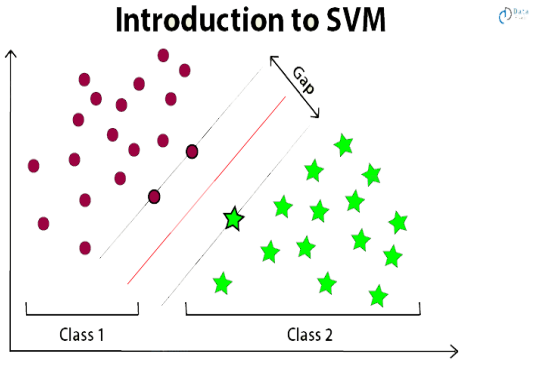


Figure 1.12: Example of a perfect SVM classification, notice the points-samples delimiting the gap between the classes - these are the support vectors

1.6.5 The Approximate Bayesian Computation

In 1984 Diggle Gratton [24] distinguished two forms of statistical models: those that are prescribed in terms of

known distributions, with known likelihood functions, and those that are implicit, from which we can simulate samples but do not have access to an explicit expression for the likelihood. Approximate Bayesian Computation is one of the most widely used method of the second group.

The basic outline of what subsequently became known as ABC (Approximate Bayesian Computation) was introduced by Pritchard [72] for solving an application in population genetics. Lets look at the following theoretical example:

Given one sample of unknown Z parameters, we have a number of observations. These observations can be translated into a Y number of summary statistics. Now we have a sample with Y dimensions, our Y summary statistics.

In order to infer its parameters we simulate K samples using all combinations of the parameters for a given range. Our aim is for the final data set of K simulations to have each parameter represented by a uniform distribution, this is known as the "prior distribution". For each of our K simulations we translate its observations and again obtain Y summary statistics. Now we can place all of our simulations and their corresponding statistics into a matrix of K rows and Y columns. This matrix is the most common input for ABC inference software.

Lets say that we only have one parameter in question to be inferred. For that parameter we can place our simulations in a Y+1 dimensional space, where each simulation would be represented by a dot in that space. There are Y dimensions corresponding to our summary statistics, plus one for the parameter in question. Remember that each simulation was created using a value for that parameter. In this space we can also place our sample which fits into that space, except for one dimension: the unknown parameter. At this step, it is a common technique in ABC to discard a large chunk of the simulations that are regarded as too distant from the sample. This is usually done using an euclidean distance measurement. The percentage of the discarded simulations is referred to as the "tolerance"of the ABC model.

In our example we discard 80 percent of our simulations and keep the 20 percent of the closest simulations.We are thus in a Y+1 dimensional space with K/5 dots (20 percent of the original K), the simulations that are left. Now using a single simulation and its position in the Y+1 dimensional space we can project our sample in that space as well, where it "should be" positioned, using linear regression thus making a prediction for its unknown parameter. If we do this for each of our K/5 simulations we end up with K/5 predictions and we can create a distribution of these predictions, which is known as the "posterior distribution".

It is common that each of these predictions is also scaled in "importance" using its distance from the sample - the further away the simulation the less important its regarded. It is up to the user to chose which metric of that posterior distribution of predictions, mean/mode/median is our final prediction of the unknown parameter. This is done for all of the parameters at the same time.

One of, if not the most important details about ABC is the selection of the summary statistics to be used. Theoretically any metric or measurement that is correlated with population structure can be used a summary statistic. In population genetics there are dozens of metrics that have been developed over the years. Obviously the better a metric reflects a scenario, the more useful it is to the ABC inference of that scenario. It is important to also note however that one should be careful when adding summary statistics, as previous works have shown that by increasing the number to high on can cause statistical noise [45] as well as decreased accuracy and stability to the ABC [80].

Finally some attention must be paid to the use of parameters that are not subject to inference and not randomized, but are used in the simulations. These parameters will be referenced as "Assumptions" for the purpose of this work, since they will be assumed to be known and of a certain value. These "known" parameters should be supported by previous experiments and real world data [79], since it has been shown to interfere with the conclusion [80].

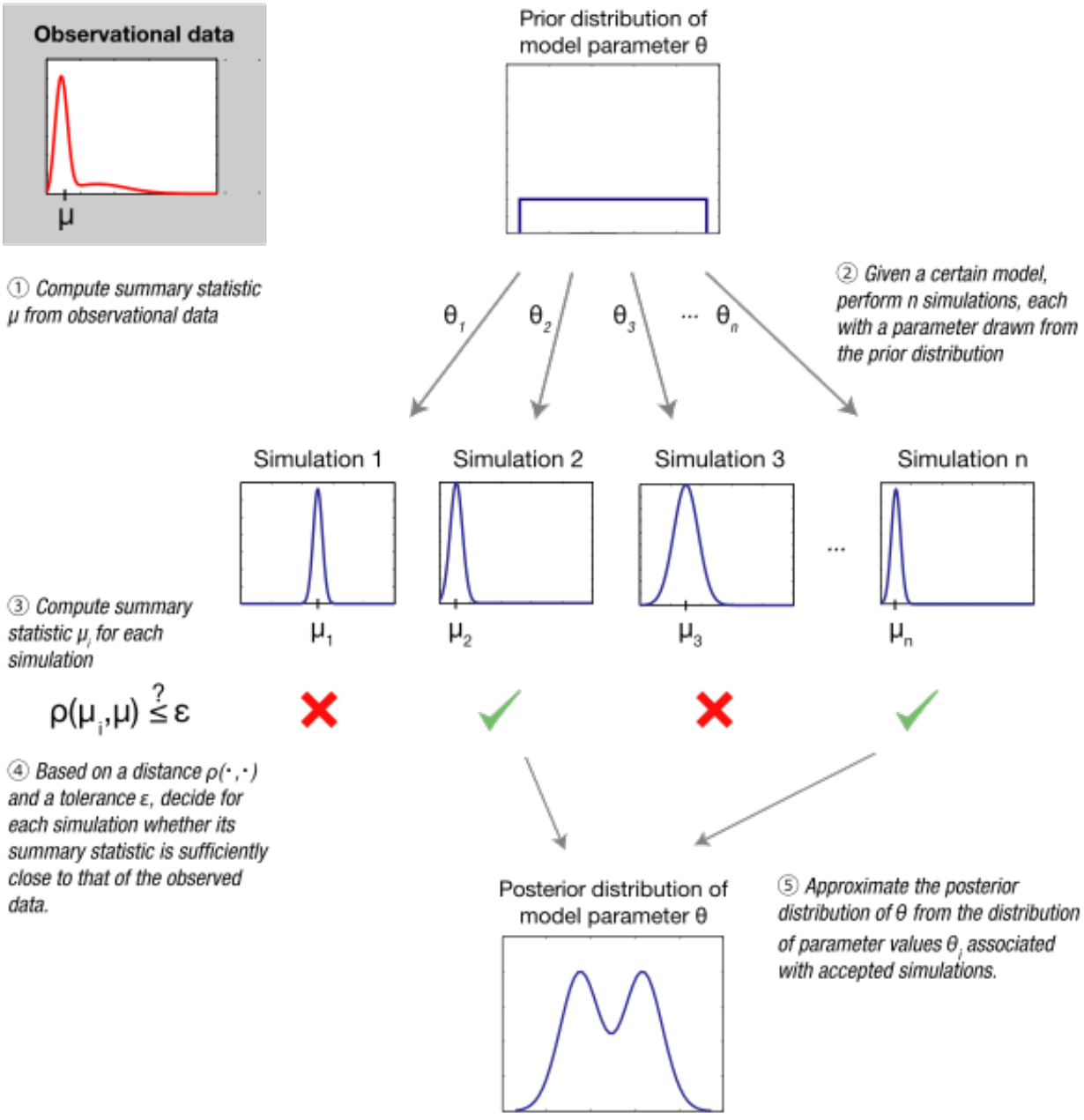


Figure 1.13: Visual representation of the ABC process

Chapter 2

Ambracia: a Corinthian Colony in Epirus

Figure 2.1: Uncovered amphitheater of Ambracia, modern day Arta, Greece

2.1 Historical Background of the models

As we have previously mentioned, our intent is to simulate a certain historical event, in order to weight the difficulty of estimating some of the factors involved in that scenario. Here we give a very brief overview on the background of that historical event: the Colonization of Epirus by the Corinthians and the creation of the city of Ambracia. We will first mention a few key factors that characterised the period that this event took place, known as the second phase of Greek colonization. Then we will take a look Epirus during that time and finally focus on the city of Ambracia. In the last section of this chapter we will briefly discuss the model that is based on this historical event along with the expected difficulties that come with studying it.

2.2 The second phase of ancient Greek colonization

The historical event we will be attempting to simulate belongs to a broader historical period, known as 'the second phase of Greek colonization'. During this period Greek city states organised the creation of new cities far from the southern Greek mainland all around the Mediterranean and the Black sea. These migrations were unique and differed from the preceding first phase of Greek colonization, which brought the Ioanians, Dorians and Aeolians on the shores of Aegean Islands and Asia Minor, which happened more 'naturally' due to population movements and demographic pressures during the Greek 'Dark Ages'. They also differed from the later settling of Greek speaking cities by Alexander the Great during his campaign, as these were a product of a monarchic kingdom while the colonies we are looking into, are products of free societies. Ultimately the creation of these colonies is fundamentally connected to the idea of the 'polis' , the independent city state [82].

In politics and history, a colony is a territory under the immediate political control of a state, distinct from the home territory of the sovereign. For colonies in antiquity, city-states would often found their own colonies. Some colonies were historically countries, while others were territories without definite statehood from their inception. The 'Metropolitan state' is the state that owns the colony. In Ancient Greece, the city that founded a colony was known as the 'metropolis'. "Mother country" is a reference to the metropolitan state from the point of view of citizens who live in its colony, the "apoikia".

Some of the earliest examples of this type of colony we will be looking into, can be found in southern Italy by the middle 8th century b.c. in an area that what would later be called Magna Grecia. This phenomenon would dominate the Greek world for the next couple of hundreds of years. By default these colonies would be organised using the template of and the follow the customs of their mother cities, their 'metropolis'. Populated by citizens moving from their metropolis or from other colonies, their citizens usually lost their political rights in their place of origin and sometimes were restricted from ever coming back [2]. The city state of Corinth is a proud example of this process. During the Archaic and Classical period Corinth had settled dozens of colonies, creating a network of cities along the western Balkan coast and southern Italy which fueled its trading capabilities.

Figure 2.2: The second phase of Greek colonization saw the spread of Greek settlements around the Mediterranean and the Black sea.

The process of the creating a colony itself, is not very clear to us. Its location was usually chosen by a number of factors such as geographical and geopolitical position as well as available land and valuable resources. It was believed that the colonising population was comprised of males only who would subsequently mate with local women, but today this is not considered the rule. Women, as well as entire families have been recorded to move to these colonies after the initial settlement. The size of the colony varied and could potential supersede that of the metropolis reaching numbers as high as 50.000 people [19].

2.3 Epirus during the Geometric, Archaic and Classical periods

Epirus has been occupied since at least Neolithic times by seafarers along the coast and by hunters and shepherds in the interior. A number of Mycenaean remains have been found in Epirus, especially at the most important ancient religious sites in the region, the Necromanteion (Oracle of the Dead) on the Acheron river, and the Oracle of Zeus at Dodona. [40, 1].

In the Middle Bronze Age, Epirus was inhabited by the same nomadic Hellenic tribes that went on to settle in the rest of Greece [1]. Aristotle considered the region around Dodona to have been part of 'Hellas' and the region where the Hellenes originated [34].

Figure 2.3: Rough map of ancient Epirus during the 4th century B.C.Depicted are its culture groups and important cities, based on archaeological or historical record

The Dorians are thought to have invaded Greece from Epirus and Macedonia at the end of the 2nd millennium BC (circa 1100–1000 BC), though the reasons for their migration are obscure. Epirus at that time is considered part of the Proto-Greek linguistic area during the Late Neolithic period. By the early 1st millennium BC, all fourteen 'Epirote' tribes including the Chaonians in northwestern Epirus, the Molossians in the centre and the Thesprotians in the south, were speakers of a strong 'North-west Greek dialect'. This dialect is closely related to Doric proper, while sometimes there is no distinction between Doric and the Northwest Greek. Whether it is to be considered a part of the Doric Group or the latter a part of it or the two considered subgroups of West Greek, the dialects and their grouping remain the same[25]. Unlike most other Greeks of this time, who lived in or around city-states, the inhabitants of Epirus lived in small villages and their way of life was foreign to that of the poleis of southern Greece[34, 1].

During the 7th and 6th century colonies were set up along the coastlines of Epirus by southern Greek states such as Corinth and Elis [35]. Beginning in 370 BC, the Molossian Aeacidae dynasty built a centralized state in Epirus and began expanding their power at the expense of rival tribes. In 295 BC and what is considered the peak of that kingdom, Pyrrhus came to throne. He is most famous for his costly wars against the Romans and Carthaginians, two rising superpowers of the era, but nonetheless brought great prosperity to Epirus, building the great theater of Dodona and a new suburb at Ambracia, a now prominent city, which he made his capital.

Figure 2.4: Coins from ancient Epirus

2.4 The case of Ambracia

Initially a local tribal Epirote settlement [83], later turned into a trading station and finally to a fully fledged city, ancient Ambracia was build on the banks of the river Arahthos and its ruins lie right bellow the modern day city of Arta. According to historical records it was founded by Gorgos, son of Cypselus, the tyrant of Corinth. After the expulsion of Gorgo's son, Periander, the citizens set up a democracy stationed between the local Epirote tribes of Athamanes, Kassopeans and Molossians. As a colony of Corinth it remained loyal to its mother city siding with her both in the Persian wars in which it contributed, according to records, around 500 hoplites and the in the Peloponnesian war with around 3000[46].

Figure 2.5: Recovered Mosaic from Ambracia

A prominent and flourishing city, it was regarded as a proud example of a democracy by Aristotle who in his work "Ambracian State" analyzes it's robust government. Its later years saw it first, becoming a semi independent city under Macedonian supervision during Philip's the second conquests and then being absorbed into the Epirote state, even becoming its capital under King Pyrrhus of Epirus. It was finally sacked by the hands of the Roman general Marcus Fulvius Nobilior at 189 B.C. and later its population forced to flee to the nearby city of Nicosia at 31 B.C.

The archaeological record shows a very organised city with main avenues of around 15 meters and smaller one of about 5, creating similar building blocks of around 20 houses. The houses are all equal, around the same size of 15x15 meters. Only later, around the Hellenistic period do we start to see true economic disparity with houses of expensive decorations showing up in the record. Its 2 necropolises, outside of the city walls, contributing around 300 human skeletal remains but also referring around 900 names written on tombs and columns as well as. These names include both prominent citizens such as generals and aristocrats but also cooks, artists and craftsmen[5].

Figure 2.6: Example of recovered column with written name, Ambracia

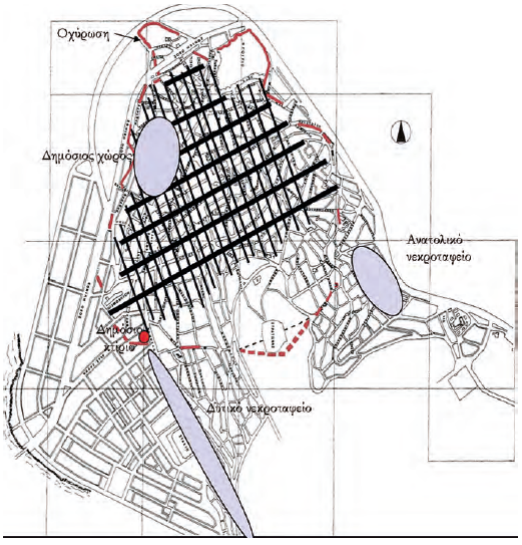


Figure 2.7: Sketch map of Ambracia, based on the archaeological findings. Depicted with light blue are the two necropolises (lower left and lower right), a public space (upper left), the walls of the city with a red line and a central public building with a red dot.

2.5 Project: Apoikia

Apoikia is a broad project centered around the second phase of Greek colonization which aims to explore that process through multiple lenses. One of the most exciting of those lenses is the archeogenomic one, with the city of Ambracia being a prime subject for it.

As we have seen there is both historical, archeological and even linguistic evidence that attempts to explain the formation of the city. Still, neither of those can truly reveal the genetic composition of its inhabitants. Were they all citizens of its mother city, Corinth or were they local people drawn in to the settlement. This wont be an easy task as a) we are looking at events at a very small time scale (about 800 years from the founding to its destruction) b) at populations that were probably related to each other.

On the other hand, the questions are many: Did the Corinthians drive out the people living there or did they incorporate them to their new society. What was the size of that population relative to the metropolis or the local one. What was the relation of the colony with its metropolis, did they exchange people or did the colonists distance themselves? How quickly was the colony increasing in size?

Our aim is discover which of these questions could be answered by sequencing the remains of individuals belonging to these three populations: The inhabitants of Ambracia, the inhabitants of a local, indigenous settlement as well as the inhabitants of Corinth.

For that purpose we will attempt to recreate the founding of the colony, as well as an extended period after that. This will be done by simulating three populations in time through a multitude of different scenarios. We will then attempt to correctly identify the different scenarios using only the genotypes produced by the simulations and metrics we can obtain from them, just as if we had randomly sequenced individuals from those populations.

Figure 2.8: View of Tomaros mountain from the theater in Dodoni, Epirus

Chapter 3

Materials and Methods

In this chapter we will discuss the processes and tools by which we intend to achieve our task. Although both ABC and ML methods have been used many times in the past and also applied to genotype data, many steps taken here were completely novel and experimental. Because of this, both a complete "rework" and smaller adjustments took place in the process whenever test results implied that was required. We will begin by describing the original design of the thesis step by step, as well as the problems that arose, when they did in the timeline of the thesis.

3.1 Overview

Our goal is to create a data set of simulations that includes and describes all the different possible interactions between two population groups and a third one being created by them and these interactions. We then intend to translate each simulation to a set of metrics, summary statistics that are also available to researchers of aDNA. Finally we will conduct internal training on the data sets either through ABC or ML models with the aim of predicting the parameters from which those simulations were generated by only using the summary statistics. The success of this method implies that it is possible for someone that uses these metrics on real sequencing data to infer the demographic parameters that lead to those genotypes. The first major step of the thesis was to construct the simulated data set as well as translate them in a rigorous and effective way into useful summary statistics.

In order to create and prepare the data to be used in the ABC, a small pipeline was set up. This pipeline is one of the centerpieces of this thesis, as it is the source of the simulations attempting to reproduce the scenario in question using all the different parameters, as well as the source of the summary statistics. The pipeline is designed to run multiple times and cover a wide range of values for each a parameter.

In each run of the pipeline, a set of values, each corresponding to a parameter, is selected at random. What these parameters are and how the effect the simulation is discussed later in this chapter. These parameters are stored and used by the simulator as a scenario from which genotypes are outputted. These genotypes are then funneled through our selected software to generate summary statistics. The summary statistics are formatted and stored. Finally for each run the set of parameters used to generate it and the summary statistics from that data set are paired and stored into a new file. This final file contains rows equal to the number of runs. Each row represents a run of the simulations as a vector with its first 14 dimension corresponding to the 14 parameters used in the simulations and the rest of the dimensions corresponding to the summary statistics generated by the simulation's genotypes.

In the next page you can find a visual representation of the pipeline described above, in its original form. Modifications where made during the thesis in order to handle the problems encountered. Each part of the pipeline is described in the following pages.

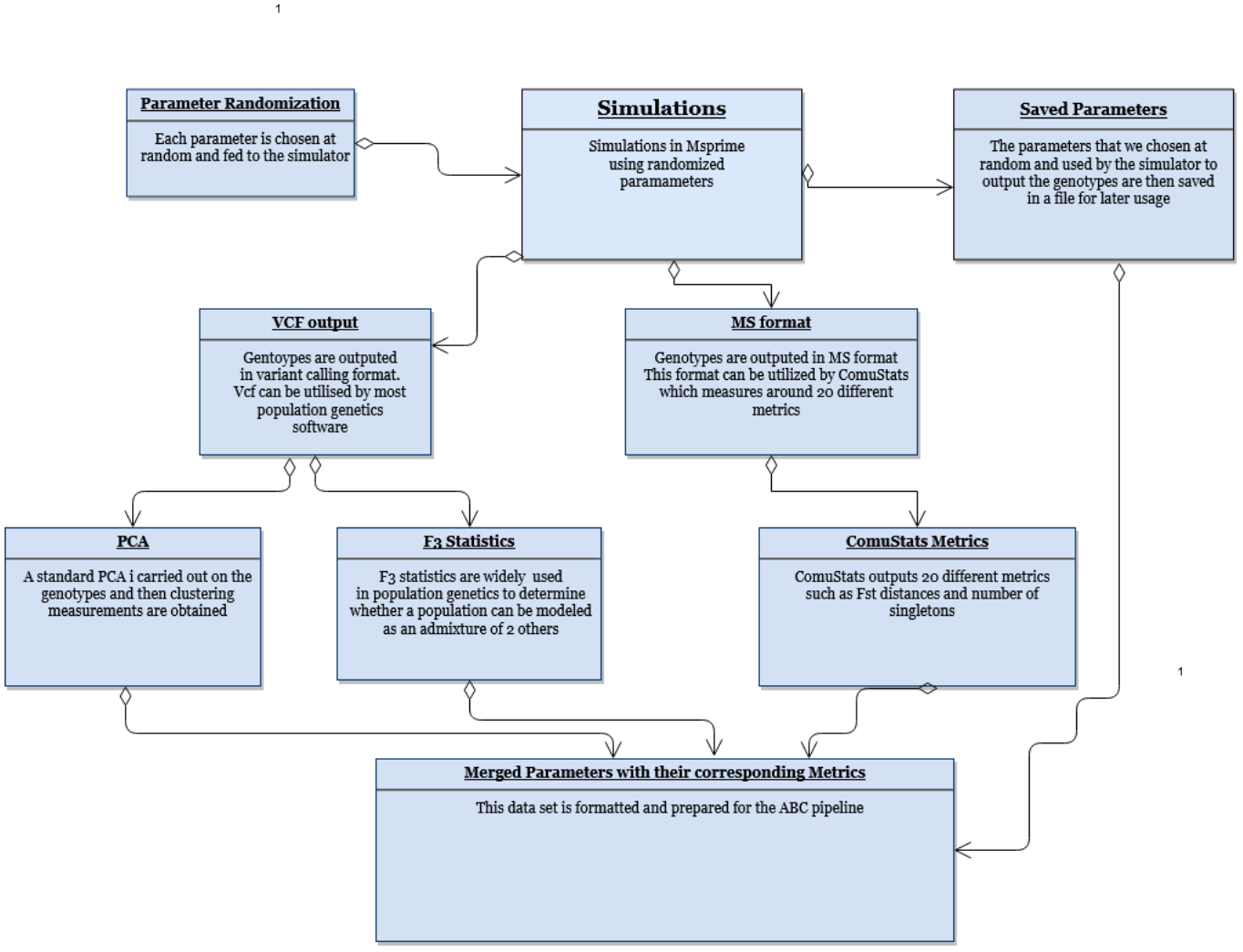


Figure 3.1: Overview of Data Creation for ABC

3.2 Data preparation - Initial Version

3.2.1 Simulations

In order to use ABC, a certain number of simulations were required. For this task the Msprime python-3.7 module was chosen [47].

Msprime - A Coalescence Simulator

Msprime is a backwards coalescence simulator. In simple terms these kind of simulators use the mathematical models of coalescent and re-create scenarios moving from the "present" backwards in time until all individuals of the simulation converge into their Most Recent Common Ancestor. Coalescent simulations are pivotal for understanding population evolutionary models and demographic histories, as well as for developing novel analytical methods for genetic association studies for DNA sequence data.

Coalescent simulators traditionally scale badly in terms of sequence length, especially when combined with recombination and large sample sizes, compared to Sequentially Markov coalescent (SMC) simulators. These in turn however have disadvantages such as decreased accuracy and the discarding of long range linkage information [47].

Msprime stands as one of the newest coalescence simulators available. It is largely based on the ms software build by [41] but offers improvements in many aspects. It has been shown to perform similarly with Sequentially Markov coalescent in smaller genome sizes, with a comparable scaling and apparently scales better with sample sizes.

These numbers alone would make Msprime a sound tool, however it's interface is arguably it's best feature. It can be run natively in python3 and both its parameters and output be seamlessly weaved in one's python code. The user can set up the scenario using python objects and also obtain any information required about the simulation's output by handling Msprime's unique python Classes. These include migration event's, ancestral sampling, Newick trees, mutation information and lists of ancestral individuals. Overall Msprime is highly recommended as a tool for researchers with a modest understanding of python and coalescent as it is a powerful and comprehensive, yet welcoming and easy to understand tool with a smooth learning curve.

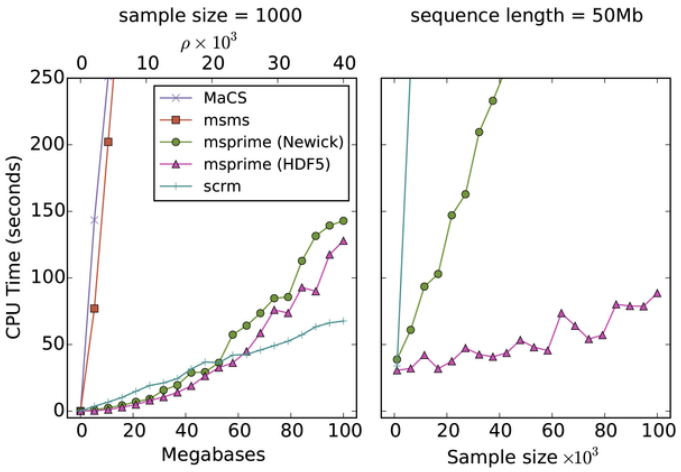


Figure 3.2: Comparison of the average running time over 100 replicates for various coalescent simulators with varying sequence length and sample size.

Setting up a simulation

For someone interested in the actual code of a simulated scenario the following link is provided, leading to a python 3 script on Github where a simple 3 population scenario is performed. Please notice the dependencies required before running it on your machine.

The simulations completed for this thesis follow the same procedure as the one in the link: First, an overall loop is created corresponding to the number of simulations desired. In our case we aim for around 300 thousand simulations. In each loop a new scenario is simulated and the genotypes created by the simulation are outputted into a file.

In each loop before running the Msprime simulation, the parameters for it need to be set up. In our case some of the parameters are stable and unchangeable - the assumptions, and some are to be predicted later through the ABC. The latter are selected at random between a set range, recorded, saved in a file and then supplied to the Msprime simulate function to execute the scenario. In order to represent each value of each parameter equally we used a uniform sampling method. Along with these randomised parameters , a list of populations and their "demographic events" is also required. In our case the only event defined a priory is the creation of a "Colony" population. This is simulated as a population split from either of the two already existing populations referred to here as the "Metropolis" and the "Locals" population.

Each simulation follows a similar tree-like structure for our three populations. A visual example of what we mean by "tree structure" of three populations is given bellow. As we have mentioned each run (loop) uses a different, randomized set of parameters for the same tree structure. A different set of these parameters can result into a completely different scenario and thus to different genotype compositions.

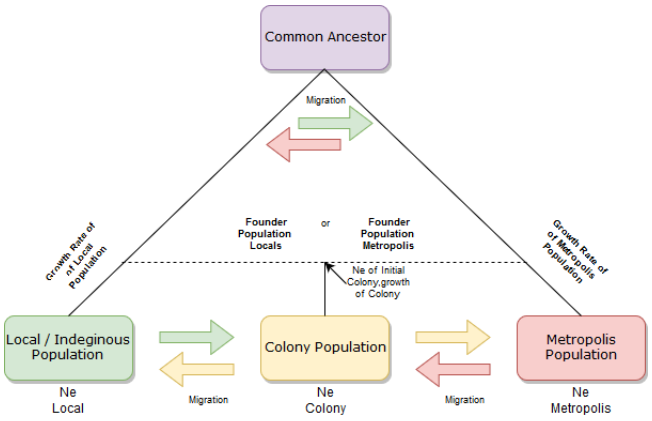


Figure 3.3: Representation of the tree structure of the simulated populations.

For each loop the same simulation, with the same randomised parameters, is run 100 times independently to simulate 100 different independent genomic segments. In order to speed up simulation time these 100 simulations are split into the available threads of the machine through Python's "multiprocessing" package. Once all 100 simulations are complete the segment genotypes are outputted and sewed together into entire genomes.

Bellow you can find a pseudo code representation of this whole process described until now.

Algorithm 1 Simulation Generation

```
procedure OVERALL SIMULATIONS LOOP
    for Loop in number of simulations do
        Parameters ← Randomized
        Assumptions ← Set
        Demography ← Set
        for Second Loop in number of segments do
            SIMULATE(Demography,Assumptions,Parameters)
        COMBINE(Genotype Segments)
    OUTPUT(Genotypes Combined)
```

3.2.2 The Parameters

The parameters that are randomized in each scenario and the range from which their values can be selected, are depicted on the table bellow.

Parameters List	
Parameter Name	Parameter Range
Original Population Size of Colony	200 - 1000
Effective Population Size of Colony	400 - 1000
Effective Population Size of Metropolis	400 - 1000
Effective Population Size of Locals	400 - 1000
Migration Rate Colony to Metropolis	0.0001 - 0.1
Migration Rate Metropolis to Colony	0.0001 - 0.1
Migration Rate Colony to Locals	0.0001 - 0.1
Migration Rate Locals to Colony	0.0001 - 0.1
Migration Rate Metropolis to Locals	0.0001 - 0.1
Migration Rate Locals to Metropolis	0.0001 - 0.1
Growth Rate of Colony	0.0001 - 0.1
Growth Rate of Metropolis	0.0001 - 0.1
Growth Rate of Locals	0.0001 - 0.1
Original Population of Colony	0.0001 - 0.1

In every simulation only three populations are used, each one representing a group of people from our historical scenario: The "Colony" population representing the city of Ambracia, the 'Metropolis' representing the city of Corinth and finaly the 'Local' population representing the various people occupying Epirus at that time. The different parameters used in each simulation are what ultimately causes the differences in genotype composition between each simulated run. These parameters can be divided into 4 groups:

Effective Population Sizes

The first group is comprised of the effective population sizes, sometimes also referred as inbreeding effective population size and is controlled by four parameters. The effective size represents the number of individuals that belong to a population and participate in its gene pool. This measure does not count the actual number of individuals in that population but rather an idealised version of it by making an unrealistic but convenient simplifications such as random mating, simultaneous birth of each new generation, constant population size, and equal numbers of children per parent.

Populations with high effective sizes tend to have overall increased genetic diversity as well as decreased per individual homozygosity. In a natural population the effective and actual population sizes should be correlated, however many factors such as population substructure, can cause them to differentiate. In a city, for example where a 2 class system is taking place and one class does not participate in the gene pool (e.g. slaves that are not allowed to intermix with free-men) one would expect to the effective and actual population of the city to differ greatly.

There are four parameters in this group. The first three are corresponding to the effective population size at the time of sampling for our three populations: Metropolis, Locals, Colony. As we mentioned these do not necessarily represent the actual size of these populations but never the less can be used as a rough estimate for them, especially for making comparison between the three populations (e.g. the Metropolis is roughly two times that of the Colony). In our simulations all 3 populations represented either large cities (Corinth, Ambracia) or entire tribal groups (Local Epirotes). In order to represent that, we chose to constrain our effective sizes between 400 and 1000. On the lower spectrum of the range the effective size is enough to represent a large city of antiquity, while on the higher side a group of cities or entire region.

The fourth parameter is the original population size of the colony. The colony has two different effective population size in order to simulate its creation. If the initial population of the colony is much smaller than that of the Metropolis and it reaches a much larger final population size then that is a unique scenario, different than one where the final size is similar to the initial but both much smaller than the Metropolis. In coalescent terms a severe but short-lasting decline in effective population size is termed a population 'bottleneck'. Bottlenecks are normally associated with external catastrophic events such as an ice age or severe disease, but they can also be associated with the colonisation of a new habitat by a population. This fourth parameter is used for exactly that reason, to simulate the bottleneck (or lack of) associated with the creation of a colony. For logical coherency we also restricted the effective size of the colony to be smaller than that of its origin population.

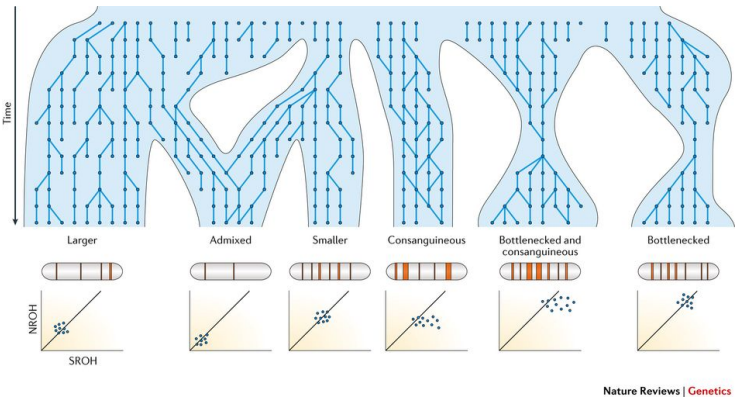


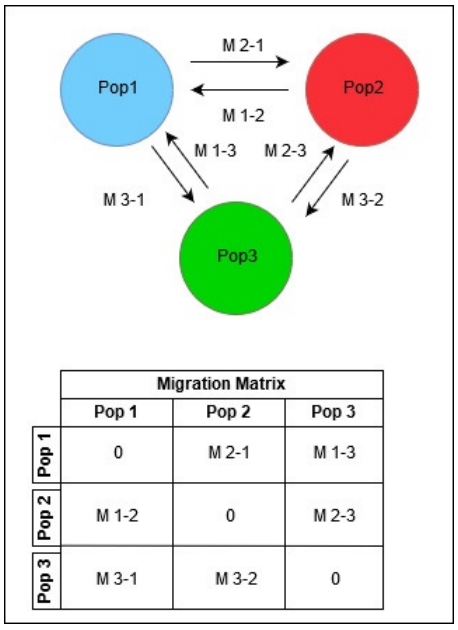
Figure 3.4: Example of effects of population size to number and length of ROHs. ROH stands for Runs of Homozygocity [17]

Migration

The second group of six parameters is comprised of the migration rates between these three populations. With the words human migration one can refer to a number of different things: The prehistoric out of Africa spread of humans over the globe, the transmission of farming from Anatolia to the all of Europe in the Neolithic, the ravaging hordes of "barbarians" of late antiquity, even the asylum seekers leaving the war torn countries of today. Movement of entire populations, groups of people or individuals has always been a core function of mankind in its history. In the confines of this thesis, the word

migration is used from a coalescence viewpoint, to describe the "jump" of an individual and his lineage from one population group to another.

In coalescence simulations individuals living inside the same population are free to admix with each other but not with individuals from other populations. This is where migration rates come in: for each passing generation a certain percentage of the population can be migrants from another population. This way we can simulate contact between groups of people and gene flow between populations in our simulations. Once an individual has migrated to a new population he or she is considered by all means a normal member of that population.



Migration Matrix			
	Pop 1	Pop 2	Pop 3
Pop 1	0	M 2-1	M 1-3
Pop 2	M 1-2	0	M 2-3
Pop 3	M 3-1	M 3-2	0

Figure 3.5: Visualisation of migration rates and their position on the migration matrix

In our simulations six migration rates where used. The Colony population for example, has two migration rates, one to represent the percentage of individuals originally from the Metropolis population and one for the people originally of the Locals population. When referring to a migration rate two population names are given, in this thesis the first name always corresponds to the recipient population. Thus a value of 0.1 for migration rate Colony - Metropolis means that 10 percent of the Colony population for each generation are migrants from the Metropolis population. The migration rates have a range between 0.0001 and 0.1 to simulate either an almost non existent contact to an intense migration event. Normally migration rates can change through time, but for simplicity reasons as well as the small time frame of our simulation, once the migration rates have been set at the initiation of the simulation, they cannot be changed until its end.

For those looking for the migration rate inside the code, they can be found inside what is known as the migration matrix, a simple matrix with rows and columns equal to the number of population. Each cell of the matrix corresponds to the migration rate between Population number Row - Population number Column with the diagonal cells always equal to zero since they represent migration from a population to itself.

The migration rates are probably the most important parameters used in the simulation, as they can create fundamentally different scenarios, even when not interacting with the rest of the parameters.

Growth rate

The third group of parameters are the growth rate of the populations. One growth rate parameter per population controls the speed at which the population increases in size. It is known that populations on the rise or who have recently gained size quickly have distinct genotype composition [65]. Furthermore a population with a recent bottleneck as well as growth can differentiate greatly from its former self due to intense genetic drift. If with couple this also with migration from an outside source, this could easily lead into a population looking nothing like its predecessor. This is a fairly important point, as our scenario, the Coloniation of Epirus could very well stand within this hypothetical.

For the purpose of this work for each simulation, three values are chosen at random to determine the growth rates of each population. These are chosen between the values of 0.0001, simulating a de facto stable population and 0.1, simulating a population explosion.

Origin of The Colony

For our final parameter we attempted to explore a more difficult question rather than just statistical inference of numerical parameters. As we have mentioned in the previous chapter, it is known that the Corinthians are the people who set up the initial colony of Ambracia however it is not quite clear from historical and archaeological evidence what the original population of the colony was. In order to cover both scenarios, of a Corinthian or a Local original population each simulation has a random value of 0 or 1 with a 50 percent chance. In half of the scenarios the colony population is created a split from the Metropolis population and in the rest from the Local population.

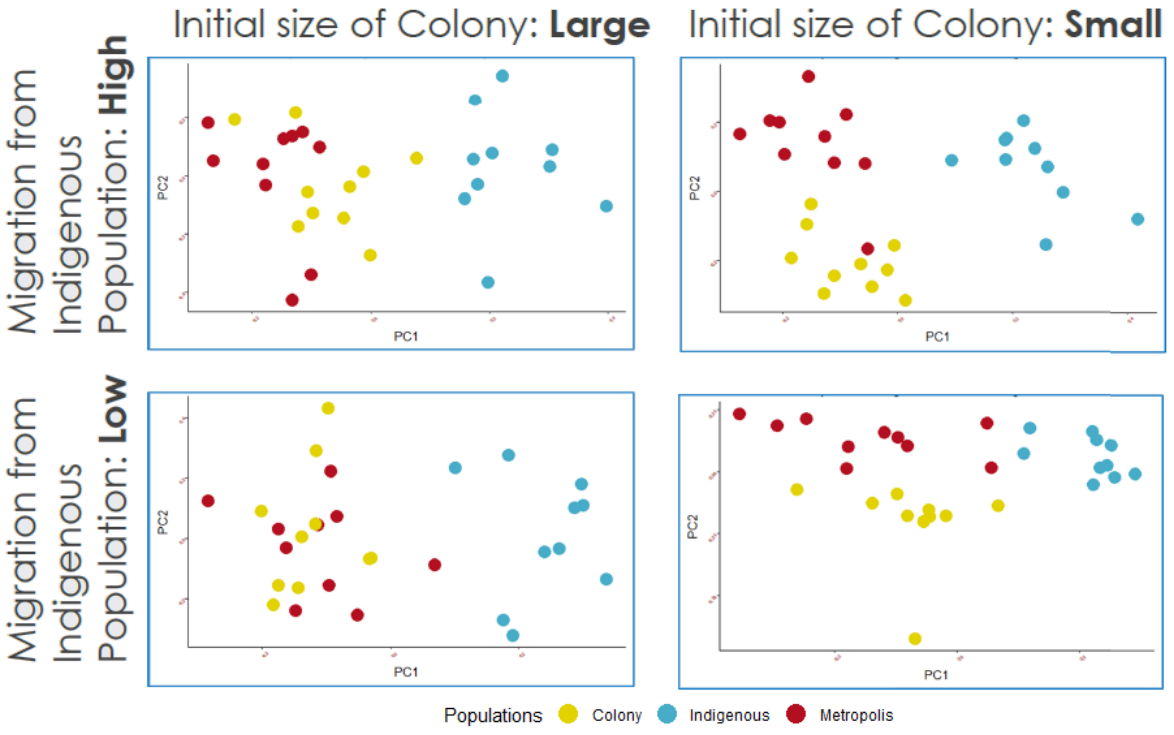


Figure 3.6: PCA examples of how population size coupled with migration can effect a population. Here the Colony population is always a split from the Metropolis

3.2.3 The Assumptions

Along with the parameters that are changing in each simulation run there is a set of parameters that remain static, unchanging for all of the simulations. As we have mentioned before these are assumed to be of a certain value and unchanging through the different scenarios. There are two basic assumptions taken in our simulations.

Mutation Rate

The mutation rate is one of the most basic parameters required to run a Coalescence simulation. Simply put, the mutation rate is the probability of a new mutation occurring in each generation, per individual and per genetic position. The higher the mutation rate the quicker two population diverge from each other.

In the past few years efforts have been made to calculate the mutation rate in humans. This has been done by either direct observation of modern people and their offspring, comparing divergent lineages by using fossils as time records or population genetic approaches. These have yielded different between 1 - 2*10 $^{-8}$ [8, 12] as a mean overall mutation rate in humans. In our work, we used the more recent results of a 1.45*10 $^{-8}$ [61] and set our parameter accordingly.

Generation time

Coalescence Theory does not use the conventional metrics of time. Instead everything is calculated in generations. In order to make real world predictions, one has to convert a number of generations to conventional time measurement. This is usually done using a Generation Time simplification, which assumes that all individuals of a species have a certain amount of time from when they are born to when they leave their first offspring. This of course is not completly accurate, however its serves its puprpose, as it allows us to easily convert generations to time. In this thesis a standard generation time of 20 years was used for all simulations.

Genome Size and Recombination

We decided to simulate genetic segments of a 500.000 bp size. For each simulation 100 segments were generated with the same parameters leading to genomes of 5.000.000 bp in size. Since we wanted to simulate independent segments we used no actual recombination in our calculations, which is available through Msprime.

3.2.4 Obtaining the metrics

After every simulation is finished the genotypes are outputted in two formats, a Varriant Calling Format (VCF) and an MS format. The VCF output is the automatic output generated by Msprime, while the MS format is created using python by handling the genotypes through some custom lines of code.

Once the genotypes have been outputted in the two different formats, they can be then utilized by other software to obtain the metrics or 'summary statistics' in ABC language. With the exception of the PCA, each software generated the statistic 100 different times, one for each segment. The summary statistics selected for the first ABC test, along with the software used to generate them are he following:

3.2.5 PCA Clusters

Principal Component Analysis is one of the most used exploratory tools in population genetics. It is able to cluster together individuals that share a genetic similarity. Our initial idea was to use this information, obtained through the eigenvectors, as a summary statistic in our ABC. PCA was performed on the whole stitched-together genomes using PLINK v1.9 [18] and the vcf file as input. After obtaining the first 10 eigenvectors of the PCA a custom clustering was performed using python.

Our aim was to obtain a metric of how tight or loose each population clustered as well as how close or far away the center of each population cluster was from the others. For each population the mean of its eigenvectors was calculated, between the individuals belonging in that population and used as the center of that population. Then the average distance of an individual of the population from the center of that population was calculated. Finally the distance between the centers of the different populations was also calculated. In total six metrics were obtained: three describing the clustering within each population using the average distance from the center and three between the populations using the distance between the centers.

3.2.6 F3 Statistics

F3 statistics were also chosen as a summary statistic since, as we talked about in the introduction, they are a very popular tool, known for providing indication of admixture and are also very rigorous.

For the F3 statistics the vcf output was converted to eigenstrat format using Convertf. Once this was complete, the new eigenstrat file was used as input for the qp3Pop software, using the default settings, to calculate the f3 statistic for our populations with this ordering: Locals, Metropolis, Colony in order to test the Colony population as a mixture of the other two populations.

3.2.7 CoMuStats

Finally the MS format file was used as input for CoMuStats. CoMuStats was created as an extention of CoMuS, a custom Coalescence simulator created by Pavlidis [76]. It is largely based on msABC [66] with some extra statistics. It's purpose is take MS formatted genotypes and output a number of summary statistics to be used for ABC in particular.

CoMuStats was used with the settings -npop 3 20 20 20 -ms which outputs 20 different metrics. The output of CoMuStats is file with rows equal to the number of simulations. Every row is divided into columns, with each column representing a metric. In total this basic run of CoMuStats generates 20 different metrics. Thus from each simulation we generate a matrix of 100 rows and 20 columns.

Distributions of Summary Statistics

Now that we posses 100 measurements for our summary statistics, for each simulations, we need to convert them into something more useful for our ABC and later for our classification problems. The standard way of converting these 100 values would be to use a metric such as their mean, mode or median. Indeed both for the f3 statistics and for each individual metric generated by CoMuStats we calculated the mean for further use. However we decided to translate our 100 measurements with a second, novel method.

Each simulation/scenario generated a 100 measurements for each metric. These correspond to a distribution, unknown to us and specific for the simulation/scenario used to generate the 100 measurements. Because two different simulations might generate two different distributions, but these distribution have the same mean or because in some other cases means don't represent well the density of their distribution, we chose to use a novel method that would better represent the whole distribution rather than just the mean.

Our idea was to first select certain values and use them as set points which would define all of the distributions for a certain metric. After all the simulations finished, for each summary statistic we isolated the minimum and maximum values among all of the simulation runs. For example for the f3 statistics we had 300.000 simulations each with 100 measurements, leading to a total of 30.000.000 values of f3 statistic. From those we selected the minimum and maximum values. These were stored and the same process was done for all summary statistics.

Now that we have the minimum and maximum for each statistic, we will use them as anchors points to get the distribution. We select 9 more points which are equally spaced between these 2 points for every summary statistic. Now we have 11 points - values for every summary statistic. For every simulation, thus for every distribution of 100 measurements, we can sample that distribution at these 11 points, obtain 11 probabilities and get a ruff estimate of what the distribution looks like.

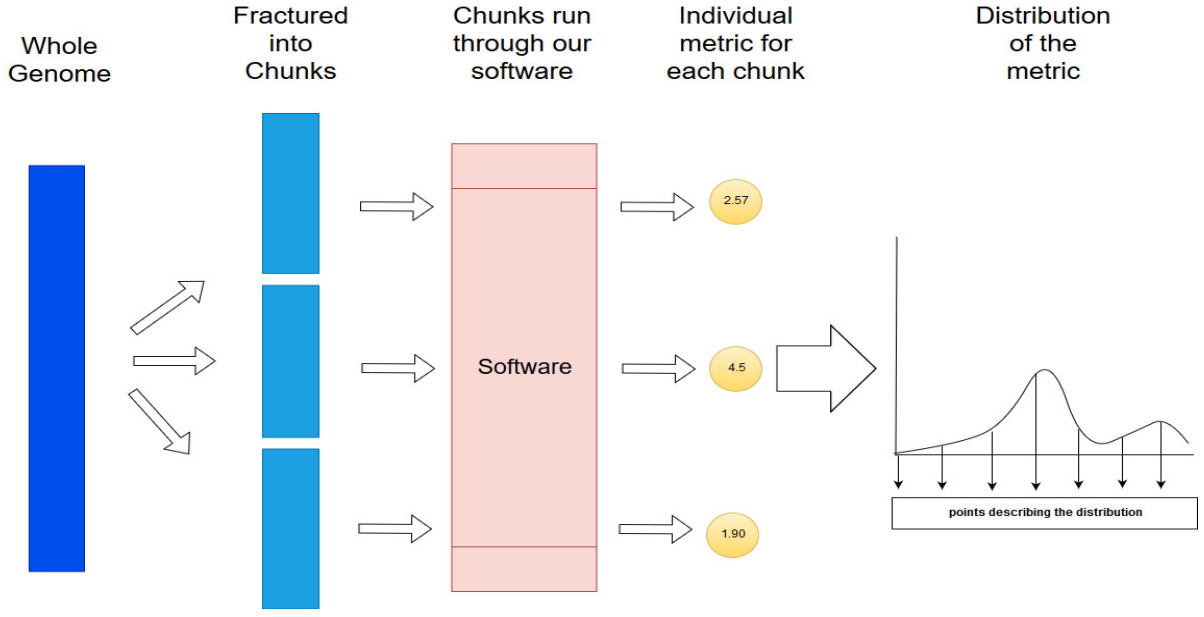


Figure 3.7: Process of acquiring distribution points from chunks instead of singular values for each metric

To get the actual distributions, the metrics where loaded into R where the density function was used. Using the density function, 11 points where then extracted from the distribution of each metric within a certain range (defined by the minimum and maximum of that metric from all the simulations). This points are enough to explain the distribution to a certain level (see figure above).

3.2.8 The Problem

After completing our pipeline we tested it using a small number of simulations. Quickly the main problem with this method became apparent: our pipeline was very slow. One round of simulations and summary statistics calculation took an average of INSERT NUMBER HERE. This was unacceptable since our aim of 300.000 simulations would take an approximate 27 weeks. Since all of our efforts would be based on this pipeline, we went back to the drawing board.

After investigating which processes slowed down the entire pipeline it was decided that a rework was required.

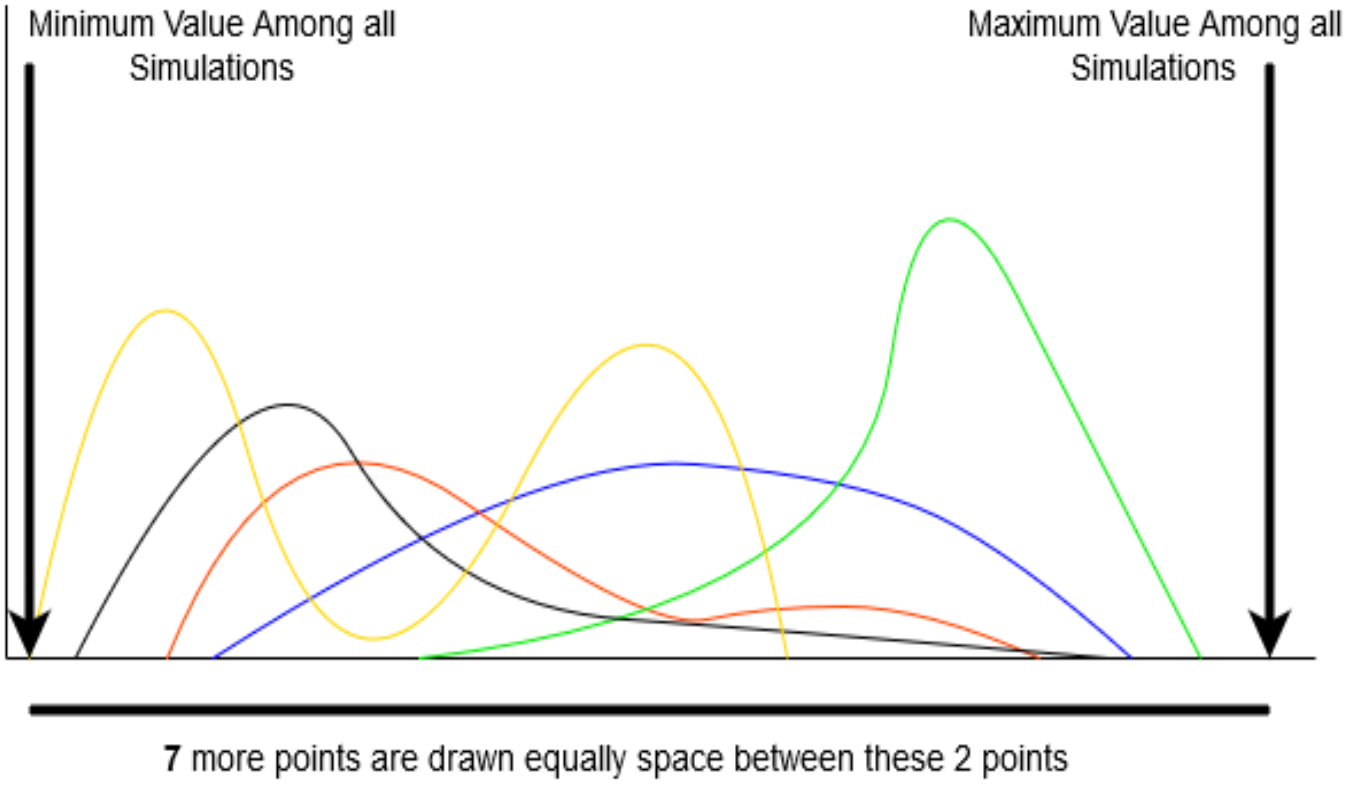


Figure 3.8: Visual example of different distribution. Each different colour represents a distributio of a summary statistic for one simulation. The arrows on the edges represent our 'anchor' points, all distributions fall within these boundaries.

3.3 Data preparation - Reworked

From our assessment the simulation were actually running smoothly and at an acceptable speed. It is the post genotype generation part of the pipeline that provided the delay. Both the multiple conversions between formats and some of the software used proved inefficient in terms of computational speed.

After some consideration it was decided to limit software usage to that of CoMuStats. This of course meant that we would lose an important part of our summary statistics. To compensate for that, we decided to modify the way we ran CoMuStats in two ways, which we will be discussing bellow.

3.3.1 CoMuStats Modifications

Since we decided to remove the usage of Plink for the PCA and of qp3Pop for the F3 statistic we decided to make our CoMuStats runs more complex in an effort to make its summary statistics more indicative of each scenario.

To do that we first changed the parameters used in each CoMuStats run. The flag ı-sepPops was used to separate each population. Previously CoMuStats would generate 20 summary statistics that described the whole data set containing all three populations, whereas now these 20 summary statistics would be generated 4 times: one for the data set as a whole and once for each population independently. For example the summary statistic 'Number of Singletons' would be generated once by counting the number of singletons in the entire dataset, once for the Metropolis population, once for the Local population and finally once for the Colony population. This way we raised our summary statistics from 20 to 80.

In addition to this, the flag ı-pairwiseFst was used in connjuction withı-sepPops to calculate the Fst distances between each population, adding 3 more summary statistics.

Figure 3.9: Artwork for CoMus and CoMustats. You can find more information on CoMuS here

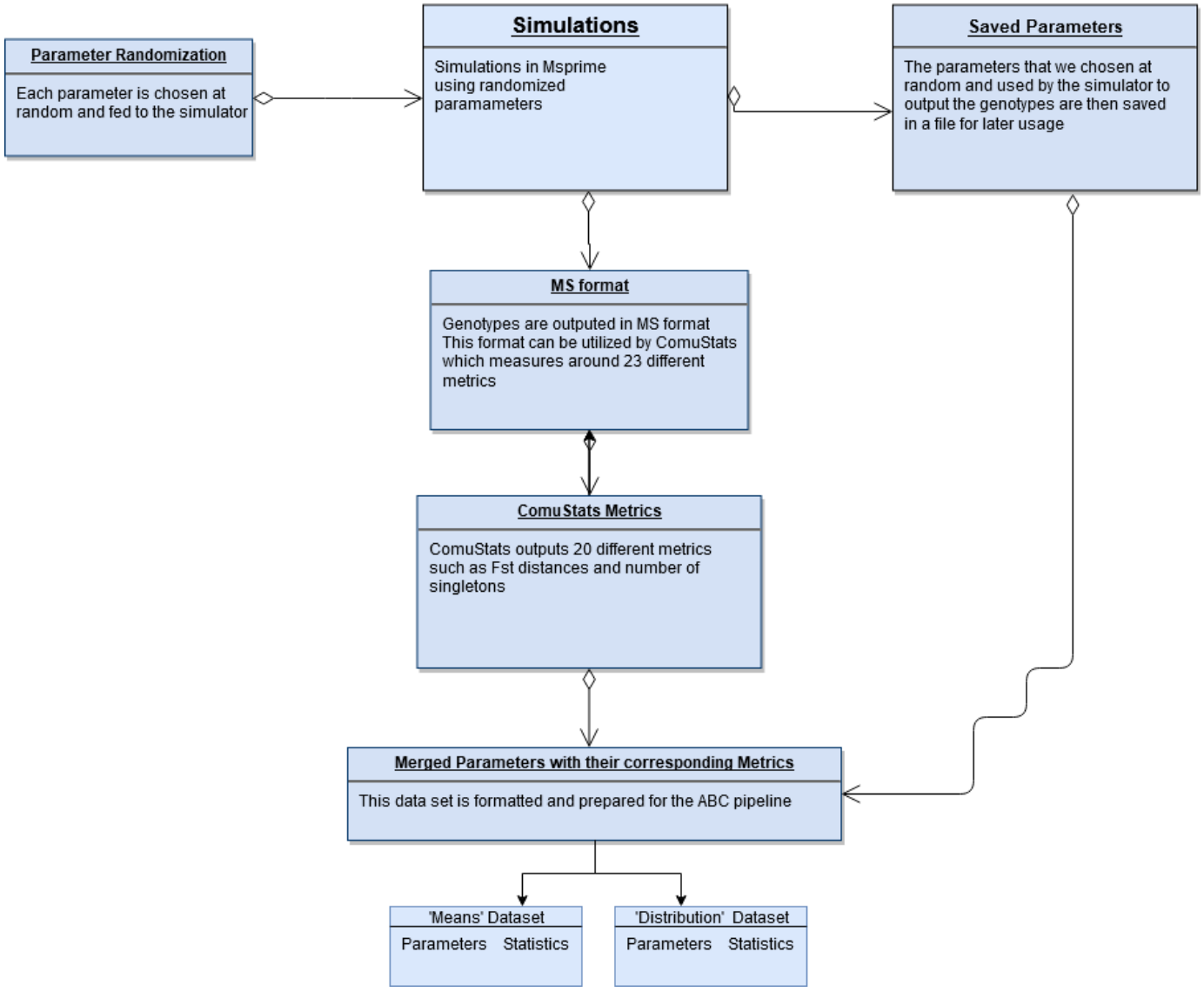


Figure 3.10: Overview of Reworked version of Data Creation for ABC

Finally we modified the code of CoMuStats, just for this thesis, by adding the calculation of F2 and F3 measurements for 3 populations by using the new flag -f3 pop1 pop2 pop3. The F2 statistic is, as we mentioned in the introduction, very similar to the Fst however since it is required to calculate the F3 we added it as well. Each pair of populations now also generates an F2 measurement. Also a user defined triplet of populations, with the first population inserted tested as product of admixture of the other two, also generates a F3 measurement. By this addition we extended our summary statistics to 87.

The final command line used for CoMuStats looked like this

ı-npop 3 20 20 20 -pairwiseFst -ms -sepPops -f3 3 1 2

with population number 3, the Colony being the one tested for admixture.

3.3.2 New Data preparation

Once we got our customized version of ComStats with the new flags set up, we were ready to repeat the pipeline. This time the speed of the simulations and summary statistics generation was deemed sufficient with an average run of 5.5 seconds We aimed at 300.000 simulation runs, generated and stored parameters, genotypes and summary statistics. Once this was complete we tranformed each simulation's 100 segments with the 2 methods detailed in the previous section. This time the process was simpler since all of the summary statistics were already merged together in one file, the output file of ComMuStats.

Again the CoMuStats output file for each run was a matrix with 100 rows but with 87 columns this time. With the process described in the previous section each such matrix was transformed a into a)a vector of means with 87 dimensions and b) a vector with 11 values for each summary statistic, representing its distribution, thus with 957 (!) dimensions. In the end we combine all these vectors into two final matrices one with 300.000 rows and 87 columns and one with 300.000 rows and 957 columns. For clarity we will refer to the first former as 'Means Summary Statistics' and the later as 'Distribution Summary Statistics'

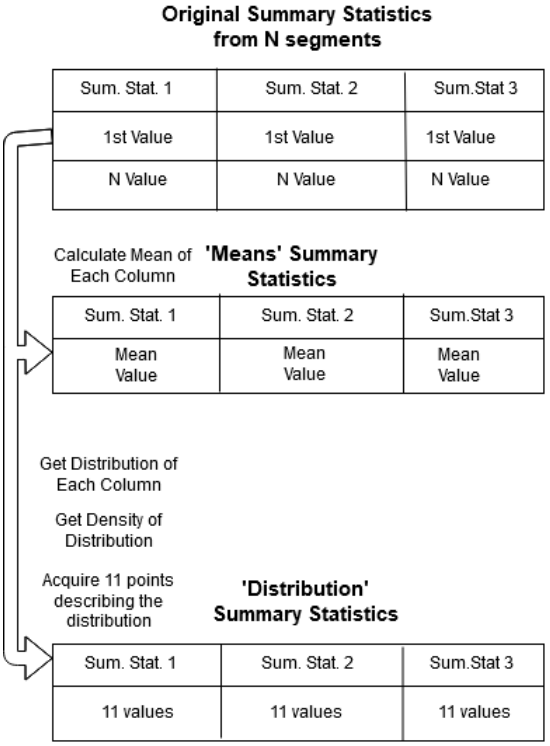


Figure 3.11: Again, the creation of the 2 summary statistics matrices

3.4 ABC Predictions

To prepare our final two data sets for ABC we need only one more thing: to merge each simulation's parameters with its corresponding vector of summary statistics. Before doing so however, we decided to create a matrix which contains the maximum and minimum values for each summary statistic. This will be used later in the ABC. All the parameter files were scanned and their minimum and maximum values calculated and stored in a new file, referred to as the 'Logit' file.

Finally one parameter was left out of the ABC estimations since it differed from the others. The parameter 'Original Population of Colony' is a categorical / binary parameter, rather than a true numerical one and thus could not be inferred at the same time as the others.

Having merged the parameters with the two summary statistics matrices we are now ready to run the ABC inference.

3.4.1 Running the ABC

For our ABC parameter inference the 'abc' package of R was selected to be used. Each one of the two matrices created was independently loaded into R. Along with them, the Logit file was also loaded. The process described bellow was followed for both of the matrices.

First a 10000 rows were isolated from the matrix and stored into a new matrix. This matrix will be called the test data set. The remaining 290000 rows will be used as the training data set. Now we will test the predicting power of our method by estimate the parameters of each row of the test data set. We created a 10000 repetition loop where in each iteration one row from the test data set is selected, split into a parameters and a summary statistics vector and where the later is inserted into the abc function.

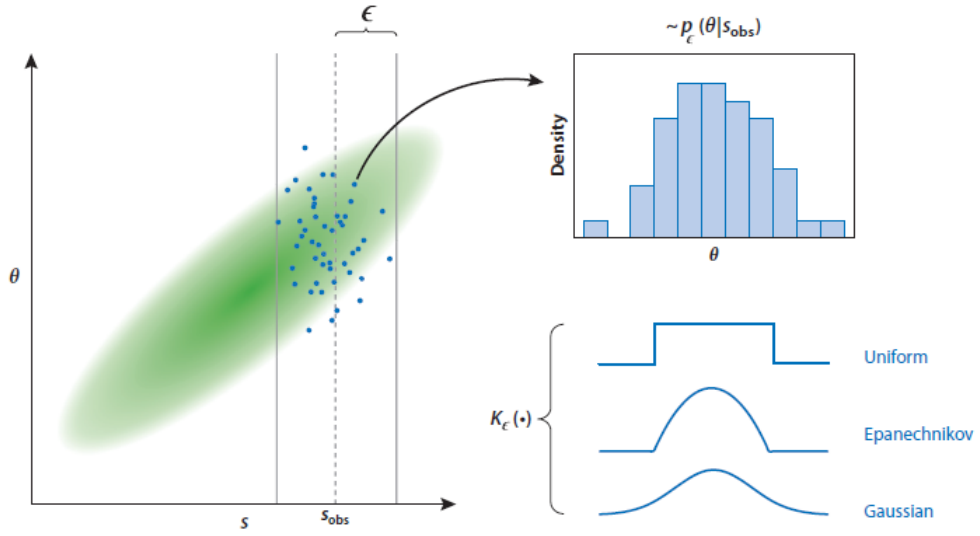


Figure 3.12: From [10] here the ABC algoirthm is showcased. The Y axis represents the parameter theta and the X axis represents the summary statistics. Each dot is a simulation. After removing distant simulation the rest of the thetas are placed in a distribution and the uknown theta calculated using that distribution.

Each time the abc function uses the training data set to predict the parameters of the tested vector. Here we utilised the data of the Logit file created before, as a "logit matrix" for the abc function. This matrix does not permit the function to predict values for each parameter, outside of the minimum and maximum values provided by the matrix. The abc function was also executed using the 'ridge' method of the package. After some initial tests with different levels of tolerance the 0.15 chosen as the standard for the rest of the predictions.

This means that only 15 percent of the total simulations were used for the prediction of the values, leading to around 19300 simulations being utilised for the regression step. For each repetition of the loop the abc function runs with the described parameters and estimates the parameters of the test vector. The true parameters are outputted along with the predicted mean, median and mode output values of the abc function and stored into a new file. Once the process is finished we can apply our own custom error measurements to asses the effectiveness of our predictions. The above process is a custom,manual application of the so called "leave one out" validation method.

3.5 Scenario Selection

We quickly realised that the aforementioned 'Original Population of Colony' parameter could not be inferred the same way the others could through normal ABC. This problem was more of a classification problem rather than a regression one. After we complited our parameter estimations for 13 out of our 14 parameters we decided to test some of the available classification algorithms for this last parameter.

3.5.1 The Two Different Scenarios

All of our simulations belong to one of two scenarios, with the parameter taking the values: 1 where the original population of the Colony is an offshoot of the Metropolis population and 0 where it is of the Local population. However migration from the non-parental population coupled with a small effective population size or bottlenecks can greatly skewer the identity of the population. This is where classification algorithms are required, should there be some differentiating summary statistics or a combination of them, then some of the available algorithms would be able to correctly identify the scenarios and thus this parameter.

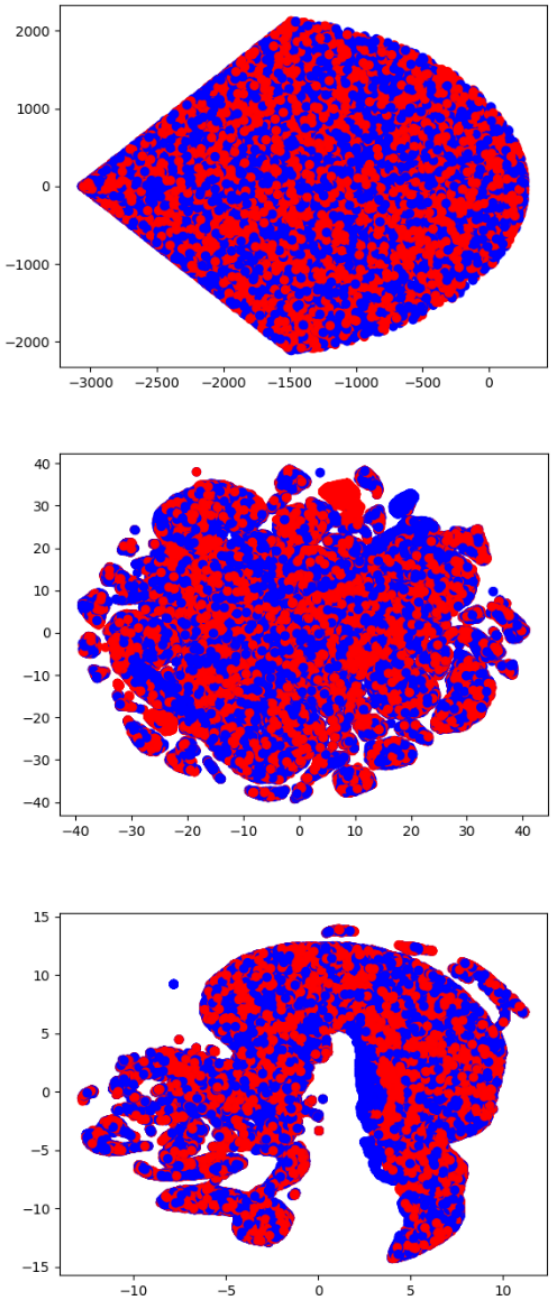


Figure 3.13: Visualization of the two models using a)PCA b)tSNE c)uMAP on the Summary Statistics of each simualtion (red : model 0 - blue : model 1.

3.5.2 ABC Random Forests

Our first attempt at classification came in the form of another ABC package of R named 'abcrf' [70]. This package combines the classic ABC algorithm with a random forest classifier. Our use of the package mimicked our previous ABC runs. Again we loaded each of the two summary statistics/parameters file separately into R. For each file we split off 10000 rows to be used as our test / validation data set. We used the rest of the rows as input for the abcrf function with default settings and only changing the number of trees parameter with values first between 100 and 1000 and then between 100 and 10000. This functions has an internal error estimation. Nevertheless we also performed an outside validation using our test data set, using the best performing number of trees from the internal error estimation.

Leaving R behind we moved on to one of the most used python 3 collection packages: 'scikit-learn' more commonly known as simply sklearn. There is multitude of available ML classification tools provided by sklearn. Theoretically we should have attempted to use most, if not all of them to select the best one. Because of time constrains however we chose to use two of them: the Random Forests Classifier and the SVM classifier.

3.5.3 SKlearn Random Forests

We chose to compare our ABC random forest (RF) classifier with sklearns RF classifier. This time we chose to omit using the Distribution summary statistics since the ML methods are notorious for their computing time which scales with the number of samples and feaures (sumamry statistics) and the particular data set contained more than 900 features, as we saw previously.

Like previously we loaded our data sets, this time in python3. Unlike preciously however, this time we performed what is known as model and feature selection before using the classifier. We made use of the 'pipeline' function of Sklearn which allows one to perform a number of procedures, step by step on a data set.

The pipeline contained an initial step of normalisation using sklearn's 'StandardScaler', which removes the mean of each feature and scales to variance. Then, features with low variance are removed using sklearn's 'VarianceThreshold' with a threshold of 0.16 . Finally a certain number of hyperparameters are tested in all of their combinations in the RF classifier using sklearn's 'GridSearchCV' and the provided hyperparameters. The hyperparameters tested can be seen in the figure bellow. This was validated using an inner Kfold validation of 10. The best performing combination of parameters from the inner Kfold is then used in an outer Kfold validation of 5 to estimate the final error rate and the overall performance of our classifier.

Hyper-Parameters List	
Hyper-Parameter Name	Options
Criterion	'gini' / 'entropy'
Number of tress	100/150/500/1000
Max number of features considered each split	'sqrt' / 'log2'
Depth of trees	20/30/40/50/'None'

3.5.4 SKlearn Support Vector Machines

As an additional classifier other than random forests, we selected sklearn's SVM. We followed the same method as aboce using again the pipeline function with StandardScaler, VarianceThreshold and the GridSearchCV. Again we used both an inner and outer Cross Validation by selecting and using the best hyper parameters.

Hyper-Parameters List	
Hyper-Parameter Name	Options
SVC C	1/2/5/7/10
Kernel	'rbf' / 'poly'
Max number of features considered	20/30/50
Degree for Poly kernel	1 / 3

Chapter 4

Results

In this chapter we will look at the results produced by this thesis. We will first look at the output of our simulations and then move on to the error rates for the ABC predictions and finally for the classifications.

4.1 Simulation Results

After completing 300 000 simulations and obtaining the summary statistics on them in about 450 hours or 19 days, we decided to test whether our simulations covered the desired range of values for the parameters. Besides covering the expected range of the parameters, we also want these values to idealy be represented in a uniform distribution. This was not true in our case for all parameters, as we will see bellow and could potentially interfere with the results, which we will discuss in the next chapter. Overall we can see that in most of the parameters do have a uniform distribution with the exception of the Ne of the Colony both Initial and Final.

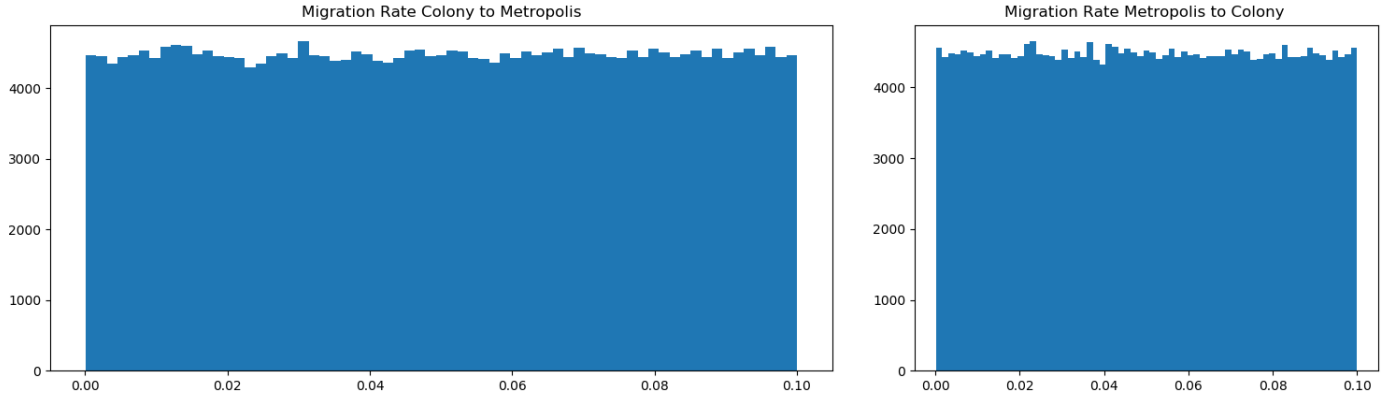


Figure 4.1: Prior Distribution of 2 Parameters

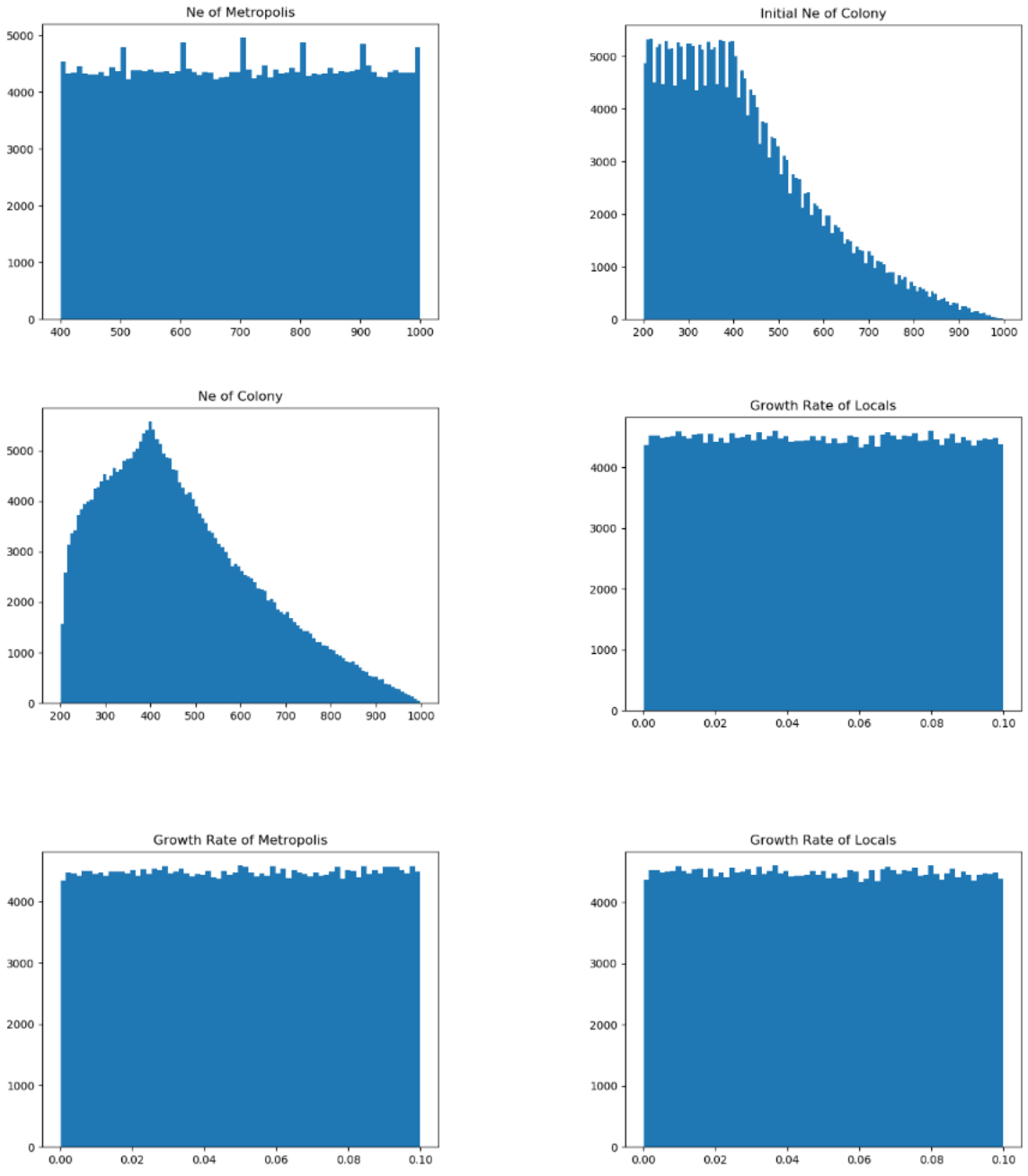


Figure 4.2: Prior Distribution of 6 more Parameters

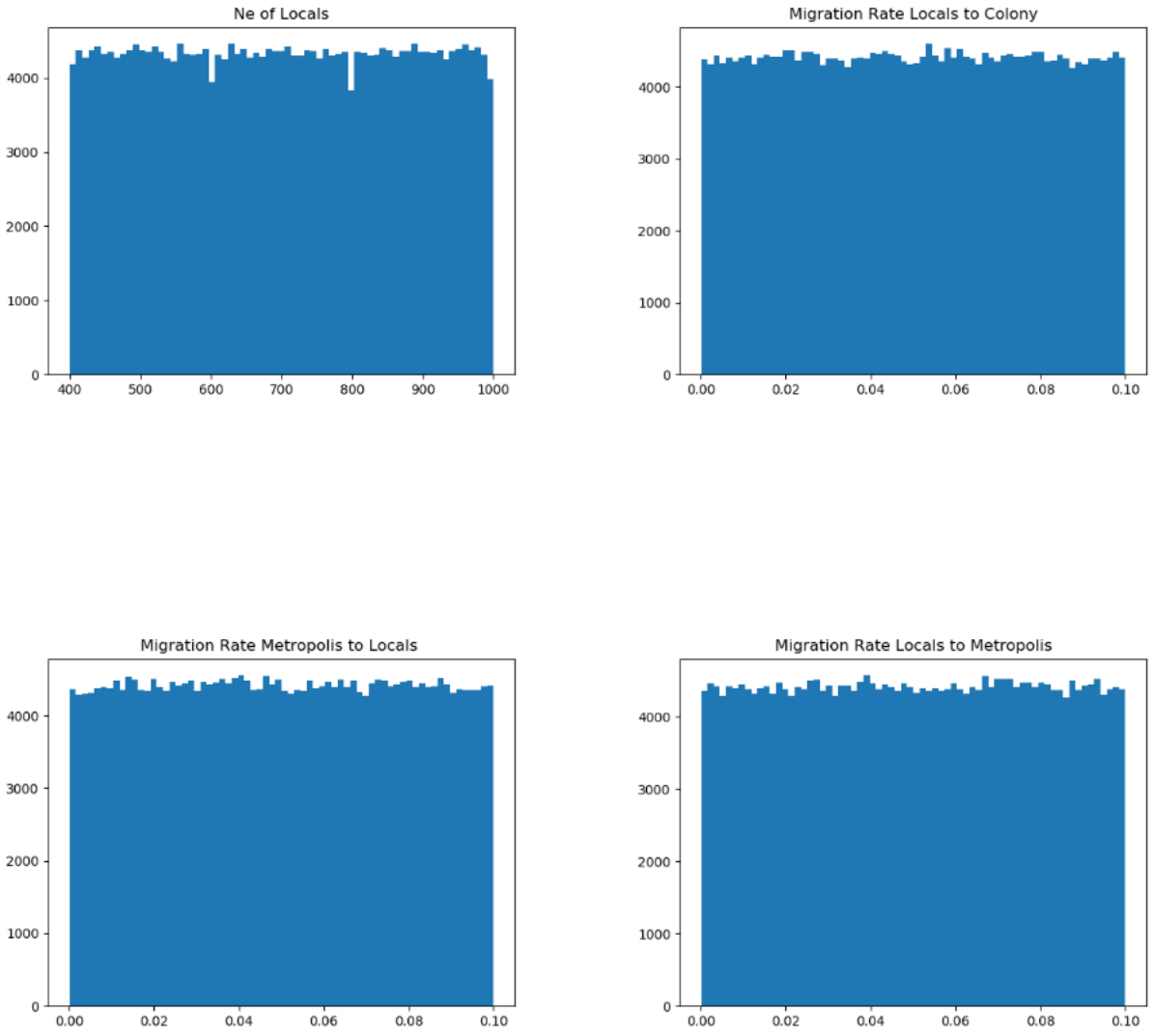


Figure 4.3: Prior Distribution of final 4 Parameters

4.2 ABC Results

After completing 10000 leave one out predictions for each of the two summary statistics data sets we calculated their prediction errors in order to asses their estimations. For the error estimation we used the Root Mean Squared Error divided by the maximum - minimum of the parameter. First we calculate the square error for each individual prediction using the formula bellow:

$$IndividualError = (Actual - Predicted)^2$$

Then, once we have this measurement for each prediction we calculate the root of the mean of those squared errors and finally divide it by the difference of the maximum and the minimum the parameter can take. This final step is used to normalise the error rates between parameters.Bellow you can find the formula for the Root Mean Squared Error (RMSE).

$$RMSE(PredictedParameter, AactualParameter) = \frac{\sqrt[2]{mean(IndividualOfErrors)}}{MaximumOfParameter - MinimumOfParameter}$$

Having calculated the RMSE for each parameter individually and for each one of the two data sets, we created the tables in the following page to depict and compare them. We selected the mean for each predictions distribution as it yielded the best results, when compared with the mode or the median.

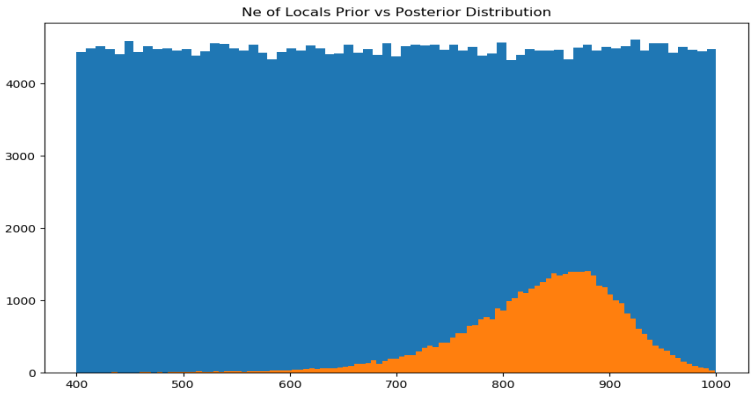


Figure 4.4: Example of both Prior (blue) and Posterior (orange) Distribution of a parameter. Here the prediction would be either the mean, median or mode of the orange distribution, around 850-900.

Parameter Error Rates - Means Datset	
Parameter Name	RMSE
Original Population Size of Colony	0.144
Effective Population Size of Colony	0.143
Effective Population Size of Metropolis	0.201
Effective Population Size of Locals	0.207
Migration Rate Colony to Metropolis	0.157
Migration Rate Metropolis to Colony	0.150
Migration Rate Colony to Locals	0.162
Migration Rate Locals to Colony	0.126
Migration Rate Metropolis to Locals	0.129
Migration Rate Locals to Metropolis	0.119
Growth Rate of Colony	0.163
Growth Rate of Metropolis	0.172
Growth Rate of Locals	0.163

Parameter Error Rates - Distribution Date set	
Parameter Name	RMSE
Original Population Size of Colony	0.157
Effective Population Size of Colony	0.194
Effective Population Size of Metropolis	0.248
Effective Population Size of Locals	0.220
Migration Rate Colony to Metropolis	0.264
Migration Rate Metropolis to Colony	0.313
Migration Rate Colony to Locals	0.226
Migration Rate Locals to Colony	0.229
Migration Rate Metropolis to Locals	0.247
Migration Rate Locals to Metropolis	0.158
Growth Rate of Colony	0.218
Growth Rate of Metropolis	0.223
Growth Rate of Locals	0.216

In the following page we also present examples of the ABC prediction process. In each individual plot we have the 300 000 points representing the simulations. In the X axis we have a certain Parameter and in the Y axis a certain Summary Statistic. We are thus looking at how using the rejection of simulation the ABC is able to infer the parameter in question. The center of the Circles is our prediction for the parameter based on that summary statistic.

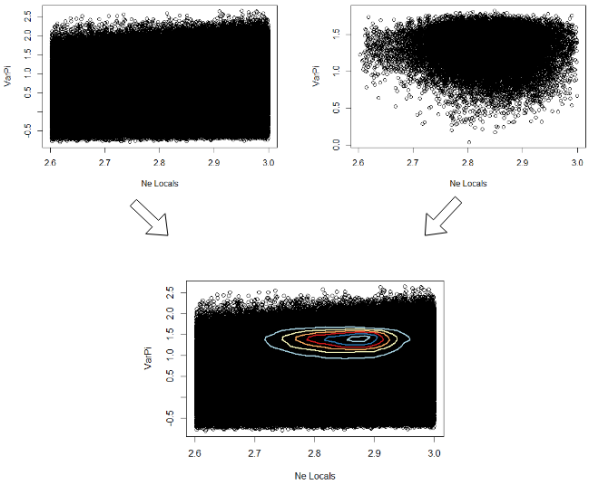


Figure 4.5: Visualisation of how the next plots are created. The Posterior distribution is combined with the Prior into a new plot that displays both.

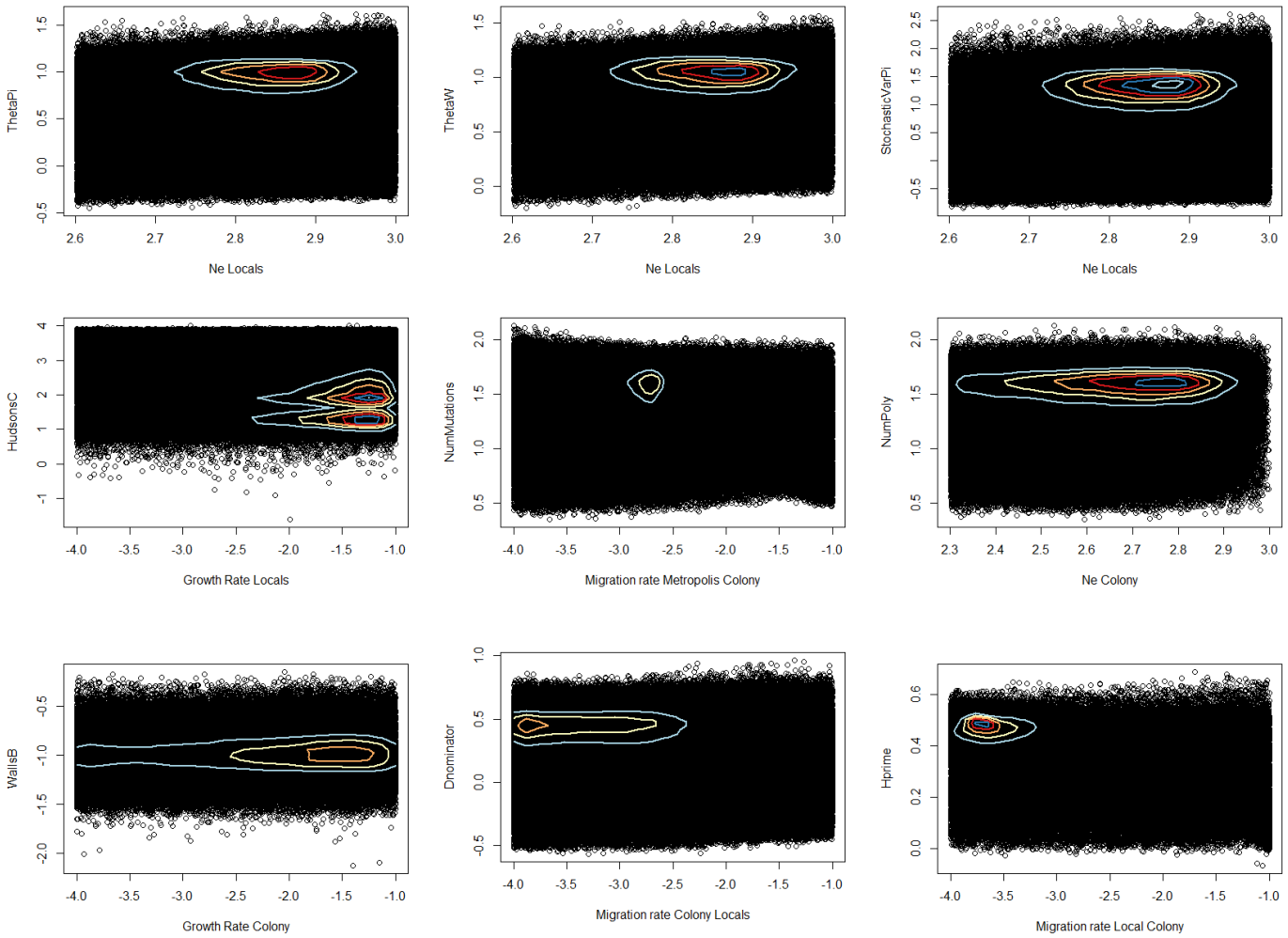


Figure 4.6: Combination of Prior and Posterior Distributions of one Parameter and one Summary Statistic. The lines correspond to the distribution of the prediction for that parameter based on that summary statistic

4.3 Model Selection Results

As we have shown earlier, that assigning to one of the two scenarios of colonization is not an easy task. Here we present the results of our classification efforts through the various tools tested.

4.3.1 ABC - RF

Our first classifier is the ABC-RF classifier. We first tested for the optimal number of trees to be use be steadily increasing the number between 100 and 10000, using an inner error rate of ABC-RF. We saw a tiny but continuous decrease in the error rate until around the 5000 trees mark, where the prediction rate started to fluctuate. We chose 5000 as the optimal number of trees as between 100 and 10000 it yielded the best internal errors and used it in our outer error testing in which we run a 1000 predictions using the leave one out validation. This gave us an average of 0.260 error rate or a 0.74 success rate of assignment.

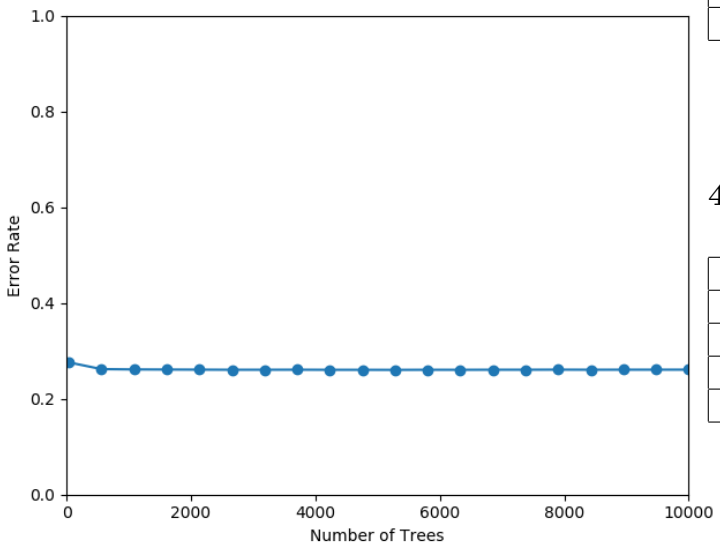


Figure 4.7: Error rates for different number of trees, for the ABC-RF function. As we can see there is little difference between each iteration after the first 1000 trees.

4.3.2 SKlearn RF

Next is the SK-learn version of RF. Here of all the combinations of th hyperparameters, the best one chosen is depicted bellow. This gave us an inner success rate of 0.778 and a final outer success rate of 0.761 .

Hyper-Parameters List	
Hyper-Parameter Name	Best Options Selected
Criterion	'gini' / 'entropy'
Number of tress	100/150/500/1000
Max number of features considered each split	'sqrt' / 'log2'
Depth of trees	20/30/40/50/'None'

4.3.3 SKlearn SVM

The SVM classifier is next. Again of all the combinations of hyperparameters the best one chosen was : 0.68. This gave us an inner success rate of and an outer success rate of 0.674.

Hyper-Parameters List	
Hyper-Parameter Name	Best Option Selected
SVC C	1/2/5/7/10
Kernel	'rbf' / 'poly'
Max number of features considered	20/30/50
Degree for Poly kernel	1 / 3

4.3.4 Best Results By Classifier

Best Error Rates per Classifier	
Classifier	Best Parameters Error
SK learn SMV	0.674
ABC RF	0.74
SK learn RF	0.761

Chapter 5

Discussion

In this final chapter we will discuss the efforts and the results of the current thesis. First we will mention again some of the difficulties encountered and mistakes made during both the design and the execution of this work. Then we will comment on the results presented here and the effectiveness of the process and finally discuss future modifications to the approach and its possible applications.

5.0.1 Discussing the Design process

First, lets take a more critical look on our design of the simulation and summary statistics extraction process. We will talk about the choice and effectiveness of our summary statistics a bit later.

In our design of this process we did not consider a very important parameter: time. After waiting for around 30.000 simulations and doing some initial ABC test, we came to the realization that to achieve a sufficient number of simulations for the ABC to be effective, we would require an unacceptable amount of time. The simulation process by itself was running at an efficient manner. The problem with the initial version of the simulations/summary statistics generation, was the use of 'outside' software.

These software are: plink, convertf and qp3Pop. By outside we are referring to software not created by ourselves or someone in the lab, thus not in our control to modify, in order improve performance. The problem was actually two fold: both the speed of the software itself and the conversions between formats that the software required, consumed a lot of time. Plink for example, while theoretically able to handle vcf files, which Msprime could output , required some modifications on the vcf file, such as the labels of the SNPs and their chromosomes. On the other hand qp3Pop required an 'eigenstrat' format file which had to be converted from the vcf file. It would also seem that qp3Pop was not designed to handle a large number of small SNP files but rather one file with the whole genome.

This left us with CoMuStats, which however was created by Dr. Pavlidi's laboratory and thus we could modify to serve our purpose. This was not an easy process but we successfully manage to incorporate the F2 and F3 calculations into CoMuStats.

We chose to leave the PCA out of the summary statistics along with its clustering metrics, as it would require a great effort for uncertain effectiveness. This however leaves the question of how effective PCA represents these aprameters, or how useful it would be for parameter inference. These questions are open for further exploration and perhaps coupled with / compared to, newer dimensionality reduction techniques such as tSNE and uMAP.

By discarding the outside software and conversions and sticking with CoMuStats, we were able to reduce the time for a cycle of simulation and statistics extraction by about ten times, from an average of almost a minute to around 6 seconds.

By checking the output of simulations we were able to confirm that for almost all parameters we indeed obtained a uniform distribution between the desired ranges. The two exceptions to this were the initial and final size of the Colony population. This is due to the difference in the process that these two parameters were selected at random before each simulation. The initial size of the colony was also selected from a random uniform distribution. However that values was then checked to be less than that of the effective population size of the parental population. Thus in simulation runs where, for example the effective population size of the Metropolis was selected at random to be 500 and the randomly selected initial size of the colony was 700, the size of the colony would be recalculated until it was less than 700. This was done to serve a form of realism, as we thought it would be wrong to allow the colonizing population to be larger than that of its origin population. This is the reason the distribution for that parameter is tilted to the left, or in more formal terms is a 'right tailed' one. Since the final population is dependant on the initial population it also suffers from the same effect. We realise however that this could be causing problems in our results in inference and classification and suggest that should someone try a similar approach, not constrain the prior parameters the same way we did.

Finally as a side note, the range of our parameters is not set in stone. We chose the ranges based on the needs of this thesis and set them enough to give us room for the predictions. Should on follow a similar approach we would suggest increasing the range of parameters to compensate for more diverse scenarios. Something like that of course, would also require a larger number of simulations as well, which is another reason we chose to stick to these ranges.

5.0.2 Discussing the Results

Leaving the data generation behind, here we will discuss the results generated by our ABC methods and the SK learn classifiers. Generally speaking, the results of both the parameter inference and the classifications showed that our selection of summary statistics was not off-point. Even in the difficult case of predicting one of the two scenarios we had some positive results. As we mentioned briefly in the introduction, the particular example we chose to explore, that of a theoretical 2nd phase Greek colonization, would provide unique difficulties.

The fact that the entire scenario is a short one, lasting about 800 years, as well as that reality that the populations are probably closely related were already disconcerting. When compering it to other evolutionary scenarios such as the out of Africa model or when dealing with widely diverged populations we would expect this method to perform much worse with our data. Still, in spite of all these problems, which we translated into the scenario, we still got some positive results. To us, this shows that this method of collecting multiple summary statistics from hundreds of simulations is a valid way of identifying scenarios, yet it requires further improvements.

ABC results

Now lets look at the results from the ABC. We compared our results from the two data sets the Means and the Distribution data set. For 10000 leave one out predictions we recorded the error rate and calculated the RMSE for every parameter.

To our disappointment it appears that the Mean of the summary statistics outperforms the 11-point distribution summary statistics measurements. On every parameter the RMSE error of the Mean data set appears to be almost half of that of the Distribution data set. The average RMSE between all paramters was 0.157 for the Means data set and 0.229 for the Distributions one.

Our aim was to create a more detailed summary statistic representation by providing the distribution of each metric instead of just the means. The Distribution data set contained in the end, eleven-times more summary statistics than the Mean data set, to a total of 957. It is known that sometimes too many statistics can cause noise in the ABC. It would appear that this is also the case here, by increasing the amount of summary statistics we have reduced their effectiveness.

For someone wishing to imitate this methodology of using the distribution, we would suggest a) removing the statistics that have very low variance or b) calculating the correlation between each summary statistic and every parameter and removing the lowest ones. We believe that although in our examples the Mean data set performed better there is still room for improvement of the Distribution data set.

Having said that the author of this thesis is fairly pleased with the results from the ABC, as it seems that among the simulations the parameter can be somewhat estimated, even if not precisely. We would recommend further use of ABC in population genetics and suggest one to try this method we real world data at hand. We highly suggest CoMuStats as a summary statistics generator from ms like simulations, but one may wish to use a different set of summary statistics. For someone wishing to replicate this method we provide a link to a script which transforms a vcf file to an ms format and runs its through CoMuStats (requires CoMuStats to be installed).

Classification results

Finally let's take a look a the classification results. These efforts were made with the single goal of predicting the origin of the colony population parameter As we have mentioned multiple times in this thesis, this was from the beginning considered to be the hardest parameter to predict. The simple reason being that multiple different scenarios could produce the same results in summary statistics. One can easily see this by looking at the figures of 34. Even with some more advanced dimensionality reduction techniques rarely do scenarios 0 and 1 not overlap.

Nevertheless we attempted to use a number of classification techniques coupled with feature and model selections to accomplish this task. The best results produced by each classifier can be seen bellow. As you can see the best performing one was the the SK learn Random Forests. We are not sure why this particular classifier performed the best but we do know its has not to do with the features as the it went through the same pipeline as the SVM one. We aslo noticed that the other Random Forest classifier (ABC-RF) with which it shares the same algorithm also performed pretty well. Ths SVM classifier performed substencially worse than the RF ones. This could be due to the fact that the samples do not separate in any plane, as we saw that even with different dimensionality reduction techniques we couldn't separate them or show any substantial form of structure.

It seems that our chosen summary statistics do indeed contain information that can be used to decipher the parameter of recent origin of population as with our best classifier we are correct about 3 out of 4 times. Still compared to other types of data this is a poor classification result. With this in mind one can expect for some other classifier to perform even better than our best one. Furthermore it is possible that our summary statistics were not best suited for this particular estimation.

5.0.3 Possible Applications

As one may understand, this thesis was of an experimental and exploratory character. Nowhere in this thesis were real data used for our analysis. We recommend that, would someone want to follow a similar approach, one should try to incorporate data from real populations to see how they perform in the ABC.

One can also download our code from hereand generate his/her own simulations for testing their data. To do this all you have to do is download and install all the prerequisites mentioned here and simply execute this file to generate the summary statistics. The scenario we have created here would ideally be used with a 3 population example where one is a product of either one of the other two populations. The scenario however can be changed by the user to be pretty much anything, given that the user has some experience with Msprime.

5.0.4 Closing Remarks

This thesis was a theoretical exploration into a historical scenario. We attempted to simulate the different possible colonization models under a multitude of different parameters. In the process of this thesis we have demonstrated that it is possible to estimate with a decent accuracy the parameters used to generate each simulation. We have also shown that even in a difficult to predict scenario such as the one we explored we can still reach 75 percent success rate. We believe that all of these methods can be taken a step further, as what we created here were not a clear cut / distinct scenarios but more realistic in between ones. As more people experiment with Machine Learning and population genetics we expect this field to change dramatically, as it has for the past decade.

Bibliography

[1] 411 Named Contributors 4. *Encyclopaedia Britannica, Current Edition*. Benton Foundation, 2010.

[2] A.J.Graham. *Colony and mother city in ancient Greece*. Ares Publications, 1983.

[3] Can Alkan. Whole genome sequencing of turkish genomes reveals functional private alleles and impact of genetic interactions with europe, asia and africa. *BMC Genomics*, 2014.

[4] Paolo Anagnostou. Inter-individual genomic heterogeneity within european population isolates. *PLOS ONE*, 2019.

[5] Anthi Angelopoulou. The burial grounds of ambracia. *Aktens des International kolloquions am Deutscen Archaoligischen Institut Abteilung Athen*, 2009.

[6] Haiqin YangEmail author. Deep learning and its applications to natural language processing. *Springer*, 2019.

[7] Edwards AW. Statistical methods for evolutionary trees. *Genetics*, 2009.

[8] Richard Durbin Aylwyn Scally. Revising the human mutation rate: implications for understanding human evolution. *Nature Reviews Genetics*, 2012.

[9] D Charlesworth B Charlesworth. Population genetics from 1966 to 2016. *Heredity*, 2016.

[10] Mark A. Beaumont. Approximate bayesian computation. *Annual Review of Statistics and Its Application*, 2019.

[11] Doron M. Behar. The genome-wide structure of the jewish people. *Nature*, 2010.

[12] Catarina D Campbell. Estimating the human mutation rate using autozygosity in a founder population. *Nature Reviews Genetics*, 2012.

[13] Chensi Cao. Deep learning and its applications in biomedicine. *Acience Direct*, 2018.

[14] Meredith L. Carpenter. Pulling out the 1 percent: Whole-genome capture for the targeted enrichment of ancient dna sequencing libraries. *The American Journal of Human Genetics*, 2013.

[15] L. L. CAVALLI-SFORZA and A. W. F. EDWAR. Phylogenetic analysis: Models and estimation procedure. *Handbook of Statistical Genetics*, 2004.

[16] L.Luca Cavalli-Sforza. The human genome diversity project: past, present and future. *Nature Reviews Genetics*, 2005.

[17] Francisco C. Ceballos. Runs of homozygosity: windows into population history and trait architecture. *Nature Reviews*, 2018.

[18] Christopher C Chang. Second-generation plink: rising to the challenge of larger and richer datasets. *Science*, 2015.

[19] G.M. Cohen. *The Hellenistic Settlements in Syria, the Red Sea Basin, and North Africa*. UNIVERSITY OF CALIFORNIA PRESS, 1975.

[20] The 1000 Genomes Project Consortium. A global reference for human genetic variation. *Nature*, 2015.

[21] Senne Cornelis. Forensic snp genotyping using nanopore minion sequencing. *Scienctific Reports*, 2017.

[22] Corrina Cortes. Support vector networks. *Machine Learning*, 1995.

[23] Simon Myers Daniel Falush Daniel John Lawson, Garrett Hellenthal. Inference of population structure using dense haplotype data. *PLOS Genetics*, 2012.

[24] Peter J. Diggle and Richard J. Gratton. Monte carlo methods of inference for implicit statistical models. *Royal Statistical Society*, 1984.

[25] Mendez Dosuna. *A History of Ancient Greek. From Beginnings to Late antiquity - Chepter: Doric dialects*. Cambridge University Press, 2007.

[26] Petros Drineas. Genetic history of the population of crete. *Anals of Human Genetics*, 2008.

[27] Mohd Fareed. Review single nucleotide polymorphism in genome-wide association of human population: A tool for broad spectrum service. *Egyptian Journal of Medical Human Genetics*, 2013.

[28] Ronald Fisher. *The Genetical Theory of Natural Selection*. Oxford, 1930.

[29] Qiaomei Fu. The genetic history of ice age europe. *Nature*, 2016.

[30] Willard HF. Ginsburg GS. Genomic and personalized medicine: foundations and applications. *Science Direct*, 2009.

[31] Richard E. Green. A draft sequence of the neandertal genome. *Science*, 2010.

[32] Wolfgang Haak. Massive migration from the steppe is a source for indo-european languages in europe. *Nature*, 2015.

[33] Jan Christian Habel. Population genetics revisited – towards a multidisciplinary research field. *Biological Journal of the Linnean Society*, 2015.

[34] Nicholas Geoffrey Lemprière Hammond. *A History of Greece to 322 B.C.* Clarendon Press, 1986.

[35] M.G. Hansen and T.H. Nielsen. *An inventory Archaic and Classical Poleis Centre for the Danish National Research Foundation*. Oxford University Press, 2004.

[36] D.L. Hartl and G.C Clark. He number of alleles that can be maintained in a finite population. *Genetics*, 1964.

[37] D.L. Hartl and G.C Clark. Principles of population genetics. *Sinauer Associates, Sunderland*, 1997.

[38] Philip W. Hedrick. Inbreeding depression in conservation biology. *Annual Reviews*, 2000.

[39] Russell Higuchi. Dna sequences from the quagga, an extinct member of the horse family. *Nature*, 1984.

[40] Simon Hornblower and Antony Spawforth. *The Oxford Classical Dictionary*. Oxford University Press, 2005.

[41] Richard R. Hudson. Generating samples under a wright–fisher neutral model of genetic variation. *Oxford Academics*, 2002.

[42] Morozova I. Toward high-resolution population genomics using archaeological samples. *DNA Research*, 2016.

[43] N. M. Haldane J. B. S. Haldane, A. D. Sprunt. Reduplication in mice. *Journal of Genetics*, 1915.

[44] Thomas H. Jukes Jack Lester King. Non-darwinian evolution. *Science*, 1969.

[45] P. Joyce and P. Marjoram. Approximately sufficient statistics and bayesian computation. *Statistical Applications in Genetics and Molecular Biology*, 2008.

[46] Donald Kagan. Politics and policy in corinth. *Dissetation from Ohio State University*, 1958.

[47] Jerome Kelleher. Efficient coalescent simulation and genealogical analysis for large sample sizes. *PLOS Computational Biology*, 2016.

[48] Motoo Kimura. *The Genetical Theory of Natural Selection*. Oxford, 1968.

[49] Rollins LA. Mitochondrial dna offers unique insights into invasion history of the common starling. *Molecular Ecology*, 2011.

[50] Geoffrey Hinton Laurens van der Maaten. Visualizing data using t-sne. *Journal of Machine Learning Research 1*, 2008.

[51] Iosif Lazaridis. Ancient human genomes suggest three ancestral populations for present-day europeans. *Nature*, 2015.

[52] Iosif Lazaridis. Genetic origins of the minoans and mycenaeans. *Nature*, 2017.

[53] James Melville Leland McInnes, John Healy. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 2019.

[54] Michela Leonardi. Evolutionary patterns and processes: Lessons from ancient dna. *Systematic Biology*, 2017.

[55] Hubby JL. Lewontin RC. A molecular approach to the study of genic heterozygosity in natural populations. ii. amount of variation and degree of heterozygosity in natural populations of drosophila pseudoobscura. *Genetics*, 1966.

[56] J.Z Li. Worldwide human relationships inferred from genome-wide patterns of variation. *Nature*, 2008.

[57] Wentian Li. Application of t-sne to human genetic data. *Journal of Bioinformatics and Computational Biology*, 2017.

[58] Yu Li. Deep learning in bioinformatics: introduction, application, and perspective in big data era. *Science Direct*, 2019.

[59] Ashot Margaryan. Population genomics of the viking world. *Preprint*, 2019.

[60] Matthias Meyer. A high-coverage genome sequence from an archaic denisovan individual. *Science*, 2018.

[61] Vagheesh M.Narasimhan. Estimating the human mutation rate from autozygous segments reveals population differences in human mutational processes. *Nature Communications*, 2017.

[62] M.Nordborg. Coalescent theory. *Handbook of Statistical Genetics*, 2004.

[63] J. Víctor Moreno-Mayar. Early human dispersals within the americas. *Science*, 2018.

[64] Paul Muir. The real cost of sequencing: scaling computation to keep pace with data generation. *BMC*, 2016.

[65] Magus Nordborg. *Coalescent Theory*. Department of Genetics, Lund University, 2004.

[66] P. Pavlidis. msabc: a modification of hudson's ms to facilitate multi-locus abc analysis. *Molecular Ecology Resources*, 2010.

[67] Nikolaos Alachiotis Pavlos Pavlidis. A survey of methods and tools to detect recent and strong positive selection. *Journal of Biological Research-Thessaloniki 2017*, 2017.

[68] Karl Pearson. On lines and planes closest fit to system of points in space. *Uknown*, 1901.

[69] Trevor J. Pemberton. Genomic patterns of homozygosity in worldwide human populations. *The American Journal of Human Genetics*, 2012.

[70] Jean-Michel Marin Pierre Pudlo. Reliable abc model choice via random forests. *Bioinformatics*, 2016.

[71] Cosimo Posth. Language continuity despite population replacement in remote oceania. *Nature Ecology and Evolution*, 2018.

[72] J.K. Pritchard. Population growth of human y chromosomes: a study of y chromosome microsatellites. *Oxford Academic*, 1999.

[73] Kay Prüfer. The complete genome sequence of a neanderthal from the altai mountains. *Nature*, 2013.

[74] A. Raveane. Population structure of modern-day italians reveals patterns of ancient and archaic ancestries in southern europe. *Science Advances*, 2019.

[75] Wrigth S. The genetical theory of natural selection. *Journal of heredity*, 1930.

[76] P. Pavlidis S. Papadantonakis, P. Poirazi. Comus: simulating coalescent histories and polymorphic data from multiple species. *Molecular Ecology Resources*, 2016.

[77] Pontus Skoglund and Iain Mathieson. Ancient genomics of modern humans: The first decade. *Annual Reviews*, 2018.

[78] George Stamatoyannopoulos. Genetics of the peloponnesean populations and the theory of extinction of the medieval peloponnesean greeks. *Journal of Human Genetics*, 2017.

[79] Laurent Stefan. Statistical inference of complex demographic models in drosophila melanogaster and two wild tomato species. *Dissertation, LMU Munich*, 2011.

[80] Mikael Sunnåker. Approximate bayesian computation. *PLOS Computational Biology*, 2013.

[81] the 1000 Genomes Project Consortium 2016. A global reference for human genetic variation. *Science*, 2016.

[82] Dimitra Tsagari. *Europe of Greece, colonies and coins from Alphabanks collection.* Alpha Bank, 2015.

[83] Eleni Vasileiou. Pottery production and sedentism at the end of bronze age. the case of epirus. *ARCHAEOLOGICAL RESOURCES AND FUND UTILIZATION*, 2017.

[84] Roseina Woods. Ancient dna of the extinct jamaican monkey xenothrix reveals extreme insular change within a morphologically conservative radiation. *PNAS*, 2018.

[85] SEWALL WRIGHT. The genetical structure of populations. *Annals of Eugenics*, 1949.

[86] Stephen W.Scherer. Structural variation of chromosomes in autism spectrum disorder. *ScienceDirect*, 2008.

[87] Kae Yanase. Functional snps of the breast cancer resistance protein-therapeutic effects and inhibitor development. *Science*, 2006.