UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

BIOENGINEERING

NONPARAMETRIC APPROACHES FOR POPULATION STRUCTURE ANALYSIS USING CIPs WILD POTATO GERMPLASM COLLECTION

THESIS

Thesis for the Professional Engineering Degree in Bioengineering

AUTHOR

Tamara Fátima Ortiz Ruiz (ORCID: 0000-0003-4381-2431)

ADVISOR

Paul Cárdenas Lizana (ORCID: 0000-0001-7814-2293)

Lima – Peru

2023

DECLARACIÓN JURADA

Yo, Mónica Cecilia Santa María Fuster identificada con DNI No 18226712 en mi condición de autoridad responsable de validar la autenticidad de los trabajos de investigación y tesis de la UNIVERSIDAD DE INGENIERIA Y TECNOLOGIA, DECLARO BAJO JURAMENTO:

Que la tesis denominada "NONPARAMETRIC APPROACHES FOR POPULATION STRUCTURE ANALYSIS USING CIPs WILD POTATO GERMPLASM COLLECTION" ha sido elaborada por la señorita Tamara Fátima Ortiz Ruiz, con la asesoría de Paul Antonio Cárdenas Lizana, identificado con el DNI N°40803000, y que se presenta para obtener el grado de Bioingeniero, ha sido sometida a los mecanismos de control y sanciones anti plagio previstos en la normativa interna de la universidad, encontrándose un porcentaje de similitud de 0%.

En fe de lo cual firmo la presente.

Dra. Mónica Santa María Fuster
Directora de Investigación

En Barranco, el 30 de abril de 2024

Dedication:

This research is wholeheartedly dedicated to my parents, Oscar and Rosi, and my sister, Mariana, for their endless encouragement and support throughout this process. To all my friends and loved ones who have been a constant source of inspiration and strength.

Acknowledgements:

I would like to acknowledge and express my gratitude to the following people, without whom this research would not have been possible:

The genebank team at CIP, especially my advisor, Julian Soto, and all associate researchers who guided me along the way.

My teachers at UTEC, and my advisor, Paul Cárdenas, for providing me with the tools and resources I needed to carry out this project.

My friends and family, who supported me and motivated me all throughout this process.

TABLE OF CONTENTS

Page

TABLE INDEX

Page

FIGURE INDEX

ANNEX INDEX

ABBREVIATION INDEX

AMOVA	Analysis of molecular variance
ANOVA	Analysis of variance
AFLP	Amplified fragment length polymorphism
AM	Association mapping
ARTC	Andean roots and tubers
ASD	Allele sharing distance
BIC	Bayesian information criterion
CGIAR	Consultative Group for International Agricultural Research.
CIP	International Potato Center
DA	Discriminant analysis
DAPC	Discriminant analysis of principal components
DBMC	Density-based mean clustering
FIGS	Focused identification of germplasm strategy
GD	Genetic diversity
GGP	Genome-wide genotyping
GRIN	Germplasm Resource Information Network
GWAS	Genome wide association studies
HPC	High-performance computing
HWE	Hardy-Weinberg equilibrium
ipPCA	Iterative pruning PCA
ITPGRFA	International Treaty on Plant Genetic Resources for Food and Agriculture
KL	Kullback-Leiber

LCA	Latent class analysis
LLBO	Log-marginal likelihood lower bound
LD	Linkage disequilibrium
LE	Linkage equilibrium
LS	Least squares
MAF	Minor allele frequency
MAS	Marker assisted selection
MCMC	Markov Chain Monte Carlo
ML	Maximum likelihood
MP	Maximum parsimony
NGS	Next generation sequencing
NJ	Neighbor joining
NMF	Nonnegative matrix factorization
PCA	Principal component analysis
PIC	Polymorphic information content
PVX	Potato virus X
PVY	Potato virus Y
RAM	Random access memory
SNP	Single nucleotide polymorphism
SMTA	Standard Material Transfer Agreement
sNMF	sparse nonnegative matrix factorization
STR	Simple tandem repeats
SVD	Singular value decomposition
VB	Variational Bayesian

RESUMEN

ENFOQUES NO PARAMÉTRICOS PARA EL ANÁLISIS DE ESTRUCTURA POBLACIONAL DE LA COLECCIÓN DE GERMOPLASMA DE PAPA SILVESTRE DEL CIP

Las especies silvestres de papa poseen genes importantes relacionados a resistencia a enfermedades, tolerancia a estrés abiótico, y otras características de interés agrónomo; sin embargo, continúan siendo las menos exploradas. Esta investigación buscó desarrollar una metodología de análisis accesible y replicable en R para evaluar la diversidad genética y estructura poblacional de la colección de papas silvestres del Centro Internacional de la Papa (CIP) a través de enfoques no paramétricos. Se trabajó con datos de polimorfismo de nucleótido único (SNP) de 1248 accesiones de papa silvestre, de las cuales la mayoría no habían sido genotipadas previamente. Los parámetros de diversidad genética se calcularon antes del análisis de estructura. La estructura poblacional se analizó vía métodos paramétricos, como inferencia variacional Bayesiana, y métodos no paramétricos, como técnicas basadas en reducción de dimensionalidad y distancia genética. El análisis de distancias genéticas reveló agrupaciones según nivel de ploidía, clado taxonómico, y región de origen. Los resultados de estructura poblacional de los distintos métodos revelaron flujo génico significativo entre subpoblaciones, y confirmaron similitudes en la identidad genética de individuos de regiones geográficas similares y con características taxonómicas asociadas. El análisis se programó de manera que pudiera ser replicado y escalado de acuerdo con los requerimientos del investigador. Los métodos no paramétricos utilizados produjeron resultados comparables a los producidos por métodos paramétricos, demandando menor costo computacional, y estableciéndose como una alternativa práctica y efectiva para estudios de genética poblacional. Los resultados de este estudio dan nuevas perspectivas sobre la diversidad y arquitectura poblacional de la colección de papas silvestres del CIP, permitiéndole a otros investigadores entender las relaciones genéticas inter e intraespecíficas de las especies y ampliar la base genética de la papa. La metodología de análisis en R producida permitirá llevar a cabo estudios de genética poblacional con datos SNP en distintos cultivos de manera más rápida y eficiente, promoviendo su uso en distintos programas de mejoramiento genético.

PALABRAS CLAVES:

Papa silvestre; Diversidad genética; Estructura poblacional; Métodos paramétricos: Métodos no paramétricos; Genética poblacional

ABSTRACT

NONPARAMETRIC APPROACHES FOR POPULATION STRUCTURE ANALYSIS USING CIPs WILD POTATO GERMPLASM COLLECTION

Wild potato species hold important genes related to disease resistance, tolerance to abiotic stress, and other traits of agronomic interest; however, they remain being the least explored. This study aimed to develop an accessible and replicable R analysis workflow to explore the genetic diversity and population structure of the International Potato Center's (CIP) wild potato germplasm collection through nonparametric approaches. We worked with single nucleotide polymorphism (SNP) data from 1248 wild potato accessions, most of which had been genotyped for the first time. Genetic diversity parameters were calculated prior to structure analysis. Population structure was analyzed through parametric methods such as variational Bayesian inference, and nonparametric methods, such as dimensionality-reduction and distance-based techniques. Distance-based analysis revealed clustering based on ploidy level, taxonomic clade, and region of origin. Population structure results from different methods revealed significant gene flow between subpopulations, and confirmed similarities in the genetic makeup of individuals from similar geographical regions and with associated taxonomic characteristics. The analysis was programmed such that it can be replicated and scaled according to the researcher's requirements. Nonparametric methods produced comparable results to those produced through parametric methods, requiring a lower computational cost, and establishing themselves as a practical alternative for population genetics studies. The results of this study provide new insights into the diversity and population architecture of CIPs wild potato collection, allowing researchers to understand the inter and intraspecific genetic relationships between species and broaden the genetic base of potato germplasm. The produced R analysis workflow will allow other crop population genetics studies using SNP data to be carried out in a quicker and more efficient manner, promoting their use in genetic improvement programs.

KEYWORDS:

Wild potato; Genetic diversity; Population structure; Parametric methods; Nonparametric methods; Population genetics

INTRODUCTION

Genebanks, or germplasm banks, are biorepositories that aim to preserve crop biodiversity and ensure its availability for use today and in the future [1]. Global food security is being threatened by climate change, environmental shocks, and rising population numbers [1]. Crop wild relatives are a critical asset for addressing food security needs, given their rich genetic diversity can be used to enhance plant performance. They have been used in numerous crop improvement programs to produce more nutritious and resilient crop varieties, and have contributed greatly to the livelihoods of farming communities [2]. However, in the past few decades we have witnessed a decline in crop biodiversity driven by contemporary food and agricultural systems, more specifically by the destruction of natural ecosystems through intensive crop production, urbanization and land use changes [3]. In this context, genebanks play a fundamental role in safeguarding the diversity of important crops and their wild relatives.

The International Potato Center's (CIP) genebank holds the largest potato, sweet potato, and Andean roots and tubers (ARTC) germplasm collection in the world [1]. Given genebanks are mainly concerned with the long-term preservation of crops, most resources are focused on maintaining the viability and integrity of the accessions, to ensure users can access the material they require. CIP has yet to develop comprehensive strategies for understanding their genetic makeup, population structure and traits [4], [5]. Increased access to bioinformatic and genotyping tools give us the opportunity to utilize "big data to optimize the use of biodiversity in breeding" [1]; however, handling and analyzing these large genetic datasets requires the implementation of novel approaches. More efforts should be focused on genotypic characterization, diversity and population structure studies, and on producing standardized workflows that provide researchers with the tools they need to carry out these type of investigations [6].

Description of the research problem

Despite the large amount of genetic material available, little is known regarding the population genetics of the potato collection, which refers to the study of the genetic identity, genetic diversity, and population structure of a population. Increased access to genotyping technologies allows us to produce larger and larger genetic datasets; however, there are certain difficulties related to the analysis of this big data, such as computational cost, lack of standardized analysis pipelines, and limitations pertaining to the standard parametric methods used in the field. Currently, CIP does not have a standardized analysis pipeline, meaning individual researchers need to develop their code from scratch for each project. The wild relative accessions continue being the least explored, despite their characterization being essential to identifying genes associated to disease resistance, tolerance to abiotic stress, and tuber quality [7], [8], [9]. These are all traits of agronomic interest. Very few studies on genetic diversity and population structure have been done using CIPs germplasm collection, and none have focused on the wild potato varieties [10], [5].

Most difficulties pertaining to the analysis of these large genetic datasets relate to the methods used to analyze population structure. These can be grouped into parametric and nonparametric machine learning approaches [11]. There are several open access programs that allow researchers to utilize parametric or model-based clustering approaches, making them the standard in the crop science field. Nonetheless, these suffer from several drawbacks due to high computational costs, genetic assumptions of the clustering model and sensitivity to sample size [11]. A persistent challenge when utilizing these programs is that they are frequently used without considering whether the underlying models align with the study [12].

For example, STRUCTURE is a widely used population analysis tool in the plant breeding community; however, it can be impractical due to the intensive computational cost and the biological assumptions the clustering model follows, given these are rarely met in nature [13]. This results in limited applicability for studies where samples present polyploidy, overlapping generations, or nonrandom mating [14], or studies that use genetic markers without selective neutrality, without low mutation rates or with linkage disequilibrium (LD) [15]. Nonparametric approaches are seen as an effective alternative to address these drawbacks, given they have the advantage of a more efficient computational cost and no prior model assumptions [11]. Moreover, many nonparametric methods have been found to perform similarly or better to standard model-based clustering methods when applied to real and simulated data [16], [17].

Justification and motivation

As aforementioned, no previous population genetics studies have been carried out using CIPs wild potato germplasm collection. There is insufficient information regarding the genetic identity, diversity, and population structure of these cultivars, which limits their use in research and breeding programs. Characterizing the genetic identity of the accessions through genotyping will increase the efficiency of germplasm conservation, promote the use of genebank data, and allow users to strategically select accessions. It has been reported that misidentification, contamination, admixing and deterioration are common problems in any project that handles large amounts of genetic material [18], [19]. Errors in identification can occur due to human error, such as mishandling, mislabeling and admixing; however, these discrepancies can also be due to unrecognized genetic variants in the stock and accidental crosspollination during seed multiplication, causing undesired geneflow [20]. Defining the genetic identity of accessions by their SNP profiles will allow these errors to be identified, reducing costs related to the long-term maintenance of the collection [5].

On the one hand, genetic diversity studies are the basis for any plant breeding and conservation program [21]. Modern crop breeding relies on genetic diversity to introduce genes or alleles of agronomic value to breeding populations. These studies allow us to characterize genetic diversity in terms of key parameters, and facilitate its use in developing new cultivars and in accelerated breeding approaches [22]. Biodiversity studies allow for the efficient use of genetic information for breeding programs, evaluation of adaptability to different environments, understanding of evolution of different varieties, and furthering current understanding of their nutritional and quality properties [23].

On the other hand, population structure analysis allows for the grouping of individuals into subpopulations based on common gene pools, characteristics, and evolutionary relationships, which then determines their capacity to be improved by genetic selection [24]. These serve as a basis for genome wide association studies (GWAS) and admixture analysis (AM) [11] studies, which allow researchers to identify genetic loci associated with specific traits and serve as the foundation for marker-assisted selection (MAS) in breeding [25]. Furthermore, understanding the population structure of the collection makes germplasm conservation more efficient and can encourage the use of wild species in breeding programs [22]. Regional and geographical clustering within a population also allows for the application of focused identification of germplasm strategies (FIGS) to identify species carrying specific adaptative traits, which could facilitate their dissemination to parts of the world with similar characteristics [26]. Additionally, they allow collection curators to identify if there is genetic or geographical bias within the genebank and to understand if diversity in nature is properly represented, allowing them to then correct this bias.

Moreover, the workflow utilized in this study aims to be scalable and replicable for large datasets or genomic ‘big data’ without the need of a high-performance computing (HPC) environment. The R code is written in a way such that only the initial variables need to be defined. Some of the analysis methods have already been implemented and optimized as part of R packages; however, some had to be implemented from scratch. Having the code already implemented and ready for execution will encourage more of these studies to be carried out, which will further broaden the genetic base of potato crops. Regarding the population structure analysis techniques, several of the nonparametric methods chosen had not been tested previously with such a large amount of data, meaning the obtained results can be used to evaluate their performance when applied to high-dimensional data. The objectives of this project align with CIPs “Biodiversity for the future” program’s mission [1].

Definition of the research problem

The current investigation aims to address the gap in knowledge regarding wild potato species, explore nonparametric alternatives for population structure analysis that tackle the limitations of parametric techniques, and attend to the absence of a standardized analysis pipeline to carry out population genetics studies in CIP. There currently exists a gap in knowledge regarding the genetic diversity and population structure of wild potato species in CIPs genebank. No genetic assays have been carried out with CIPs wild potato collection, meaning there is little information on the genetic identity, diversity, and population structure of these accessions. Moreover, no previous studies have covered such a large number of wild potato accessions. Additionally, novel population structure analysis approaches, such as nonparametric methods, should be implemented for analyzing large and high-dimensional datasets, given the most popular techniques in the crop science field have drawbacks regarding functionality, result quality, and computational cost. The latter makes some of them unusable for high dimensional datasets without an HPC environment, limiting the capacity of researchers to produce knowledge without expensive equipment. Moreover, no analysis pipeline has been defined to carry out population genetics analyses at CIP, meaning researchers need to develop their own code from scratch.

Objectives

The general objective of the investigation is to develop an accessible and replicable analysis pipeline in R to explore the genetic diversity and population structure of CIPs wild potato germplasm collection through nonparametric approaches using SNP data. The specific objectives of the investigation are:

i) Establish the genetic identity of each accession by retaining the most informative SNP data to allow the identification of duplicates and mislabeled samples.

ii) Determine the genetic diversity within the accessions through the estimation of key parameters to facilitate their use in plant breeding programs.

iii) Explore the population structure of the collection through nonparametric and parametric approaches to identify associations between genetic profiles and relevant passport data, such as ploidy, taxonomy, and region of origin.

iv) Compare parametric and nonparametric machine learning methods for population structure analysis in terms of clustering results and computational cost.

v) Design a versatile, scalable, and replicable analysis script in R for crop population genetics studies using SNP data.

This investigation followed a non-experimental descriptive research design, given the independent variables of the data utilized were not manipulated. The objectives are focused on describing quantitative characteristics and associations within the already collected data, following a biostatistics and bioinformatics approach.

To achieve this, single nucleotide polymorphism (SNP) data from 1248 wild potato accessions, which accounts for about 50% of the wild potato collection, was analyzed. This data was preprocessed and filtered to allow the genetic identity of each accession to be defined through the most informative SNP data. Genetic diversity quantitative parameters were calculated through the 'snpReady' R package and a custom script. Parametric and nonparametric methods were used for population structure analysis to obtain membership probabilities of each individual into each inferred subpopulation. The parametric, or model-based clustering, analysis were carried out using fastSTRUCTURE. The nonparametric methods can be divided into dimensionality reduction and distance-based methods. The dimensionality reduction-based approaches used were discriminant analysis of principal components (DAPC), single value decomposition (SVD) with discriminant analysis (DA), and sparse nonnegative matrix factorization (sNMF) with alternating nonnegative least squares (ANLS) optimization. These were compared in terms of the optimal number of subpopulations identified, cluster characteristics and the computational cost as execution time. The distance-based approaches involved calculating the Nei distance matrix of the accessions and constructing a bootstrapped dendrogram through neighbor-joining (NJ) and Unweighted Pair Group Method with Arithmetic Mean (UPGMA) clustering algorithms. The analyses were mainly carried out in R and were programmed such that only the initial variables need to be defined for the rest of the code to run, allowing it to be replicated and scaled according to the requirements of the researcher. The entire code can be run in a standard 8GB RAM computer.

Limitations

Due to economic constraints related to sample preparation and sending the samples abroad for genotyping, only 50% of CIPs wild potato collection was included in this investigation. This means we were not able to get an image of the diversity in the entire wild potato collection. However, the selected samples were chosen in a representative manner, meaning the data still represents about 90% of the available species in the collection and their respective regional distribution. Moreover, the analysis workflow produced in this study will be used to analyze the remaining 50% of accessions once they are genotyped, making the entire process much quicker and easier.

CHAPTER I

STATE OF THE ART

This chapter explores existing literature and studies on wild and cultivated potato population genetics, diversity and population studies carried out using CIPs germplasm collection, and the most popular tools and methods used for these types of analyses.

Most potato population genetics studies focus on cultivated varieties or a mixture of cultivated and wild varieties. In 2011, Jacobs et al. carried out a study applying a populations genetics approach to evaluate the taxonomic and systematic relationships among wild potato species, referred to as *Solanum* section *Petota* species [27]. They used amplified fragment length polymorphism (AFLP) data from 566 South American wild potato accessions and the analysis was carried out using STRUCTURE. The results did not allow them to identify clear species and subspecies groups; however, it did clearly support certain taxa and taxa combinations while leaving others unsupported.

In 2015, a study was carried out by Hardigan et al. where they analyzed taxonomy and genetic diversity among wild and cultivated varieties [22]. The panel consisted of 74 wild potato accessions belonging to 25 species, and 213 cultivated potato accessions. The phylogenetic trees generated through SNP-based genetic distances revealed a general agreement with the existing taxonomic groups for *Solanum* section *Petota*. There was greater diversity among the wild and landrace accessions than among the cultivated accessions. Researchers were also able to identify loci with extreme genetic divergence between the wild and cultivated accessions. This research was able to offer a glimpse into potential allele markers to differentiate wild from cultivated species and loci associated to key agronomic traits.

Aside from marker-based genetic diversity analysis, a few diversity investigations focused on genome sequencing have been found. In 2019, Huang et al. analyzed the "full plastid DNA sequence data of 202 wild and cultivated diploid potatoes" in order to explore their phylogenetic relationships and compare to findings from previous studies using marker based data [28]. The study identified the same major taxonomic clades as reported in previous investigations, with differences mostly related to topology. Subclades in clade 4 were linked to geographic characteristics, and whether the accessions belonged to a cultivated species.

In 2022, Tang et al. assembled 44 diploid potato genomes, using 20 accessions from indigenous cultivated diploid groups, 4 accessions from *Solanum candolleanum* and 20 wild potato species [29]. This investigation aimed to characterize the diversity within diploid wild and landrace potato species, explore mechanisms of tuberization, and identify disease resistance genes. The phylogenetic analysis carried out identified complex interspecies relationships, with admixture between *Petota* and *Etuberosum,* existence of incomplete lineage sorting, and frequent gene flow among the wild potato species. Additionally, they found that the potato genome has a larger repertoire of disease-resistance genes in comparison to other seed-propagated solanaceous crops. Although the present study does not focus on gene association, Tang's study offers important insight into the characterization of wild potato species.

Other genome sequencing studies on wild and cultivated potatoes have focused on assessing diversity through SNP identification after sequencing. On one hand is a study by Hardigan et al. in 2017, that sequenced a panel of 67 genotypes, which included 20 wild diploid species, 20 South American landraces, 23 North American cultivars from the *Tuberosum* group. They identified 68.9 million SNPs, suggesting the genetic diversity in potatoes is much greater than in any major crops [30]. On the other hand is a study by Y Li et al. in 2018, that sequenced 201 accessions of wild potato accessions (*Solanum* section *Petota*) and identified 6,487,006 high-quality SNPs [31]. Li's team later questioned the conclusion of Hardigan's study due to their own findings and the estimated of genomic SNPs in other crops such as soybean, pigeon pea, cotton and tomato maxing out at 15 million SNPs [32]. They reanalyzed the data from Hardigan's study and used stricter SNP filtration methods, given they considered Hardigan's filtration procedures to be too relaxed, yielding false SNPs. This reanalysis decreased the number of SNPs obtained to approximately 12 million. For context, popular potato genome-wide genotyping (GGP) array kits interrogate approximately 12,000 SNPs [33]. All things considered, both studies still found increased genetic diversity in wild potatoes when compared to cultivated varieties.

Regarding population genetics studies carried out using CIP germplasm material, in 2018, a study was carried out using 250 cultivated potato accessions from the collection at CIP to evaluate the functionality of the Infinium 12K V2 Potato Array, determine genetic differences between in vitro and original plants, and analyze genetic diversity and population structure of the cultivated taxa through SNP data [10]. The study provided the SNP genetic fingerprint of the accessions, which is now available through CIPs open access database. Phylogenetic analysis showed that the accessions tended to form clusters according to taxa and ploidy level. Additionally, results suggested the triploids included in the study are genetically similar. The population structure analysis through STRUCTURE was able to identify six populations with considerable gene flow between them. This study was more so focused on proving the viability of the array than on the diversity of the accessions they used; however, it was able to display its successful application for solving misidentification errors, facilitate understanding of genetic diversity, relatedness, and population structure.

In 2020, another study was carried out using the CIP's germplasm collection, although this time sweet potato *I. batatas* accessions were used, covering 45% of the species' collection. This study aimed to evaluate the genetic identity and diversity of the accessions using SSR markers, evaluate the phylogenetic relationships and population structure, identify duplicates and mismatches, and compare the collection with a small group of accessions from the United States Department of Agriculture (USDA) genebanks [5]. Phylogenetic analysis showed redundancy in accessions from Peru and Latin America, which coincided with the similarities in morphological data. The population structure analysis suggested the presence of four ancestral populations with low levels of gene flow between them. The comparison with USDA accessions allowed 65 unique accessions to be identified.

There are many examples of population genetics studies with SNP data for a variety of crops. These usually start with data preprocessing and cleaning, such as eliminating low quality SNPs, loci with ≥ 10% missing data and monomorphic SNPs [9], [9], [34]. For genetic diversity analysis, parameters such as genetic diversity (GD) or expected heterozygosity, observed heterozygosity, allelic distributions, pairwise similarity, polymorphic information content (PIC), inbreeding coefficient and more are estimated from the SNP data [9], [10], [34]. Calculations are carried out in R through packages like 'snpReady' [34], and also through software packages like JMP [10] or Genalex [9].

Most population structure analyses are carried out through model-based clustering approaches such as STRUCTURE, which uses a Bayesian approach to assign individuals to populations based on diploid SNP genotypes, Hardy–Weinberg equilibrium and linkage equilibrium (LE) [9], [10], [35], [36]. This is the most popular method used in the plant breeding community. Additional model-based clustering approaches can be found on **Table 1. 1.**

Software	Description	Reference
STRUCTURE	Bayesian clustering approach applying the Markov Chain Monte Carlo (MCMC) estimation.	Pritchard et al. [14]
ADMIXTURE	Maximum likelihood (ML) estimation	Alexander et al. [37]
BAPS2	Bayesian clustering approach applying parallel MCMC chains	Corander et al. [38]
FRAPPE	ML approach that estimates individual admixture fractions	Tang et al. [39]
LPOP	ML approach based on latent class analysis (LCA)	Purcell et al. [40]
fastSTRUCTURE	Variational Bayesian clustering approach	Raj et al. [41]

Table 1. 1. Parametric approaches for population structure analysis

Due to the high computational cost of parametric approaches and the underlying assumptions the models rely on, nonparametric alternatives are starting to gain popularity. Most of these alternatives rely on dimensionality reduction techniques as the first step, followed by a clustering approach on the reduced data as the second step. PCA is a widely used dimensionality reduction technique where variation among individuals is captured in the eigenvalues and eigenvectors [42]. PCA summarizes overall variability to include both divergence between groups and variation within groups without differentiating. For this reason, approaches such as DAPC, a multivariate method for the clustering of individuals in a population, can be used to preserve the between-group and within-group components of variation [43]. When groups are unknown, K-means clustering is used to identify them from the transformed data. The Bayesian Information Criterion (BIC) can be used to assess the model in terms of the number and nature of clusters [43].

Although PCA is the most common dimensionality reduction technique used, there are other approaches which rely on other data transformation methods. For example, Liu and Zhao [17] proposed a two-step approach in 2006 that consists of SVD as the dimension reduction method, followed by different clustering techniques: K-means, mixture model or density-based mean clustering (DBMC). SVD tends to be more computationally efficient when the data sets are composed of a much greater number of variables compared to the number of individuals. SVD has also been paired with DA to infer structure in large data sets [44], although it has not yet been used in population genetics. Moreover, in 2014, Frichot and François [45] proposed a method based on sNMF and least-squares (LS) optimization, which is available in their 'LEA' R package. Additional dimension reduction-based methods for population structure analysis can be found on **Table 1. 2**, elaborated from Alhusain's 2018 review on nonparametric approaches [11]. Several of these two-step methods have performed similarly or better to STRUCTURE when applied to real and simulated data [17], [43].

Reference	Dimension reduction	Distance Matrix	Clustering	Package/s
Patterson et al. [16]	PCA	-	-	EIGENSOFT, SMARTPCA [46]
Jombart et al. [43]	PCA & DA	-	-	Adegnet [47]
Liu et al. [17]	SVD	Cosine similarity	K-means / Mixture model / DBMC	-
Lee et al. [48, p.]	PCA	-	Spectral clustering (K-means, mixture model)	-
Intarapanich et al. [49]	PCA	Euclidean distance	Fuzzy C-means	TW-ipPCA
Limpiti et al. [42]	PCA	Euclidean distance	Fuzzy C-means	EigenDev-ipPCA
Amornbunchornvej et al. [50]	PCA	Allele sharing distance (ASD)	Ward's clustering	-
François et al. [51]	NMF	-	Least-squares optimization	LEA

Table 1. 2. Nonparametric approaches for population structure analysis
Note. Adapted from [11].

CHAPTER II

THEORETICAL FRAMEWORK

The present chapter covers the most important concepts required for the understanding of this thesis. The first section explores the current situation regarding the importance of potato and the use of wild potato biodiversity in plant breeding programs. The second section covers general aspects of how crop genebanks work and the requirements for their correct functioning. The third section describes the basic concepts behind population genetics, including the type of data utilized and how it is obtained. Genetic diversity and population structure are areas of investigation within population genetics; therefore, the final sections of this chapter focus on these two subareas and their respective bioinformatic and biostatistical approaches. The parametric and nonparametric approaches used for population structure analysis have been divided in separate sections for organization purposes.

2.1. Potato diversity

Potato is the world's third most important crop after rice and wheat. It feeds more than one billion people worldwide and sustains the livelihoods of millions [52]. Its global total crop production exceeds 350 million metric tons per year [53]. There are over 4,000 varieties of potatoes native to Peru, Bolivia, and Ecuador, with different characteristics adapted to the harsh conditions they grow in. Additionally, there are between 100 to 180 known wild potato species belonging to regions spanning across the United Stated and Chile [52]. Considering this, it is an essential crop in terms of agriculture and food security.

Potatoes belong to the genus *Solanum*, with over 1,500 species, making it the largest genus within the *Solanaceae* family [54]. Tuber-bearing *Solanum* species are grouped into *Solanum* section *Petota*, which can then be subdivided into the *Potatoe* and *Estolonifera* subsections [55]. The *Potatoe* subsection includes the common cultivated potato as the species *Solanum tuberosum* L.

2.1.1. Wild potato diversity

Potato wild species belong to *Solanum* section *Petota* [56]. There have been various attempts to produce a consistent system for taxonomic classification. John Hawkes published a taxonomic treatment in 1990, in which he recognized 235 wild potato species [57]; however, this taxonomy was updated to 196 species by Spooner and Hijmans [58] in 2001, and to 107 species by Spooner et al. in 2014 [56]. Some of these classifications include taxonomic series and some of them partition them into clades instead, such as Spooner's 2014 conspectus. CIPs genebank currently uses a classification based on Hawkes [55] and Ochoa's [59] descriptions [60]. Of these, approximately 70% are diploid species (2n = 2x = 24), while the rest are mostly tetraploid (2n = 4x = 48) and hexaploid (2n= 6x = 72) [61]. A variety of biological factors such as interspecific hybridization, auto or allopolyploidy, varied types of sexual reproduction and the fact that most previous taxonomists have used morphology to define species is what has brought about such different taxonomic classifications [56].

Wild potatoes species are distributed along the entire American continent and grow in a wide range of habitats, soil types, weathers and temperatures [62]. Wild potato varieties tend to have greater resistance to extreme climates and to a broader range of diseases and pests than their cultivated counterparts [63]. Additionally, they have different morphological and physiochemical characteristics due to the different conditions they come from [64], [65].

2.1.2. Potato reproduction

Potato can be propagated either sexually through botanical seeds, or asexually through tubers. Species formation during evolution and domestication has relied mainly on sexual propagation [66]. On one hand, diploid potatoes encompass most *Solanum* species and are out-crossing due to gametic self-incompatibility [67]. This "prevents inbreeding and thereby promotes intraspecific genetic variation" [68]. On the other hand, tetraploid potatoes are self-compatible. In this regard, self-fertilization results in severe inbreeding depression, which is defined as the "reduced survival and fertility of offspring of related individuals" and results in a reduction of seedling germination [69]. Asexual reproduction of potatoes, also called vegetative or clonal reproduction, relies on potato tubers [70]. Potatoes are herbaceous plants that accumulate starch in the ends of their underground stems as a nutrient store. These thicken enough to form tubers close to the soil surface and once the leaves and stems die, the tubers detach from their stolons. The tubers have multiple buds from which new plants grow [70].

2.1.3. Potato breeding programs

Potato breeding programs focus on the development of new varieties through "conventional breeding and/or biotechnological approaches" to increase agricultural productivity, quality, and resistance to changing climate conditions [71]. CIP breeders focus specifically on developing "early-maturing, stress-tolerant, and disease-resistant potato varieties with characteristics desired by consumers and processors" [72]. Having broad and dynamic characterized gene pools increases the probability of having certain traits and helps assure future unanticipated demands can be met [68]. In this context, wild potato species have contributed to disease resistance, abiotic stress tolerance, enhanced yield and increased quality in plant breeding programs for over 150 years [4], [73], [74], [75].

2.2. Genebanks

Plant genebanks were created to preserve genetic material and to prevent biodiviersity loses in face of current and future changes in environmental conditions or societal needs [76]. CIPs genebank is one out of eleven genebanks created by the CGIAR, or the Consultative Group for International Agricultural Research. These collections are available as international public goods under the International Treaty on Plant Genetic Resources for Food and Agriculture (ITPGRFA) and all materials are subject to the Standard Material Transfer Agreement (SMTA) for distribution and use [2], [77].

The genebank at CIP keeps clonal and seed collections of potato, sweet potato, and Andean roots and tubers. Most accessions are kept as *in vitro* plants and through cryopreservation. CIP holds the largest *in vitro* plant collection in the world [78]. As of 2022, CIP holds 7,490 potato accessions in total, with 4,193 currently available for distribution. Of those accessions, 2,596 are potato wild relatives [79], [78].

2.2.1. Maintenance

Genebanks conserve the genetic material through highly controlled and standardized procedures, which vary according to the type of preservation. Wild potatoes are managed as botanic seeds, while cultivated varieties are kept both *in vitro* and in field. These are initiated from mother plants and maintained as identical clones. Part of the preservation procedures involve transferring to maintain optimal viability and preventing the introduction of plant pests into the material [77]. This process is called phytosanitation. Current potato germplasm conservation methods using low temperatures and sorbitol as an osmotic agent have allowed the time needed for fresh transferring to be extended from 6 – 8 weeks to two years. *In vitro* accessions are the main type of material used for characterization, genetic identity and distribution to breeders, farmers, and researchers [77].

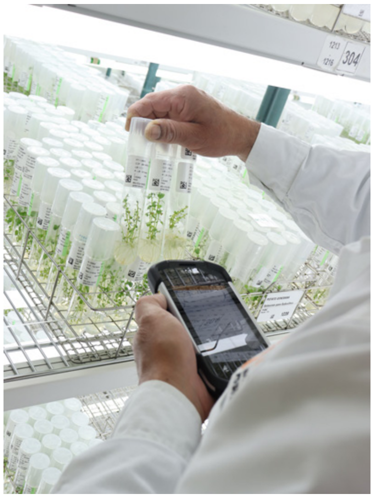


Figure 2. 1. *In vitro* collection [77]

Validating the identity of genetic stocks in germplasm collections is essential to their use in research and plant breeding programs. Internal CIP reports and studies have identified identity errors in the *in vitro* collection when comparing these samples to their field clones. Accessions may not be what they are supposed to be, meaning they carry alleles or mutations that do not match the information in the data base, or that they are a different accession all together. This can be due to mislabeling, unintentional seeding of soil, mix-ups, or accidental crosspollination during seed multiplication [20]. Standard approaches for managing genetic identity in genebank materials aim to reduce identity errors to 1% per generation; however, frequent validation of genetic identity is necessary to avoid the accumulation of errors in the long term [20].

2.2.2. Data management

CIP genebank data is currently available through the Germplasm Resource Information Network (GRIN) and Genesys platforms [1]. The GRIN data base platform is currently being or going to be implemented in more than 20 institutions around the world [80]. Each accession has a unique identifier, institution code, accession number within the specific genebank, collecting number and collecting institute code [81].

CIP is currently working to develop more responsive and user-oriented collection management strategies that allow users to search collections by specific traits, ecogeographical characteristics and more. This would allow the creation of core, mini-core, and composite collections [82], as well as carrying out focused identification germplasm strategy (FIGS) approaches to identify trait-specific genetic resources. The latter uses collection site environmental data from accessions to provide information about which germplasm has the highest probability of carrying certain adaptive traits [83].

2.3. Population genetics and genomics

In its broadest sense, population genetics is a field in biology that studies how allele frequencies at several genes or loci change over time, within and among populations, as a response to evolutionary processes [84], [85]. Common genetic differences within a specific population are referred to as genetic polymorphisms, while genetic differences that accumulate between species form the basis of genetic divergence [86]. In this context, Daniel L. Hartl, renowned researcher in the field, refers to population genetics as “the study of polymorphism and divergence” [86].

2.3.1. Genetic and molecular background

Most genetic and biological principles involved in this field are relatively simple; however, some basic definitions are presented in this section. Genotype refers to the specific set of genes that are present in an individual [86], while phenotype refers to the observable physical characteristics of an individual [87]. Allele refers to alternative forms of a gene at a given position or locus of the DNA molecule [86], [88]. In diploid organisms, or organisms with two complete sets of chromosomes [89], each cell will contain two alleles of each gene at corresponding loci [86]. Alternative forms of an allele tend to be portrayed in uppercase and lowercase letters such as *AA*, *Aa* and *aa*. If the individual inherited identical versions of a gene, the same allele, it is said to be homozygous at that locus, while if the individual inherited different versions of a gene, it is said to be heterozygous at that locus. This brings us to the principle of segregation, which states that each gamete carries only one allele of a gene [86]. This is the essence of Mendelian genetics; nevertheless, it is important to mention polygenic traits do not follow the patterns of Mendelian inheritance [90].

2.3.2. DNA polymorphisms

Modern biotechnological methods and next-generation sequencing (NGS) make genotyping more accessible, facilitating the study of genetic variation at the molecular level [91]. Single nucleotide polymorphisms or SNPs refer to the variation of a single nucleotide base pair at a specific DNA position [86]. A nonsynonymous SNP is a polymorphism that results in an amino acid replacement by altering a codon, this is called amino acid polymorphism. A synonymous SNP is then a polymorphism that does not result in an amino acid replacement given it produces a synonymous codon [86]. Prior to data analysis using SNP arrays, the SNPs that cannot be called or are monomorphic are discarded. Further filtering is done according to SNP call rate, which is "the proportion of genotypes per marker with non-missing data" [11] and minor allele frequency (MAF), which refers to the frequency of the second most common allele in a diploid population [9].

Aside from SNP, there are other types of polymorphisms such as simple tandem repeats (STR), microsatellites and minisatellites. STRs occur when there is a variation in a pattern of nucleotides repeated collectively along the DNA. Microsatellites are STRs in which the repeating unit is 2-9 base pairs long, while minisatellites are STRs in which the repeating unit is 10-60 base pairs long [86].

2.4. Genetic diversity

2.4.1. Hardy-Weinberg principle

The Hardy-Weinberg equilibrium (HWE) is one of the most important concepts in population genetics. It states that "the genetic variation in a population will remain constant from one generation to the next in absence of disturbing factors", this concept is also referred to as Mendelian inheritance [92]. The main assumptions of this model are sexual reproduction, nonoverlapping generations, large populations, equal allele frequencies in the sexes, diploidy, random mating, no migration and no mutation [93]. Given the polyploid nature of potatoes and their complicated reproduction mechanisms, which are further explained in section 2.1.2, many assumptions of HWE are not met. In general, mating patterns in nature rarely exhibit the random mating assumed by HWE [13]. Different types of non-random mating, also referred to as assortative mating, will affect the expected genotype frequencies within a population. In the case of tetraploid potatoes, these are self-fertilizing, which is an example of consanguineous mating [13]. Nonetheless, HWE are still significant in broad terms and commonly used to describe genetic diversity characteristics.

Considering genotypes *AA, Aa* and *aa*, under HWE, the allele frequencies at a single locus are given by [94]:

$$AA{:}\,p^2 \qquad Aa{:}\,2pq \qquad aa{:}\,q^2 \tag{2. 1}$$

$$p+q=1 \qquad p^2+2pq+q^2=1 \tag{2. 2}$$

In this case, p^2, $2pq$ and q^2 represent the frequencies of each genotype in zygotes of any generation, while p and q represent the allele frequencies of A and a in the gametes of the previous generation. In the case that we have a population size of n individuals and the copies of each allele are n_{AA}, n_{Aa}, and n_{aa}, we can estimate the allele frequency p in the population as [86]:

$$p = \frac{2n_{AA} + n_{Aa}}{2n} \quad (2.\ 3)$$

This brings us to the concepts of expected and observed heterozygosity. These can be estimated for a specific locus or individual. For a specific locus, the expected heterozygosity H_e is estimated from the allele frequencies of A and a following the Hardy-Weinberg model, through equation 2.1. The observed heterozygosity H_o is estimated from individual genotypes [95], [96]:

$$H_e = 2pq = 1 - p^2 - q^2 \quad H_o = \frac{n_H}{n} \quad (2.\ 4)$$

In this case, n_H represents the number of heterozygous genotypes at that locus. Heterozygosity values can also be calculated for any individual as [96]:

$$H_{oi} = \frac{n_{Hi}}{m} \quad (2.\ 5)$$

Where n_{Hi} represents the number of heterozygous genotypes in the individual and m represents the number of markers, or sometimes the total SNP calls for that specific individual without including SNPs with no calls [10].

Considering this, the polymorphic information content (PIC) of a marker can be calculated as [96]:

$$PIC = 1 - \left(p_j^2 + q_j^2\right) - (2p_j^2 q_j^2) \quad (2.\ 6)$$

The overall genetic diversity index is estimated as the average expected heterozygosity at n number of loci [97].

$$H_S = 1 - \sum_{i=1}^{n} p_i^2 \quad (2.\ 7)$$

It can be thought of as “the average proportion of heterozygotes per locus in a randomly mating population” or “the expected proportion of heterozygous loci in a randomly chosen individual” [98].

2.5. Population structure

Population structure can be defined as “the organization of genetic variation […] driven by the combined effects of evolutionary processes that include recombination, mutation, genetic drift, demographic history, and natural selection” [99]. Randomly mating or panmictic populations are expected to have similar allele frequencies between groups; nevertheless, as mentioned previously, random-mating patterns are rarely exhibited in nature, which causes population structures to arise [13]. Biological phenomena involved in mating and reproduction will contribute to population structures; however, geographical factors play a big role in the formation of subpopulations [13].

The main goal of population structure analysis is to assign a number of individuals using a number of genetic markers into a number of subpopulations [11]. Consequently, research in this field focuses on how to assign individuals into subpopulations, how to determine the best number of subpopulations, and how to make sure the population structure inferred is an accurate reflection of reality. This type of analysis relies on parametric approaches, such as model-based clustering, and nonparametric approaches, such as dimensionality reduction and distance-based techniques [100].

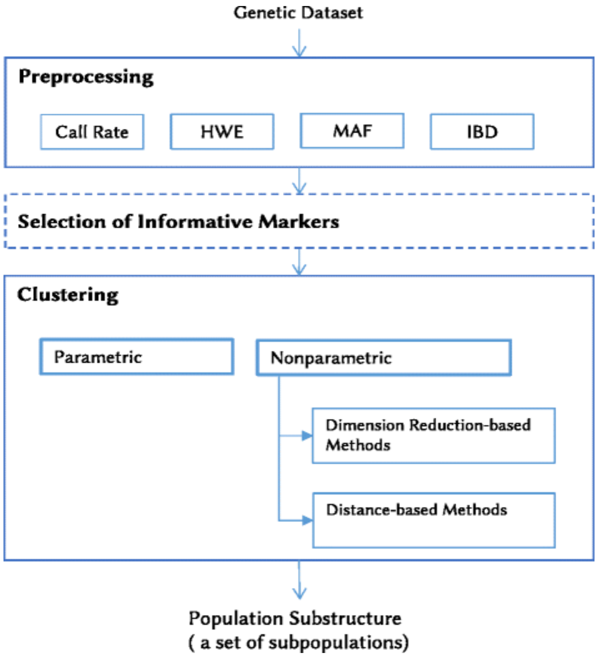


Figure 2. 2. General workflow for population structure analysis [11]

2.5.1. Gene flow

Gene flow refers to the rate of genetic mixing. Low levels of gene flow between subpopulations means they have allele and genotype frequencies that tend to be independent over time [13]. Factors such as geographical separation, ecological adaptation and the accumulation of genetic differences interfere with gene flow and ultimately lead to the distinct organism lineages we have now [101].

Matthew Hamilton, doctor in population genetics and mathematical biology, presents a useful example to explain this concept [13]. If we start with a hypothetical random-mating population divided by a river, such as the one shown in figure 3, genotype frequencies will initially follow HWE and the allele frequencies will be equal on both sides. As the river grows larger, the gene flow between subpopulations on each side will decrease. Over time, the subpopulations will have significantly different allele frequencies due to genetic drift and they will deviate from HWE. The concept of having reduced chances of mating in outcrossing individuals due to increasing distance between them is called isolation by distance [102].

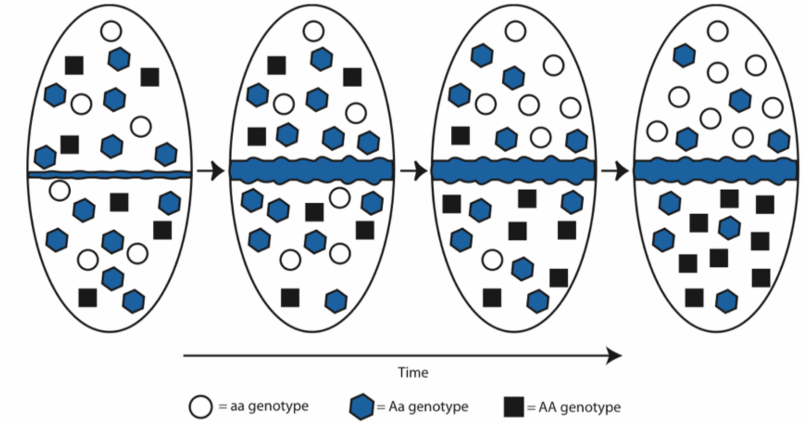


Figure 2. 3. Population structure produced by limited geneflow [13]

2.5.2. Linkage disequilibrium

Linkage disequilibrium (LD) "quantifies the non-random (statistical) association between alleles at distinct loci" [103]. LD between markers and QTL serve as the basis for advanced breeding tools such as marker-assisted selection [103]. Linkage disequilibrium is inferred when alleles are found together more often than expected through independent inheritance [104], [105].

2.6. Parametric approaches for population structure analysis

Parametric approaches for population structure analysis use model-based clustering to assign individuals into subpopulations. Ancestral proportions are inferred for each individual and these are then grouped according to similar patterns of inferred ancestry [42]. Many of the existing parametric methods apply Bayesian inference to model the probability of the observed genotypes according to individual ancestry proportions and allele frequencies [11]. These methods rely on the use of statistical inference models to estimate the allele frequencies in each population. Certain parameters, such as the number of subpopulations, must be set before carrying out the analysis.

2.6.1. Bayesian clustering

In a Bayesian clustering approach, "the partition of items into subsets becomes a parameter of a probability model for the data", which is subject to predetermined assumptions [106]. For the sake of this investigation, we will focus on the clustering method developed by Pritchard, Stephens, and Donnelly in 2000, a Bayesian clustering approach applying the Markov Chain Monte Carlo (MCMC) estimation [14]. This is the framework behind the STRUCTURE program [107]. It uses multilocus genotype data to assign individuals into source populations, while also allowing them to have "proportional assignment of their ancestry to multiple populations" [12]. It is deemed as an admixture model that follows the HWE and LE assumptions. In this context, admixture refers to individuals from two or more previously distinct or isolated populations interbreeding, which results in new genetic lineages [108]. This approach allows prior information about study samples, such as location or traits, to be included in the analysis [15].

This method assumes a model with K number of populations, each characterized by a specific set of allele frequencies at each locus. The user will have to define the optimal number of subpopulations K through procedures such as the Evanno method [109]. The individuals are probabilistically assigned to one or more populations according to admixture levels. Assuming there are N individuals genotyped at L loci, X is the vector of observed genotypes, Z as the unknown populations of origin of the individuals, P as the unknown allele frequencies in the populations and Q as the admixture proportions for each individual, the model with admixture can be described through the following expressions [14]:

$$\left(x_l^{(i,1)}, x_l^{(i,2)}\right) = genotype\ of\ individual\ i\ at\ the\ locus\ l \qquad (2.\ 8)$$

$$z_l^{(i,a)} = population\ of\ origin\ of\ allele\ copy\ x_l^{(i,a)} \qquad (2.\ 9)$$

$$p_{klj} = frequency\ of\ allele\ j\ at\ locus\ l\ in\ population\ k \qquad (2.\ 10)$$

$$q_k^{(i)} = proportion\ of\ individual\ i'sgenome\ that \qquad (2.\ 11)$$

$$originated\ from\ population\ k$$

Where $i = 1, 2, \ldots, N$; $l = 1, 2, \ldots, L$, $k = 1, 2, \ldots, K$, $j = 1, 2, \ldots, J_L$ and J_L is the number of distinct alleles at locus l. Modeling vectors P and Q using the Dirichlet distribution, the probability model is then:

$$\Pr\left(x_l^{(i,a)} = j \middle| Z, P, Q\right) = p_{z_l^{(i,a)}}{}_{lk} \qquad (2.\ 12)$$

$$\Pr\left(z_l^{(i,a)} = k \middle| P, Q\right) = q_k^{(i)} \qquad (2.\ 13)$$

$$p_{kl} \sim Dir(\lambda_1, \lambda_2, \ldots, \lambda_{J1}) \qquad (2.\ 14)$$

$$q^{(i)} \sim Dir(\alpha, \alpha, \ldots, \alpha) \qquad (2.\ 15)$$

The MCMC algorithm with admixture is used to sample from $\Pr(Z, P, Q \mid X)$ through an iterative process. It starts by “randomly assigning individuals to a pre-determined number of groups, then variant frequencies are estimated in each group and individuals re-assigned based on those frequency estimates” [15]. The algorithm comprises many iterations to allow the burn-in process to converge into “reliable allele frequency estimates in each population and membership probabilities of individuals to a population” while accounting for the conditional independence relationships between model parameters and latent variables [15]. The output obtained from the MCMC algorithm can be used to perform inference on Z, P and Q.

2.6.2. Variational Bayesian inference

Given inferring population structure in large genetic datasets presents several computational challenges, alternative clustering algorithms have been developed. In 2014, Raj et al proposed a variation of Pritchard, Stephens and Donelly's method, named fastSTRUCTURE [41]. The variational Bayesian (VB) inference approach is "almost two orders of magnitude faster than STRUCTURE" and is able to achieve population structure results comparable to those obtained through STRUCTURE or ADMIXTURE [41]; however, it still relies on HWE assumptions.

VB inference approaches population structure analysis as an optimization problem. VB works under most of the same parameters defined in the previous section; however, it is described assuming diploid individuals, thus Z is defined as Z^a and Z^b for each copy of the locus. Considering this, instead of sampling from the posterior distributions over Z^a, Z^b, P and Q to compute the moments of each variable, VB inference "approximates the log-marginal likelihood of the data by proposing a family of tractable parametric posterior distributions (variational distribution) over hidden variables in the model" [41]. This can be done using the Kullback-Leiber (KL) divergence as the statistic distance on the probability distribution. For probability distributions $q(x)$ and $p(x)$, KL divergence is defined as [41]:

$$D_{kl}\big(q(x)\|p(x)\big) = \int q(x) log \frac{q(x)}{p(x)} dx \tag{2. 16}$$

In this case, $p(x)$ is the intractable posterior and $q(x)$ is the variational distribution. To simplify the optimization problem, independence is assumed for the latent variables Z^a and Z^b, and parameters P and Q. Maximizing the log-marginal likelihood lower bound (LLBO) can be done by minimizing the KL divergence. LLBO is used as a heuristic to select the most suitable K values for the model [41]. The results of this approach consist of "approximate analytical forms for the posterior distributions over unknown variables" and "an approximate estimate of the intractable marginal likelihood", which can be used to compare different models with different numbers of populations [41].

Such as in the original Pritchard model, a Dirichlet prior is chosen to model the admixture proportions Q. The choice of prior for allele frequencies P will depend on the characteristics of the data and complexity of the structure within. On the one hand, choosing a flat-beta prior allows for higher computational speed but does not perform as well when data has subtle and heavily admixed population structure. On the other hand, choosing a hierarchical prior, referred to as logistic prior given it models population-specific allele frequencies using a logistic normal distribution, takes a much longer time but has shown to return more accurate ancestry estimates when populations are difficult to resolve [41].

2.7. Nonparametric approaches for population structure analysis

Due to increased access to genotyping technologies and the ability to produce larger genetic datasets, nonparametric approaches are increasingly being used for population structure studies, given they have the benefit of requiring less computational time and no modeling assumption requirements [42]. These can be carried out on a standard 8 GB RAM computer. The type of nonparametric methods we will focus on are dimensionality reduction techniques and distance-based methods.

2.7.1. Data storage and handling

HPC environments are commonly used to carry out the analyses of large data, given they are "clusters of powerful processors that work in parallel to process massive multi-dimensional data sets […] and solve complex problems at extremely high speeds" [110]. However, many researchers do not have access to HPC services due to the high cost of hosting an HPC cluster. Depending on the amount of data used for analysis, alternative data representation methods may have to be used to reduce random access memory (RAM) requirements and facilitate data storage and handling with regular computers. The 'adegenet' R package provides us with two object classes created exactly for this purpose: *SNPbin* and *genlight* [111], [112]. These are both S4 formal classes and aim to represent biallelic SNP data as bits instead of integers, making the storage much more compact and allowing operations to be carried out in standard computers. R is only able to handle sets of 8 bits (a byte), but this is handled through "sub-routines in C language" [112]. The efficiency of using these object types for computations is achieved by converting the bit data to numeric data one or two genomes at a time, while optimizing for speed using C language code, using parallel cores, and handling smaller objects [112]. The *SNPbin* object allows for the storage of single genomes while the *genlight* object allows for the storage of *SNPbin* objects for multiple individuals [111].

2.7.2. Dimensionality-reduction based methods

Dimensionality reduction-based methods map high-dimensional genetic data into a low-dimensional space to then perform the clustering method on the reduced dimensions [11].

2.7.2.1. Principal component analysis

PCA is a dimensionality reduction technique used to increase interpretability of large datasets while minimizing information loss [113]. By creating a set of new uncorrelated variables with maximized variance, the analysis is reduced to an eigenvalue and eigenvector problem. PCA is considered an adaptative data analysis technique given the new variables are defined by the data used and not by any previous models or assumptions [113]. It can be defined as an orthogonal linear transformation where the new variables, called principal components (PC), are obtained as linear transformations of the original variables [114]. The values of these new variables are called factor scores. The projection of these factor scores can then be interpreted as the geometrical projection of the observation onto the principal components [114].

PCA starts with a data matrix of p numerical variables, or relative allele frequencies in the case of SNP data, for n individuals. These data values will define the matrix X of dimensions $n \times p$, in which column represents a vector of observations x for allele j. The goal is to find a linear combination of the columns in matrix X such that we can achieve maximum variance [113].

$$\sum_{j=1}^{p} a_j x_j = Xa \qquad (2.\ 17)$$

In this expression, a is a vector of constants and therefore, the variance of any linear combination is $var(Xa) = a'Sa$, where S is a sample covariance matrix associated with the dataset. Linear combinations Xa are the principal components of the dataset.

Considering this, finding the linear combination with maximum variance can be done by obtaining a p-dimensional vector a that maximizes the quadratic form $a'Sa$. Unit-norm vectors are used such that the problem is equivalent to maximizing $a'Sa - \lambda(a'a - 1)$, where λ is a Lagrange multiplier. If we differentiate with respect to vector a, and equate the expression to the corresponding null vector, we obtain $Sa - \lambda a = 0$. This means that vector a must be an eigenvector and λ, the corresponding eigenvalue of covariance matrix S. The eigenvalues represent the variance of the linear combinations. The number of components one wishes to retain can be decided according to the eigenvalues obtained, given these will show the variance captured by each [113]. Different clustering techniques can then be applied on the transformed data matrix, such as will be explained next.

2.7.2.2. Singular value decomposition

Such as PCA, SVD is an eigenvalue dimensionality reduction method that aims to reduce the number of input variables while retaining the most important information. It is a matrix decomposition technique that states any matrix X can be represented as the product of three matrices $U, S\ and\ V^T$ [17].

$$X = USV^T \qquad (2.\ 18)$$

In this case, U and V are column orthonormal with eigenvectors chosen from XX^T and X^TX respectively. S is a diagonal matrix of singular values equal to the root positive eigenvalues of XX^T and X^TX. The eigenvectors of U and V are arranged such that vectors with higher eigenvalues come first [17]. The number of singular values to retain can be chosen according to the explained variance, which can be calculated by squaring the singular values and dividing by the total sum of squares to obtain a percentage. This can be visualized through a scree plot of the principal components. One can then construct a reduced version of the original data $\tilde{X}$ by multiplying the reduced versions of U, S and V such that $\tilde{X} =$

$U_r S_r V_r^T$. This will produce a matrix of the same dimensions of the original data but with a reduced rank. Nonetheless, one can limit the dimensions of this reduced data by multiplying only the reduced versions of U and S, or S and V. This choice will depend on the objectives of the dimensionality reduction [115].

2.7.2.3. Discriminant analysis

DA is a classification multivariate technique used to assign individuals into previously defined groups, based on the variables measured on each sample [116]. It also aims to find the contribution of each variable in the group separation [116]. Considering a data matrix X, DA aims to find a combination of variables such that Xa has maximum variance. The linear combinations Xa are called discriminant functions in DA, and can be found by the eigenanalysis of the D-symmetric matrix [43]:

$$PX(W)^{-1}X^T P^T D \qquad (2.\ 19)$$

In this case, matrices D and P come from the classical ANOVA model, where D represents a diagonal matrix containing uniform weights for the observations and P is defined as a projector onto the dummy vectors of matrix H, which codes the group membership for each observation. W is the matrix of covariances within groups [43]:

$$W = X^T(I-P)^T D(I-P)X \qquad (2.\ 20)$$

Solving this equation requires W to be invertible, which is not the case when the number of variables is greater than the number of individuals. Also, the inverse of W is numeric unstable when the variables are correlated, which is the case with allele frequencies from SNP data. A way to solve this issue is to perform DA on a reduced version of the original data matrix X, using PCA or SVD.

DAPC is a two-step method with PCA as a first step and linear DA as a second step. Instead of performing DA directly on X, DA is performed on the matrix of principal components XU to solve issues related to the number of variables and correlation [43]. DAPC is able to assign individuals to groups, provide a visual assessment of between-population differentiation and the contribution of individual alleles in the population structure [43]. When tested with real and simulated data, it has been shown to perform much faster than STRUCTURE while providing comparable results [43].

Another alternative is using SVD as the data reduction strategy. Elhadji Ille Gado et al. [44] presented a two-step approach using SVD as the first step, and linear DA as the second step. In this case the SVD is used to construct a low-rank approximation of the original data, and then applying DA on this reduced data. This method has not been used for population structure analysis; however, given it was proposed for high dimensionality data, it allows us to solve the issues mentioned previously regarding the number of variables and correlation. It is important to mention that the application of DA requires groups to be previously defined. This can be done through different clustering algorithms, such as K-means clustering. The number of discriminant functions one wishes to retain can then be decided according to the eigenvalues obtained, given these explain the amount of variance.

2.7.2.4. Nonnegative matrix factorization

NMF is a dimensionality reduction technique that aims to provide a low rank matrix approximation for a nonnegative matrix X with dimensions $m \times n$ such that $X \approx WH$. Considering W and H are also composed of nonnegative values, each data point in X can then be explained by an additive linear combination of meaningful components [117]. The most common way to measure the best approximation is by solving the following optimization problem, which uses the Frobenius norm [117]:

$$\min_{W \geq 0,\ H \geq 0} f(W, H) = \frac{1}{2} \|X - WH\|_{F'}^{2} \tag{2. 21}$$

In this case, W is a basis matrix with dimensions $m \times k$, H is a coefficient matrix with dimensions $k \times n$, and $\|.\|_{F'}$ is the Frobenius norm. A problem with some of the algorithms developed for solving this problem is that some do not converge as they should. The alternating nonnegative least squares (ANLS) algorithm has shown to have good performance given it possessed a good convergence property. Several of this framework have been introduced for greater efficiency and speed.

Frichot et al. [51] introduced a least-squares optimization algorithm specialized for estimating ancestry coefficients from genotypic frequency data. The process involves performing sparse NMF on the data matrix X, and then applying their variation of the ANLS algorithm to estimate W and H. This algorithm starts by initializing W entries with nonnegative values and iterating until convergence. Matrix H is then computed such that $LS_1(H)$ is minimized [51]:

$$LS_1(H) = \|X - WH\|_{F}^2, \tag{2. 22}$$

The matrix H is obtained by solving linear regression equations and setting all negative values to zero. Matrix W is then computed such that $LS_2(W)$ is minimized [45].

$$LS_2(W) = \left\| \begin{pmatrix} H^T \\ \sqrt{\alpha} e_{1\times k} \end{pmatrix} W - \begin{pmatrix} X^T \\ 0_{1\times n} \end{pmatrix} \right\|_{F}^2, \tag{2. 23}$$

In this equation $e_{1\times k}$ represents a unit row vector, $0_{1\times n}$ represents a zero vector of length n, and α is a nonnegative regularization parameter. This equation is solved by applying Kim and Park's 2011 [117] block principal pivoting method. Karush-Kuhn-Tucker conditions with a tolerance threshold of 10^{-4} [117] were chosen as the stopping criterion for the iterations.

2.7.2.5. K-means clustering

K-means clustering is a type of unsupervised algorithm used to discover the underlying structure of data distribution by grouping data with similar characteristics together. K-means partitions the data according to its "geometric closeness in the feature space" [118]. The starting point of the algorithm is the initial dataset which we aim to partition into a predefined K number of clusters. The data points are portioned, randomly or by applying a certain heuristic, into K initial clusters. The centroid of each cluster c_k can be calculated as [119]:

$$c_k = \frac{1}{N_{c_k}} \sum_{j=1}^{N_{c_k}} x_{c_k}^{(j)} \tag{2. 24}$$

N_{c_k} refers to the number of points in cluster k and $x_{c_k}^{(j)}$ refers to the points in that cluster. The data is then repartitioned by assigning each data point to the next closest centroid according to their Euclidean distance. The centroids are recalculated for each new cluster and the process is repeated until convergence, which occurs when the data points are no longer regrouped into a different cluster [119]. This method is commonly known as Hartigan-Wong's K-means clustering [120].

2.7.2.6. Inferring the number of clusters

When population groups are not known in advance, the number of clusters or subpopulations must be selected. When using K-means clustering, this can be done by using the BIC as the model selection criterion [121]. Such as DA, K-means clustering also relies on a classical ANOVA model to separate the total variance into between-group and a within-group components. We can then choose the K number of groups which allow us to minimize this within-group variation. This technique has shown to perform better than likelihood methods when predicting the number of subpopulations in genotype data [17], [43], [48].

$$Var(XU) = B(XU) + W(XU) \qquad (2.25)$$

$$B(XU) = tr(U^T X^T P^T DPXU) \qquad (2.26)$$

$$W(XU) = tr(U^T WU) \qquad (2.27)$$

$$BIC = nlog\big(W(X)\big) + glog(n) \qquad (2.28)$$

Equation 2.25 represents the K-means model applied to a PCA matrix XU. The BIC can be used to choose the best clustering model and is described in equation 2.28. In this case W(X) represents the residual variance or variance within groups g. The best runs from the K-means model can be inferred from the lowest BIC value or the inflection point.

Regarding nonnegative matrix factorization, the number of clusters is chosen according to the cross-entropy criterion results. This criterion is based on the prediction error of the ancestry estimation algorithms by comparing the predicted and obtained distribution on a fraction of masked genotypes. Smaller cross-entropy criterion values indicate better outputs and ancestry estimates [51].

2.7.3. Distance-based methods

Distance-based methods are another type of nonparametric analysis approach that involves calculating the genetic distances between individuals and then applying a clustering algorithm [11]. The genetic distance matrix can be calculated in a variety of ways and the clustering results are commonly shown as a dendrogram. Distance-based methods are also commonly used in phylogenetics, which focuses on the evolutionary relationships among species; however, not all genetic data types are fit to explore phylogenetic patterns [122].

2.7.3.1. Nei's genetic distance

Nei's genetic distance [123] can be used to estimate pairwise similarities between individuals or populations. This distance matrix can then be used to construct dendograms or phylogenetic trees. Considering x_i and y_i represent the frequencies of allele i in populations X and Y, the probability of identity of two randomly chosen genes is $j_X = \sum x_i^2$ in population X, and $j_Y = \sum y_i^2$ in population Y. The probability of identity of a gene from each of the populations X and Y is $j_{XY} = \sum x_i y_i$. Therefore, the normalized identity of genes between both populations with respect to a specific locus is [123]:

$$I_j = j_{XY}/\sqrt{j_X j_Y} \tag{2. 29}$$

The normalized identity of genes between both populations considering all loci is then:

$$I = J_{XY}/\sqrt{J_X J_Y} \tag{2. 30}$$

Where J_{XY}, J_X and J_Y represent the means of j_{Xy}, j_X and j_y over all the loci. Considering this, Nei's genetic distance is defined as [123]:

$$D = -\ln I \tag{2. 31}$$

2.7.3.2. Neighbor joining

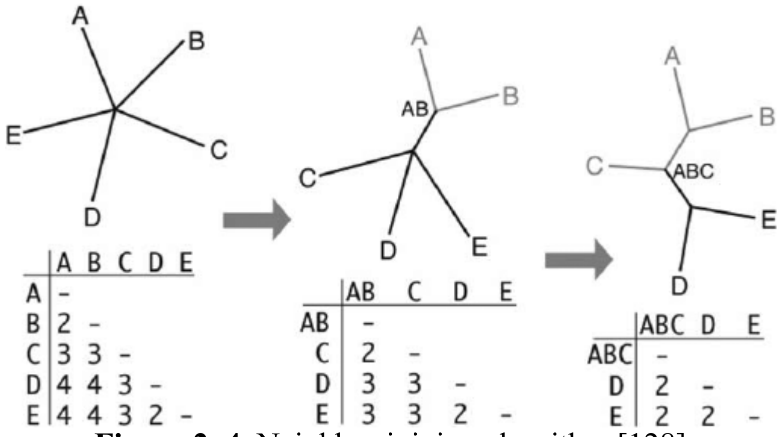
Neighbor joining (NJ) is a clustering method for constructing unrooted dendrograms, created by Saitou and Nei in 1987 [124] and improved by Studies and Kepler in 1988 [125]. It is a clustering method that does not require all lineages to diverge by equal amounts and is suited for datasets that represent species with varying rates of evolution [126]. The idea is to find pairs of operational taxonomic units (OTUs) or neighbors, from a distance matrix, that can minimize the total branch length at each stage of clustering starting with an unresolved tree [124].

It starts with a star-formed tree and "iteratively picks two nodes adjacent to the root and joins them by inserting a new node between the root and the two selected nodes" [127]. A way to do this is by selecting the pair of nodes i and j that minimize a matrix Q, where d_{ij} is the distance between both nodes, R_k is the row sum $\sum_i d_{ik}$ of row k in the distance matrix, and r is the number of nodes remaining [127]:

$$Q_{ij} = (r-2)d_{ij} - (R_i + R_j) \qquad (2.\ 32)$$

When a pair of nodes is joined, it is replaced by a new node A. The distance to the remaining node k is given by:

$$d_{Ak} = (d_{ik} + d_{jk} - d_{ij})/2 \qquad (2.\ 33)$$



	A	B	C	D	E
A	-				
B	2	-			
C	3	3	-		
D	4	4	3	-	
E	4	4	3	2	-

	AB	C	D	E
AB	-			
C	2	-		
D	3	3	-	
E	3	3	2	-

	ABC	D	E
ABC	-		
D	2	-	
E	2	2	-

Figure 2. 4. Neighbor joining algorithm [128]

2.7.3.3. Unweighted pair group method with arithmetic mean

Unweighted pair group method with arithmetic mean (UPGMA) is a clustering method attributed to Sokal and Michener [129]. In contrast to NJ, it assumes a constant rate of evolution. It starts by clustering together two nodes with the smallest genetic distance, i and j, to form a new OTU named A. A new smaller distance matrix is calculated including this new node A. Distances between an individual k and node A is calculated by [130]:

$$d_{Ak} = (d_{ik} + d_{jk})/2 \tag{2. 34}$$

An expression for the unweighted mean between clusters can then be expressed as [130]:

$$d_{(ij)k} = \frac{n_i}{n_i + n_j} d_{ik} + \frac{n_i}{n_i + n_j} d_{jk} \tag{2. 35}$$

Where $d_{(ij)k}$ refers to the distance between clusters ij and k with internal sizes n_i, n_j and n_k.

2.7.3.4. Bootstrapping

Bootstrapping is a method used to obtain confidence limits on phylogenetics by determining the robustness of a model [131]. In population genetics, it is applied by sampling individuals from the data matrix being used and adding replacements to produce bootstrap datasets. Each of these are then analyzed through the chosen clustering method and a consensus tree is constructed according to the results of all replicates [131].

The proportion of replicate trees in which a specific clade is identified is shown as a percentage. This percentage is commonly called the bootstrap value [131]. For example, if 100 bootstrap replicates were done, the bootstrap value would indicate how many times out of the 100 the same branch was observed. Considering this, bootstrap values under 50% are not considered for the final tree construction [132, p. 5].

CHAPTER III

METHODOLOGY

The present investigation can be defined as applied fundamental research given knowledge was generated in the form of data analysis methods and results. The knowledge obtained and workflow followed can then be applied in plant conservation and breeding programs, and other research projects in the area. Moreover, this investigation has a quantitative focus given it relied on the processing and analysis of numerical data, such as allele frequencies, to explore the diversity within and between species, and infer the population structure of the collection. Lastly, this investigation followed a non-experimental descriptive research design, given the data was already collected at a specific instance in time and independent variables were not manipulated. This investigation aimed to evaluate novel and user-friendly bioinformatics and biostatistics methods to describe diversity characteristics of the collection and discover associations between allele frequency data and population structure.

The summary of the methods and techniques followed to address each of the objectives of the investigation are shown in figure 3.1. The data preprocessing step allowed the genetic identity of each accession to be defined through the filtered SNP data. The genetic diversity step allowed the diversity within the whole data set to be analyzed. The population structure step focused on exploring the population structure of the collection, and the results obtained through the different analysis techniques allowed parametric and nonparametric methods to be compared. Altogether, the code used to carry out the analyses constituted an accessible, replicable, and scalable R workflow for crop population genetics. The methodology was mostly carried out in R as an R Project environment through two Rmd files: one for data formatting and preprocessing, and one for the analysis. The libraries and functions used only had to be loaded once due to working in an R project environment. These are available in **ANNEX 2** and **ANNEX 3** respectively. The fastStructure parametric analysis step was the only step not carried out in R, as the program is written for Python.

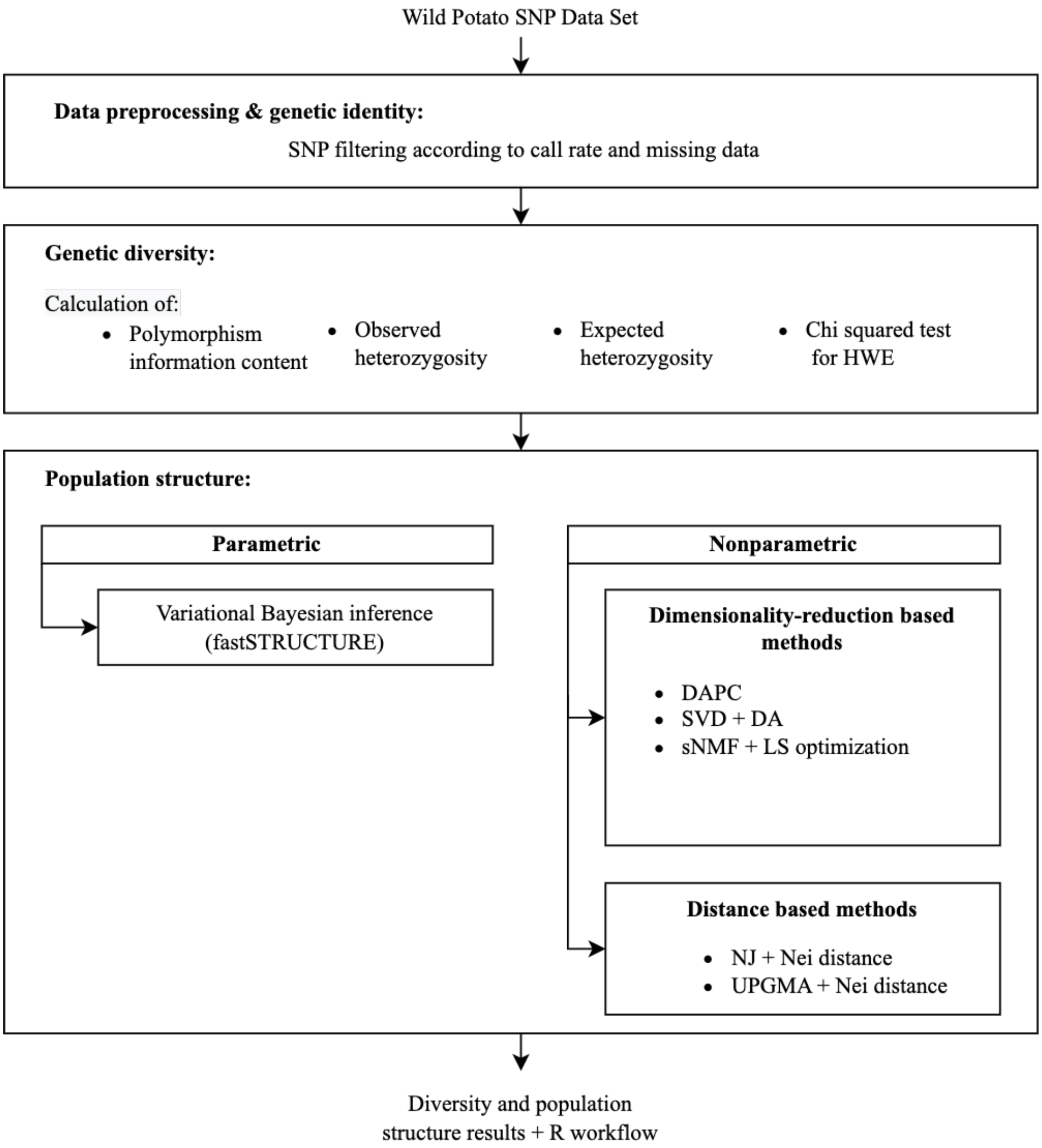


Figure 3. 1. Methodology pipeline

3.1. Dataset characteristics

The dataset utilized was generated by CIP and will be publicly available once the entirety of the collection is genotyped. The samples were genotyped in four batches; therefore, the raw data was composed of four separate datasets. These consisted of SNP data belonging to 1248 wild potato samples and 528 cultivated potato samples from CIPs germplasm collection. The cultivated potato accessions were not used in this investigation given the results would not align with the objectives of this investigation, which focused on wild potato species, and including them would limit the ability to identify subpopulations within the wild potato species. This is due to the homogeneity of their diversity when compared to wild potatoes [22], [28]. For most samples, the material was genotyped in bulk format, using 10 genotypes for each accession. The data covered a total of 31,190 SNP markers [33]. The genotype for each SNP was shown in tetraploid format as AAAA/AAAB/AABB/ABBB/BBBB regardless of ploidy. If the SNP was not able to be called, it was shown as NC. Each potato accession had its corresponding ID, CIP genebank number, species (if known), sample type, type of crossing, date of regeneration, ploidy and georeferencing data. The script used to plot data set characteristics is available in **ANNEX 4.**

3.2. Data preprocessing and genetic identity

Prior to carrying out the data analysis, the data set was filtered to remove low quality data. This allowed the high-quality SNP fingerprint of each accession to be defined. The thresholds chosen to filter the data were a call rate of 0.95 and 0.5 missing data for each sample as suggested by Ellis et al and Diaz et al [10], [35]. This means that only markers with less than 5% of missing data and samples with less than 50% of missing data were used. MAF was not chosen as a filtering criterion given the allele frequencies were extremely varied due to the increased genetic diversity of wild potato species, and the different ploidy levels amongst the accessions [22]. Given this thesis aims to offer a broad understanding of wild potato accessions it was decided to relax the filtering criteria to prevent the loss of important marker information. The script used for data filtering is available in **ANNEX 5.**

To confirm the representativity of the filtered data, a mantel test was performed on the Euclidean distance matrices of the raw and filtered data. To do this, each data set was first converted into matrix form and then to a *genlight* object with the 'adegenet' v2.1.10 package in R [47], [111]. The Euclidean distance matrix for each was calculated through a custom script and the mantel test from these matrices was performed using the *mantel.test* function with 500 repetitions and alpha of 0.05 from the R package 'ade4' v1.7-22 [133]. Once the representativity of the filtered data was confirmed, the data was saved as a csv file for faster import into the analysis code. The script is available in **ANNEX 6.**

3.3. Genetic diversity

The main parameters considered for genetic diversity analysis of the entire data set population were observed heterozygosity, expected heterozygosity, PIC and chi squared test for HWE. For this purpose, the data set was transformed into a matrix and the tetraploid genotype calls were modified into diploid numeric form in the following way: AAAA = 0, BBBB = 2, and AAAB, AABB, ABBB = 1 [134]. Once this was done, the parameters were calculated through the *popgen* function in the R package 'snpReady' v0.9.6 [96]. The data import script is available in **ANNEX 7.** The data frame to matrix conversion script is available in **ANNEX 8**. The genetic diversity analysis script is available in **ANNEX 9.**

Moreover, given the samples were genotyped in bulk format, some were genotyped along with some individuals of the same accession to validate this genotyping method. These were compared through a simple pairwise matrix using the *dist.gene* function from the 'ape' v5.7-1 R package [135] in order to evaluate dissimilarity, which would ideally be below 5%. The script used for this validation procedure is available in **ANNEX 10**.

3.4. Population structure

Population structure analysis was evaluated through parametric and nonparametric approaches. The parametric approach used was variational Bayesian inference through the fastSTRUCTURE Python2.x program [41]. This method was chosen given it's based on Pritchard, Stephens and Donelly's Bayesian clustering model STRUCTURE [14], which is the most common approach used in population structure studies [12], [15]. Due to time constraints, fastSTRUCTURE was chosen for the analysis given it has shown to obtain results compared to the original program while having much greater computational speed [41]. The regular STRUCTURE algorithm requires the entirety of a processor's computing power to complete a single run, meaning complete analyses tend to require several days or weeks [136]. The nonparametric approaches used can be divided into dimensionality reduction-based and distance-based approaches. The dimensionality reduction-based approaches used were DAPC, SVD with DA, and sNMF with ANLS. The distance-based approaches used were UPGMA and NJ clustering from the Nei distance matrix. These dimensionality reduction-based methods were chosen given they have been shown to obtain results comparable to those obtained through STRUCTURE analysis [11], [43], [48], and have been previously used on datasets with similar characteristics as the one used in this study, such as mixed reproduction, overlapping populations, and nonrandom mating [9], [34], [35], [134]. SVD with DA is the only method not previously used in crop population genetics studies. The distance-based analyses were performed prior to the other methods in order to observe preliminary grouping based on ploidy, species and country of origin, and to decide which samples would be used for the remaining structure analyses.

3.4.1. Input data

Most of the population structure analyses were carried out using the tetraploid matrix data. This matrix was constructed from the original filtered data set by modifying the tetraploid calls into numeric form in the following way: AAAA = 0, AAAB =1, AABB = 2, ABBB = 3, BBBB = 4. This data frame to matrix conversion step is available in **ANNEX 8** and the object creation step is available in **ANNEX 11**.

3.4.2. NJ dendrogram

The bootstrapped NJ dendrogram was constructed from the tetraploid allele frequency data using the *aboot* function from the R package ‘poppr’ v2.9.4 [137]. The function was set to NJ and 500 bootstrap repetitions. The distance method was set as a custom function to calculate the Nei distance between individuals according to equations (2. 29), (2. 30) and (2. 31). The resulting tree was exported as a Newick file using the *write.tree* function from the R package ‘ape’ v5.7-1 [135]. The tree was then imported into iTOL for visualization [138]. Tree annotations regarding ploidy, country of origin, species and taxonomic clade were generated using the *create_unit* function from the ‘itol.toolkit’ v1.1.5 R package. The script used is available in **ANNEX 12**.

3.4.3. UPGMA dendrogram

The bootstrapped UPGMA dendrogram was constructed from the tetraploid allele frequency data object using the *aboot* function from the R package ‘poppr’ v2.9.4 [137]. The function was set to UPGMA and 500 bootstrap repetitions. The distance method was set as a custom function to calculate the Nei distance between individuals according to equations (2. 29), (2. 30) and (2. 31). The resulting tree was exported as a Newick file using the *write.tree* function from the R package ‘ape’ v5.7-1 [135]. The tree was then imported into iTOL for visualization [138]. Tree annotations regarding ploidy, country of origin and species were generated using the *create_unit* function from the ‘itol.toolkit’ v1.1.5 R package. The script used is available in **ANNEX 13**.

3.4.4. Secondary data filtration

The dendrograms were evaluated according to the consistency of the obtained clustering and the expected clustering, considering species, ploidy, and region of origin of the individuals. The ‘best’ dendrogram was then used to remove certain individuals from the data matrix used in the structure analysis. Most accessions were genotyped in bulk; however, some were genotyped alongside individuals of the same accession to validate the bulk method genotyping. After confirming these were clustered in the same group, the individual accessions were removed from this filtered data matrix, as to not overrepresent population groups. Moreover, accessions considered to be mislabeled, duplicated accessions and hybrids were not included. All the posterior population structure analyses were carried out with this newly filtered data.

3.4.5. fastSTRUCTURE

The population structure of the data was estimated parametrically using the program fastSTRUCTURE [41]. The installation and execution code for the program was done with guidance from the fastSTRUCTURE GitHub repository [139] and the code is available on **ANNEX 15**. The data had to be converted into a STRUCTURE file format before analysis. This was done by first modifying the tetraploid calls into diploid calls through the method shown in section 3.3, given fastSTRUCTURE only works assuming diploid individuals. Then, the diploid data matrix was converted into a STRUCTURE file format through a custom script based on the *numeric2structure* function from the 'R-Genetics-Conv' GitHub repository [140], available in **ANNEX 3**. The full file conversion script using this function is available in **ANNEX 14.** Using this file as input, the fastSTRUCTURE algorithm was carried out for subpopulation numbers K from 1 to 15, both with logistic and simple priors. The optimal number of subpopulations was established using the *chooseK.py* function from the fastSTRUCTURE program, which reports the model components used to explain structure in the data and the model complexity that maximizes the marginal likelihood [41]. The clusters assignment results were visualized using the *plotQ* function from the POPHELPER v2.3.1 R package [141] and a custom script. The result plotting script is available in **ANNEX 16** and **ANNEX 17.**

3.4.6. DAPC

Population structure analysis through DAPC was carried out using the 'adegenet' v2.1.10 package in R [47], [111]. The tetraploid data matrix was first converted into a *genlight* object using the script in **ANNEX 11** to facilitate RAM requirements and data handling. A PCA was performed on this object through the *glPca* function with centering and mean imputation for missing values [43], using the 'adegenet' v2.1.10 package [47], [111]. K-means clustering was performed up to 20 maximum clusters to identify the optimum number of clusters within the data set population. This was done using the *find.clusters* function on the *genlight* object with its respective PCA. For this part, the *choose.n.clust* parameter was set to FALSE so that the function could run successive K-means with an increasing number of clusters. The number of principal components retained was chosen to ensure 80% of the variance was represented. The best number of subpopulations was defined by the lowest associated BIC or point of inflection [43]. After choosing an optimum range of K subpopulations, K-means clustering was again performed with the *find.clusters* function but now setting the *n.clust* parameter to the desired number of K subpopulations. DA was then performed on the PCA for each K, using the inferred groups from the *find.clusters* results. This was done using the *dapc* function. The number of discriminant functions retained was chosen to ensure 80% of the variance was represented. The results were plotted as a composition plot to observe membership probabilities to each cluster. The full procedure is available in **ANNEX 18**.

3.4.7. SVD with DA

The SVD was carried out on the tetraploid data matrix after centering and mean imputation for missing values using the *svd* function from base R. The number of singular values to retain was chosen in order to maintain 80% of the variance in the data. A reduced data matrix was then computed by multiplying reduced versions of the U and S obtained matrices. K means clustering up to 20 maximum clusters was then performed on this reduced data matrix using the *kmeans* function from the 'stats' R package. The BIC was calculated for each clustering result through a custom function available in **ANNEX 3**, and the best number of subpopulations was defined by the lowest associated BIC or point of inflection [43]. After choosing an optimum range of K subpopulations, DA was performed on the corresponding K clustering results using the *lda* function from the 'MASS' v7.3-60 R package [142]. The posterior membership probabilities were calculated using the *predict* function from the same package. The number of discriminant functions retained was chosen to ensure 80% of the variance was represented. The results were plotted as a composition plot to observe membership probabilities to each cluster. The full procedure is available in **ANNEX 19**.

3.4.8. sNMF with ANLS

The sNMF with ANLS approach was carried out using the 'LEA' v3.12.2 package in R [45]. The data was first converted into a *geno* type file with a modified version of the *write.geno* function to allow for tetraploid formatted data. The population structure was evaluated with the *snmf* function for 1 to 20 K subpopulations. The ploidy was set to tetraploid given this is the format the data is formatted in. The number of clusters was chosen according to the point the cross-entropy curved exhibited a plateau [143]. The script is available in **ANNEX 20**.

3.4.9. Approach comparison

The results obtained through each population structure analysis approach were compared by evaluating the number of subpopulations chosen (optimal K values), the assignment results for a single K value in terms of membership probabilities, the characteristics of the clusters obtained by each method, and the computational cost in terms of execution time. The evaluated cluster characteristics were the number of individuals with probabilities > 0.9 to belong to a single cluster, and the interquartile range, median and mean probability values for each cluster excluding individuals with probability values below 0.01. This comparison was done for a single K value for simplicity purposes. The script is available in **ANNEX 21**.

3.5. Considerations

The present investigation was carried out through a signed collaboration agreement with CIP available in **ANNEX 22**, given they own the intellectual rights over the data used. The raw data will eventually be available through CIPs open access platform on Dataverse [144]; however, until then, nor the raw data or preliminary results are to be shared without prior authorization by CIP. Only the analysis code was included as annexes in this document, given the full project files include data that cannot be shared for the time being. The results of this investigation were shared with CIP and will be used in upcoming publications and additional studies. The respective authorship information will be included.

The raw data analyzed has not been manipulated or distorted in any way, shape, or form. The analysis was carried out directly on the raw data provided by CIP. All data analysis techniques and R packages have been appropriately referenced and linked back to their original creators.

CHAPTER IV

RESULTS AND DISCUSSION

The graphs and tables presented in this chapter are results of own elaboration in response to the objectives of the thesis research.

4.1. Data set characteristics

As mentioned in the previous chapter, the original raw data consisted of four separate data sets with SNP data of 1248 wild potato and 528 cultivated potato samples from CIPs germplasm collection. The cultivated potato samples were not included in this investigation. In all cases the data covered a total of 31,190 SNP markers. Regarding the SNP markers, these were mapped across 12 chromosomes of the potato genome, ranging from 3958 markers on chromosome 01 and 1865 markers on chromosome 10. Moreover, 483 markers were mapped on unanchored scaffolds (chromosome 00) and 28 on chloroplasts. The marker distribution can be seen on **Figure 4. 1**.

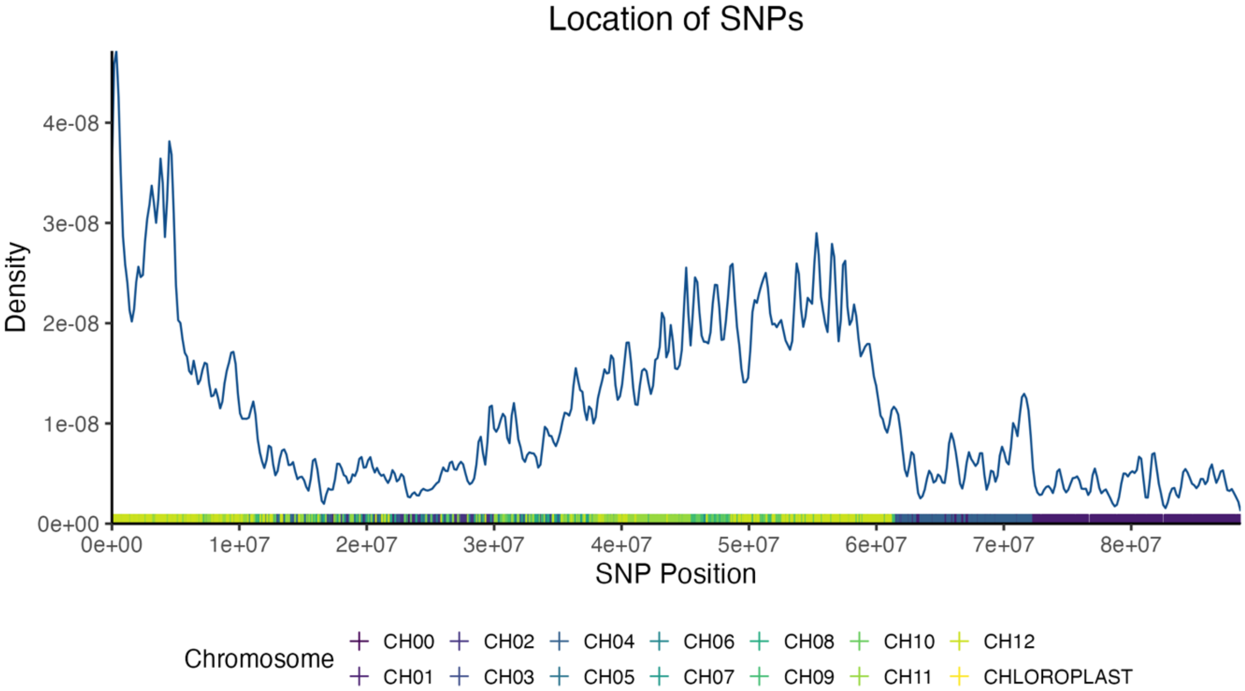


Figure 4. 1. Density plot of the 31,190 SNP marker distribution across the potato genome

Regarding the potato accessions, only the wild potato accessions were included. Despite not all of them having updated taxonomy information, the genotyped samples were chosen to represent 166 species according to CIPs taxonomic classification. Most of the accessions belong to the Peru and Bolivia region. The overall geographical distribution of the accessions can be observed in **Figure 4. 2**. Most of the accessions were genotyped in batch format, using germplasm from 10 different individuals; however, 9 accessions where genotyped in batch along with a few of the individual genotypes from the same accession.

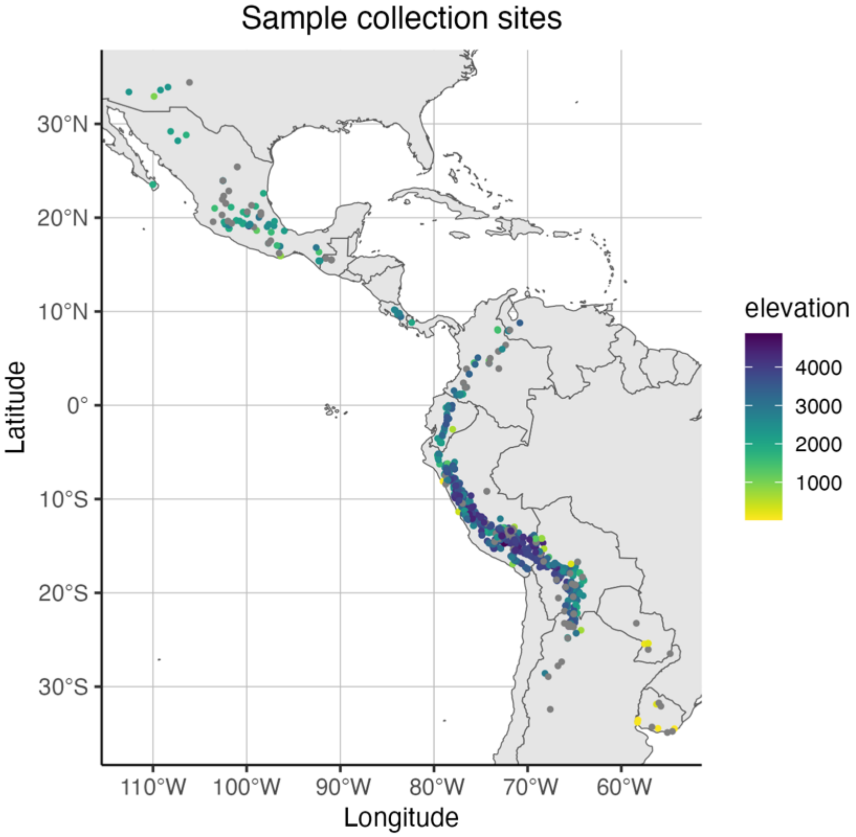


Figure 4. 2. Schematic map of geographical distribution of 1248 wild potato samples

4.2. Data preprocessing and genetic identity

The filtered dataset was composed of 1248 samples and 18,485 SNP markers as the genetic fingerprint of each accession. Filtering according to 0.95 call rate removed 12706 (40.7 %) markers. No accessions were removed due to missing data given they all met the filtering criteria of having less than 50% of missing data. In order to measure the representativity of the filtered data, a Mantel test with 500 replicates was performed on the

Euclidean distance matrices of the raw and filtered data. The test gave us an R value of 0.99 and p-value of 0.002. The obtained R value suggests a strong positive relationship between the two matrices. Furthermore, the obtained p-value allows us to reject the null hypothesis, that the two matrices are unrelated, with an alpha of 0.05. These results confirm that our filtered dataset remains representative of the original data.

4.3. Genetic diversity

The mean PIC value, which denotes the informativeness of each marker, was 0.1267, with values ranging from 0 for monomorphic markers and 0.37. The overall PIC value distribution can be observed in **Figure 4. 3**. Despite accounting for a very small proportion of the markers, it is recommended to remove monomorphic markers as part of the data preprocessing. The mean expected heterozygosity of the SNP markers was 0.1505, while the mean observed heterozygosity was 0.106, which shows a reduction in the expected genetic variability. Moreover, a total of 5028 SNP markers diverged from HWE expectations according to the chi square test results with a p-value of 0.05. These results show a departure from HWE that can be attributed to forces such as inbreeding or gene flow between populations [101], [145].

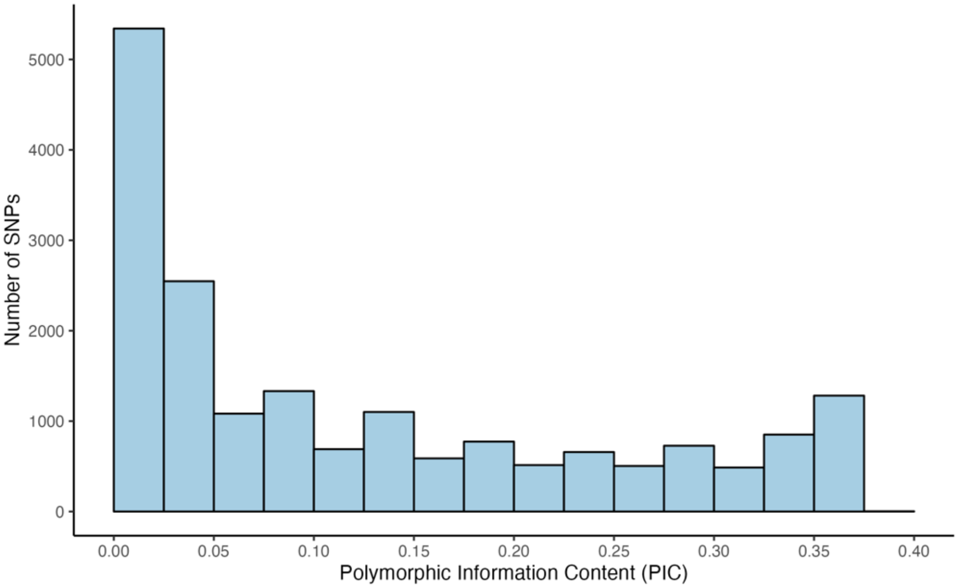


Figure 4. 3. Distribution of PIC values across 18,485 SNP markers

The mean observed heterozygosity of the accessions was 0.110, ranging from 0.01 to 0.33. Observed heterozygosity values by species and ploidy can be observed in **Figure 4. 4** and **Figure 4. 5**. The results do not show any specific pattern between species and observed heterozygosity, or ploidy and observed heterozygosity. We initially expected to find a relationship between these variables such as was found in Ellis et al. 2018 study [10]; however, this discrepancy could potentially be attributed to the bulk genotyping method used for data collection in this investigation, which had not been used previously in CIP.

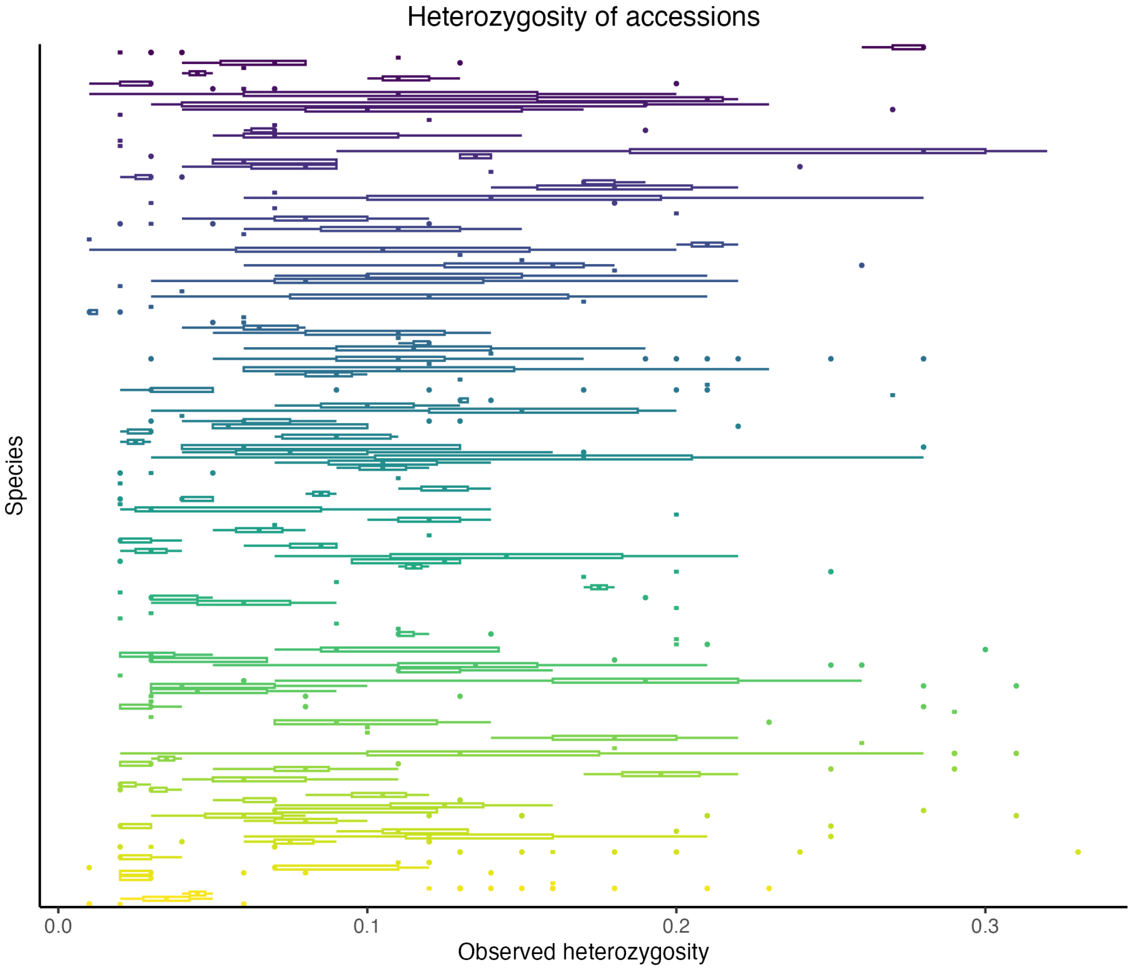


Figure 4. 4. Observed heterozygosity of 1248 samples by recorded species. The y-axis and color represent each recorded species

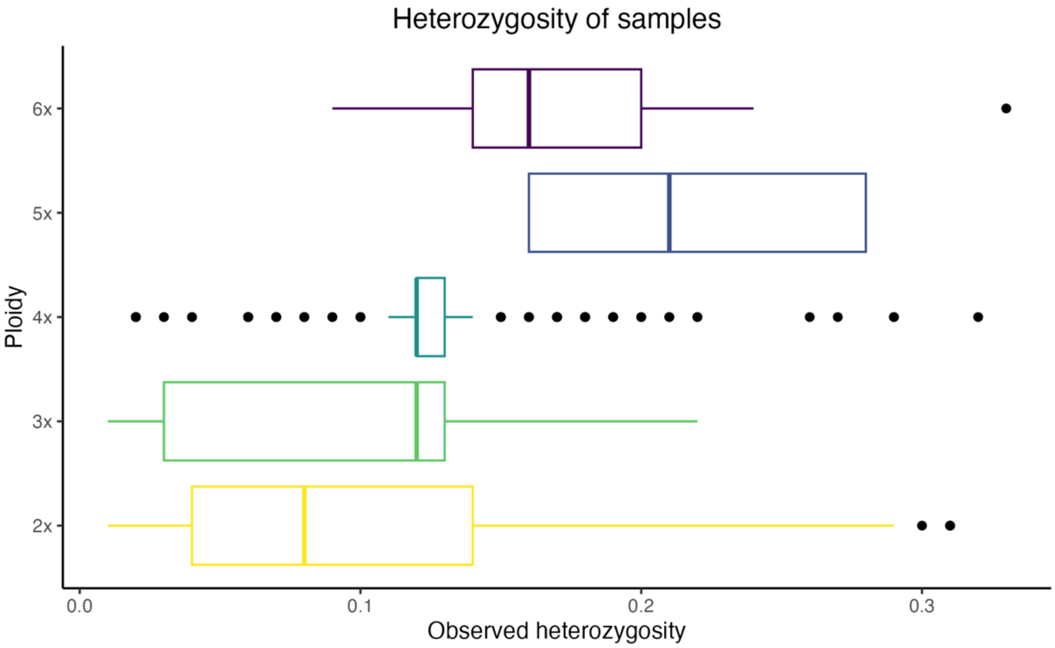


Figure 4. 5. Observed heterozygosity of 1248 samples by ploidy. The y-axis and color represent ploidy level, the black points represent outliers

In addition to calculating the genetic diversity parameters, the bulk genotyping method used was evaluated through pairwise distance matrices to measure dissimilarity between the bulk sample and the individual samples of the same accession. Of those accessions, accessions wp-1_761164, wp-201_760642, wp-9_760212 and wp1269_761143 all had dissimilarity values less than 0.05 when comparing the bulk method data to the individual sample data. Accessions wp-237_765994, wp1281_761156 and wp-120_763923 all had dissimilarity values less than 0.10 when comparing the bulk method data to the individual sample data. Accessions wp-174_762070 and wp-287_761748 showed the greatest dissimilarity values, ranging from 0.07 to 0.32. This suggests either an identity error between the samples, mislabeling or mix-up during data collection [18]. Further validations procedures should be implemented when utilizing this bulk genotyping method to ensure all germplasm material truly belongs to the same accession.

This investigation provides a first insight into the diversity of a large number of wild potato accessions; however, it is important to mention not all obtained results, specifically concerning specific parameter values, are directly comparable to those obtained in other studies. This is due to differences in the markers used for the calculation, differences in the utilized formulas, and differences in the genotyping method. The main difference in this case is the use of the bulk genotyping method, which has rarely been used for potato population genetics studies. It is possible that the diversity results would be different if individual genotyping had been carried out for all accessions. Further research into the impact of batch genotyping on diversity parameters such as heterozygosity is recommended to understand its true impact. Moreover, additional marker filtration criteria could have been implemented, such as filtering monomorphic and dimorphic markers, such that the resulting data matrix was smaller and remaining analyses were quicker.

4.4. Population structure

4.4.1. Dendrograms

Dendrograms were produced out of the 1248 wild potato accessions using NJ and UPGMA clustering as shown in **Figure 4. 6** and **Figure 4. 7**. Clusters with bootstrap values between 50% and 100% were colored from red to green. The accessions were annotated in terms of ploidy and country as can be seen in both figures. The labels were also edited to show the accession ID, the reported species, and the specific collection site.

NJ and UPGMA trees are not usually the preferred clustering methods when carrying out phylogenetic analysis given maximum likelihood or maximum parsimony clustering tend to produce more robust phylogeny results [144], [145]. However, the aim of producing these dendrograms was not to study the evolutionary or phylogenetic relationships between individuals, but to explore general consistency in clustering based on ploidy, country, and species, before posterior population structure analyses on a reduced data set. NJ and UPGMA performed reasonably fast considering the amount of data and bootstrap repetitions used. NJ took 1.220277 minutes per bootstrap repetition, while UPGMA took 1.040829 minutes per bootstrap repetition.

Regarding the obtained results, both trees show a general consistency between ploidies, country, and reported species. The dendrograms were evaluated along with CIPs genebank conservation expert and a total of 198 samples were removed from the dataset for population structure analysis. The samples were eliminated in cases were mixture or mislabeling was suspected according to clustering results. Hybrid and duplicate accessions were also removed. Moreover, the individual samples included as part of the batch genotyping validation were also eliminated. The clustering of these samples also correlated to the pairwise distance results reported in section 4.3. The UPGMA tree seems to have higher bootstrap values overall than the NJ tree; nonetheless, the low bootstrap values in the smaller clusters were to be expected considering the complexity of the data.

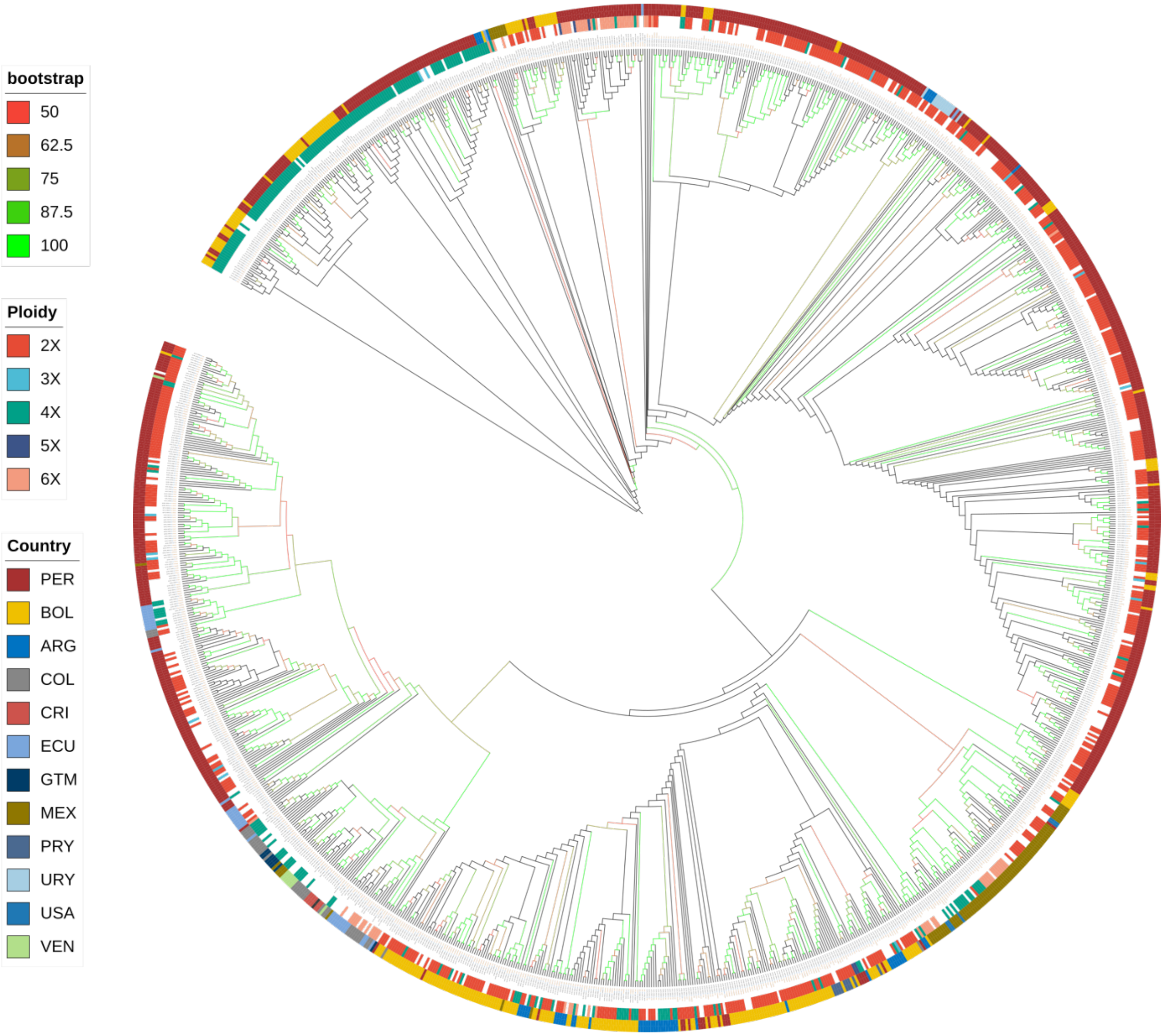


Figure 4. 6. NJ Dendrogram from the 1248 wild potato samples

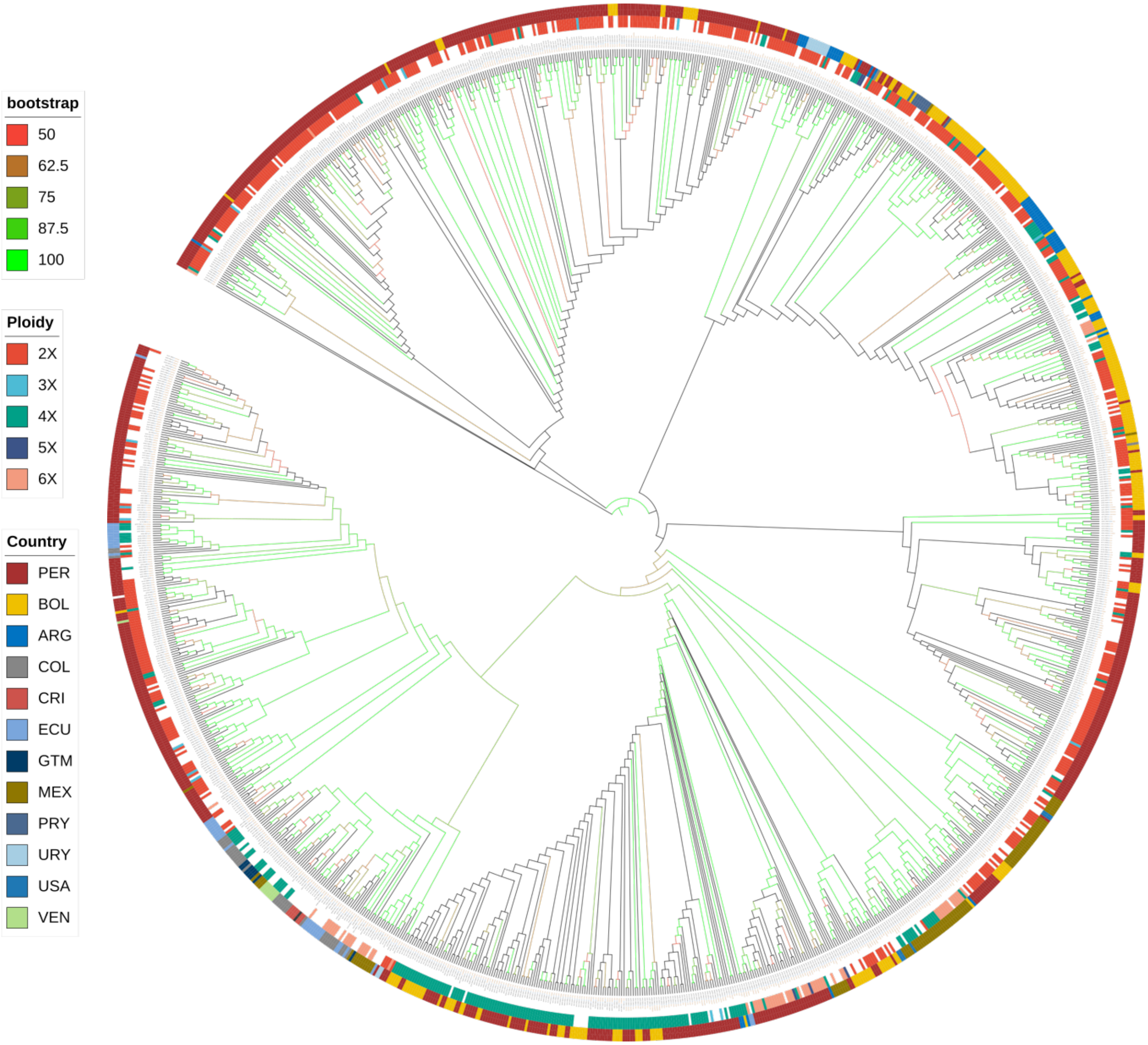


Figure 4. 7. UPGMA dendrogam from the 1248 wild potato samples

After confirming both dendrograms gave similar results, additional characteristics, such as Spooner's taxonomic clades [56], were included as annotations to further explore the structure of the data. We were able to identify a total of 13 large groups considering Spooner's taxonomic clades, ploidy, country, and the obtained clusters. **Figure 4. 8** shows the NJ dendrogram annotated with Spooner's taxonomic clades and the identified large groups. These guided our optimal K subpopulation choice in the population structure analysis.

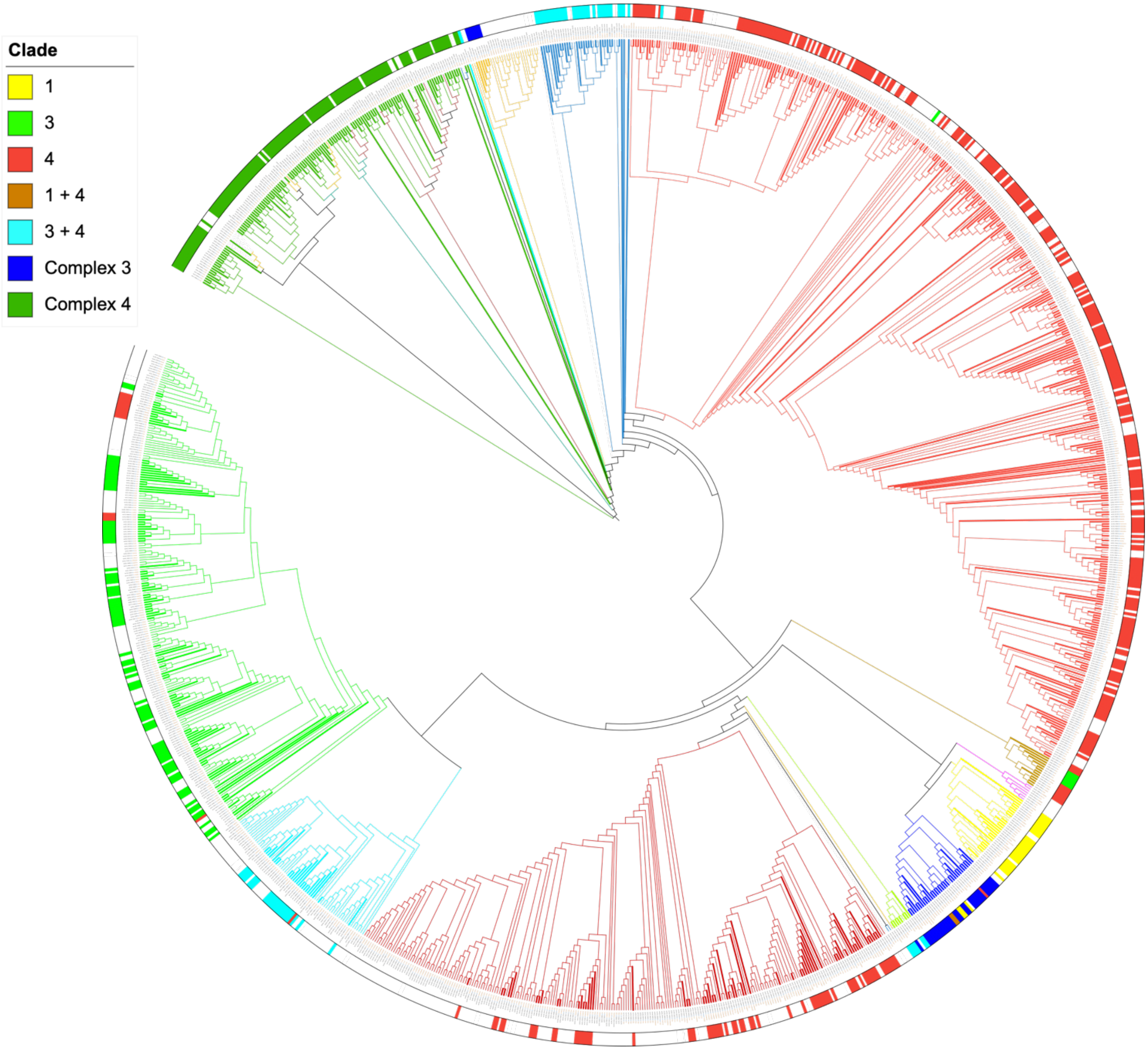


Figure 4. 8. NJ dendrogram with large groups identified based on Spooner's taxonomic clades annotations. Branch colors represent the identified large group based on clade, ploidy, country and cluster, while the external colors represent Spooner's taxonomic clade

4.4.2. fastSTRUCTURE

The model-based clustering was carried out using the Python program fastSTRUCTURE, with simple and logistic priors. When analyzing the simple prior results with the program's *chooseK* function, the model components used to explain structure in the data was 11, while the model complexity that maximizes the marginal likelihood was 14. This suggests the optimal number of K subpopulations is between 11 and 14. This range corresponds to the 12-13 large groups identified in the NJ dendrogram results, which correspond to Spooner's taxonomic clade classification [56]. The assignment plot for those K values can be seen in **Figure 4. 9**. Each bar represents an individual and the fill represents the membership probability for that cluster. The group names in the clustering process is arbitrary; however, it can be seen that there are certain individuals that are consistently grouped together in all K runs.

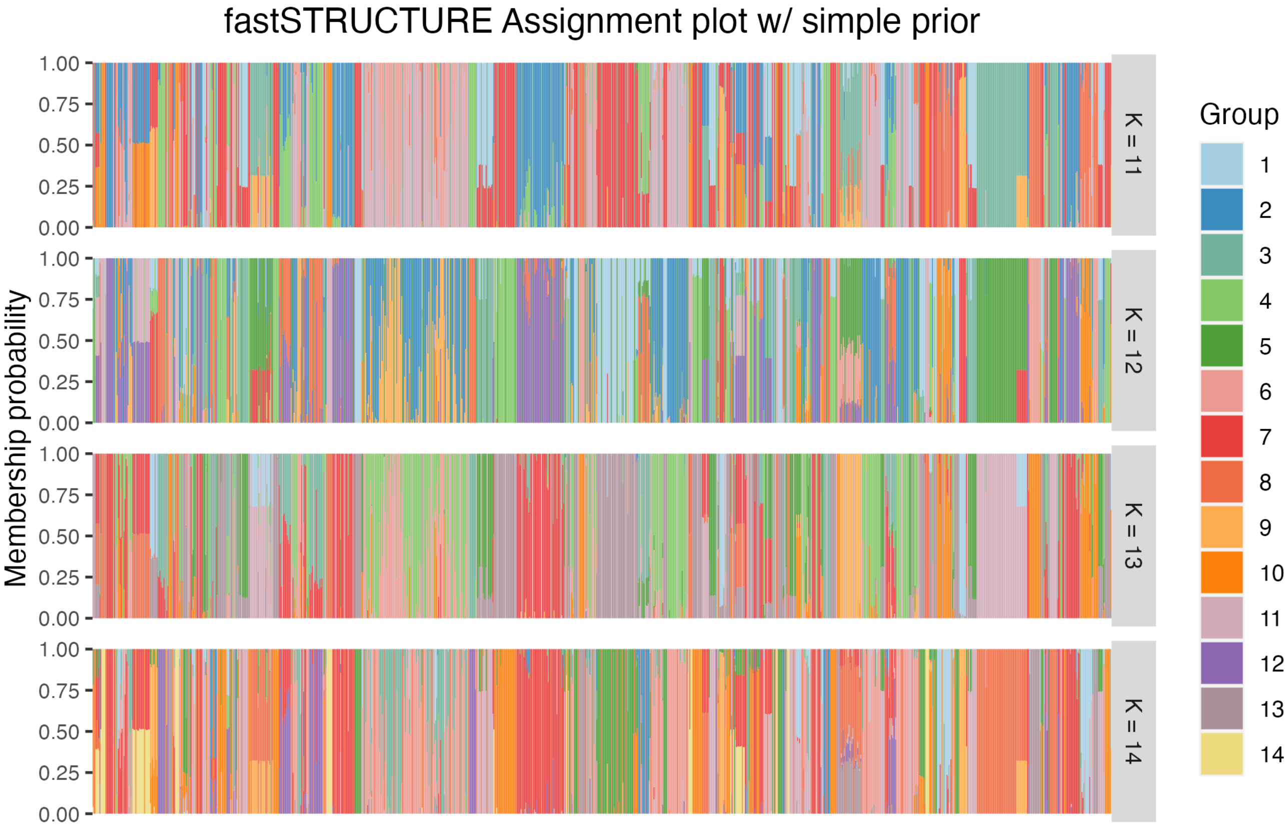


Figure 4. 9. Membership probabilities of individuals using fastSTRUCTURE with a simple prior for K values 11-14

When analyzing the logistic prior results with the program's *chooseK* function, the model components used to explain structure in the data was 1, while the model complexity that maximizes the marginal likelihood was 9. This suggests the optimal number of K subpopulations is between 1 and 9, which is not a particularly useful range given it is too large to be informative. The individual output files for each K run were evaluated and we found the marginal likelihood values for all runs from K = 9 onwards was "NaN". This would explain why the *chooseK* function gave us such a large value range, given all marginal likelihood values for K = 9 to 15 were empty. It seems this is a common problem users experience when using fastSTRUCTURE with a logistic prior [146]. Given the obtained marginal likelihood results were not reliable to give us an optimal K value range, the assignment plot was plotted for K values 11 to 14 such as the simple prior results. This plot can be seen in **Figure 4. 10**. The results are quite similar to those obtained with the simple prior.

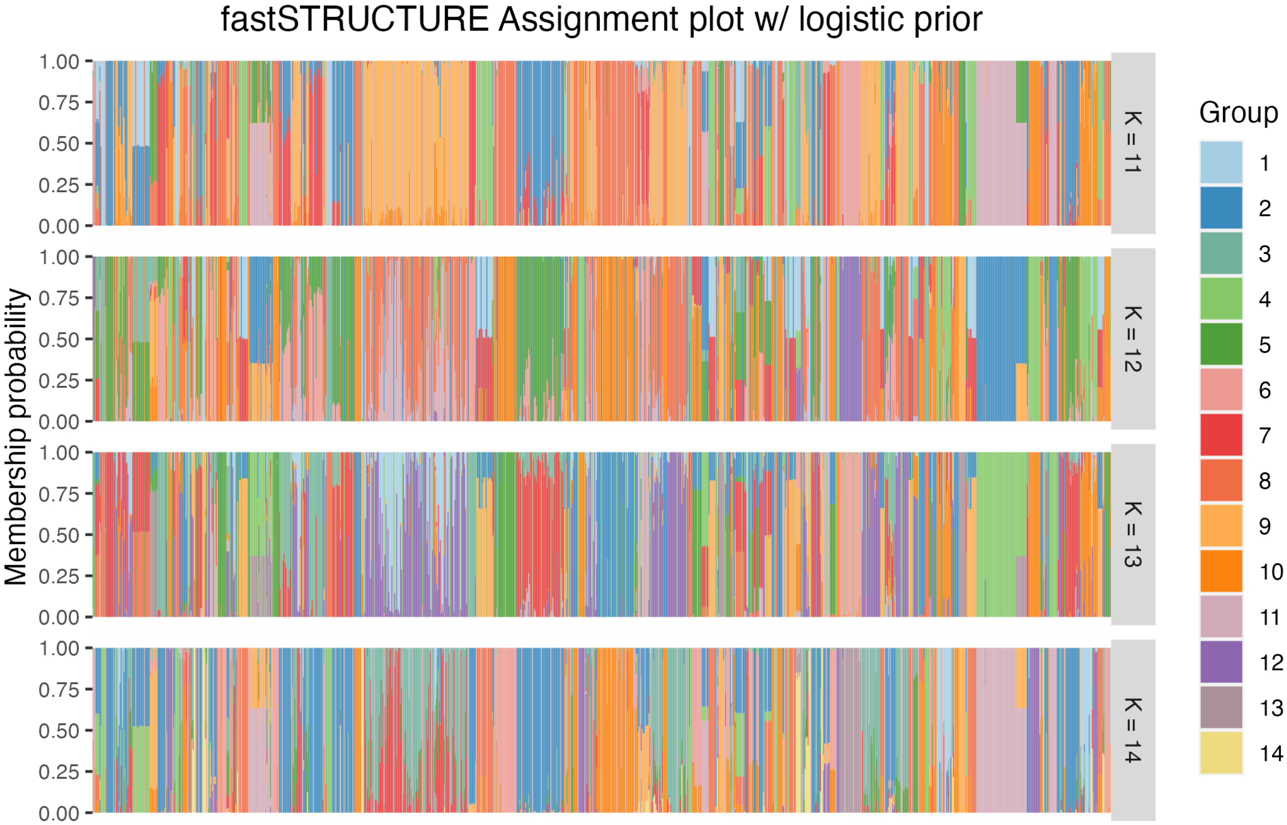


Figure 4. 10. Membership probabilities of individuals using fastSTRUCTURE with a logistic prior for K values 11-14

The individuals with the most consistent clustering across all K runs with both simple and logistic priors belong to the *S. acaule* Bitter (acl) species, which are represented by cluster 3 in K=11, cluster 5 in K=12, cluster 11 in K=13 and cluster 8 in K=14 in **Figure 4. 9**. A total of 126 individuals were assigned to this cluster with membership probabilities of above 0.9. Individuals from *f. incuyo* Ochoa. (inc) are also consistently clustered together into this group, which corresponds to the fact that this is a morphological variant of the acl species [59]. Another group of individuals clustered consistently across all K runs and with both priors are those belonging to *S. sparsipilum* Bitter *Juz. et Buk*. (spl), *S. ugentii* Hawkes & K. Okada (ugt), *S. vidaurrei* Cárdenas (vid), *S. avilesii* Hawkes & Hjerting (avl), *S. boliviense* Dunal (blv*), S. brevicaule* Bitter (brc), *S. oplocense* Hawkes (opl), *S. infundibuliforme* Philippi (ifd), S. xbruecheri Correll (lph) and *S. bill-hookeri* Ochoa ssp. astleyi Hawkes & Hjerting (ast). These are all species from Bolivia, Peru, and Argentina, and correspond to cluster 2 in K=11, cluster 12 in K=12, cluster 7 in K=13 and cluster 7 in K=14 in **Figure 4. 9**. Another group of individuals that were consistently clustered together were those represented by cluster 7 in K=11, cluster 1 in K=12, cluster 13 in K=13 and cluster 5 in K=14.

4.4.3. DAPC

The first step of DAPC was to carry out a PCA on the data. A scree plot was plotted to evaluate the explained variance by each principal component. This can be seen in **Figure 4. 11**. There is no consensus on how to select the best number of interpretable principal components [147]. Therefore, we relied on retaining 80% of the genetic variance, for which we chose the first 50 principal components.

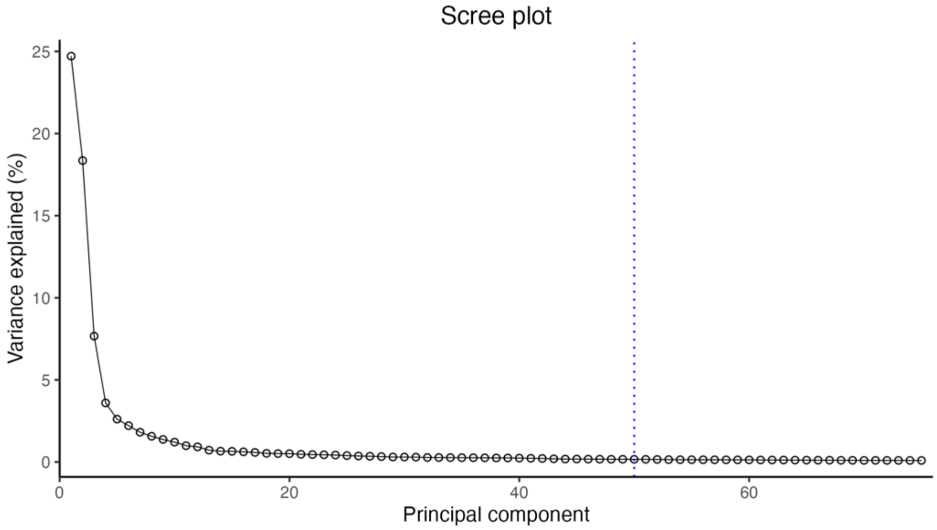


Figure 4. 11. Variance explained by each principal component

The optimum number of clusters was chosen by evaluating BIC values obtained through successive K-means clustering. The BIC plot can be seen in **Figure 4.12.** The optimum number of clusters was chosen at the inflection point of the BIC values, which occurred at K = 12, which remains close to the identified large groups in the NJ dendrogram results. K-means is a rather simple way of measuring group differentiation and might not identify the correct clusters when analyzing highly complex data [17]. It is recommended to evaluate other clustering alternatives [148]; nonetheless, in this case K-means proved efficient for identifying the best number of subpopulations and assigning individuals to the genetic clusters. Moreover, it is consistent with the type of variance partition used in DA [43]. The posterior DA were carried out for an optimum K value range of 11 to 14. To represent 80% of the variation in the original data, 5 linear discriminants were retained for all K values. The variance explained by each linear discriminant for K = 13 can be seen in **Figure 4. 13**.

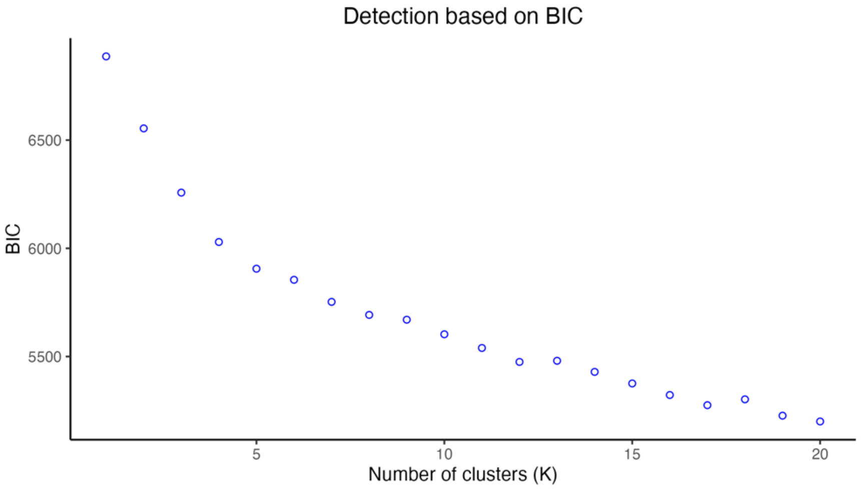


Figure 4. 12. Inference of number of clusters in DAPC according to BIC

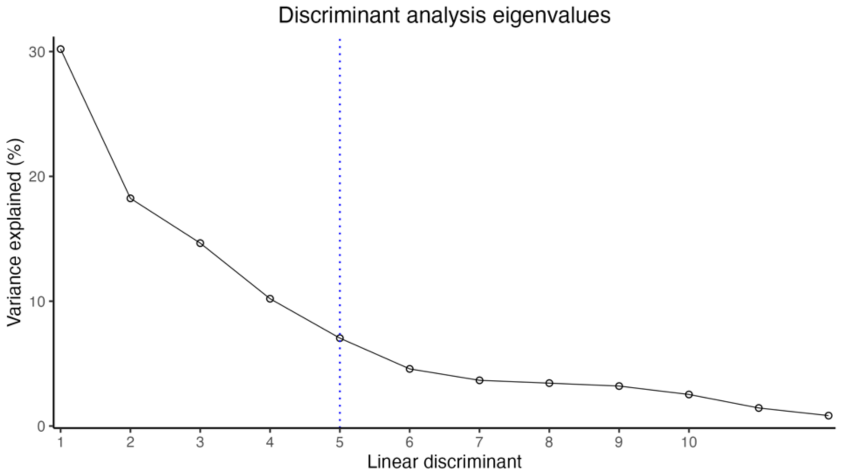


Figure 4. 13. Variance explained by linear discriminants in DAPC for K = 13

The assignment plot for K values 11 to 14 can be seen in **Figure 4. 14**. Such as in the fastSTRUCTURE clustering results, each bar represents an individual and the fill represents the membership probability for that cluster. There seems to be higher single cluster membership probabilities for all individuals, and much less admixture than in the fastSTRUCTURE results. Results for K = 13 and K = 14 show less admixture than results for K = 11 or K = 12.

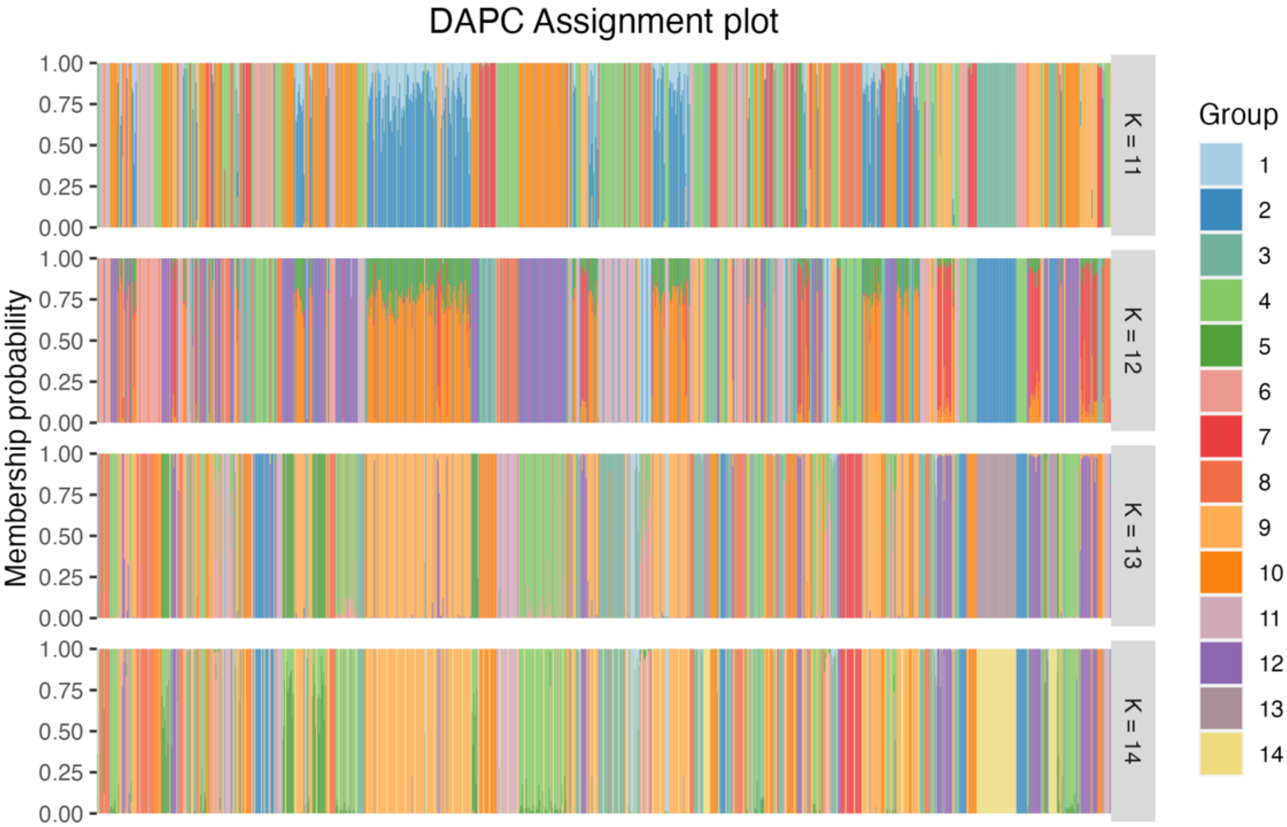


Figure 4. 14. Membership probabilities of individuals using DAPC for K values 11-14

Individuals belonging to cluster 3 in K=11, cluster 2 in K=12, cluster 13 in K=13 and cluster 14 in K=14 in **Figure 4. 14** are consistently clustered together across all K values. Such as in the fastSTRUCTURE results, the individuals in this group belong to the acl and inc species; however, individuals from *S. demissum* Lindley (dms) and *S. albicans* Ochoa (alb) are also in this group. Another group of individuals consistently clustered together are those in cluster 10 in K=11, cluster 12 in K=12, cluster 4 in K=13 and cluster 4 in K=14 in **Figure 4. 14**. These belong to the spl, ugt, vid, avl, blv, brc, opl, ifd, lph and ast. Individuals from species *S. tarijense* Hawkes (tar), *S. hoopesii* Hawkes & K. Okada (hps) and *S. incamayoense* K. Okada & A. Clausen (inm) are also included in this group. Moreover, cluster 9 in K = 13 and K = 14 shows high levels of membership probabilities. This group contains individuals from 30 different species; however, most belong to *S. bukasovii* Juz. (buk). All of these individuals belong to Peru and Bolivia. Finally, it is interesting that individuals from USA and Mexico are clustered together with membership probability values above 0.9 in cluster 9 in K = 13 and K = 14, and hints at shared climate tolerance characteristics. The geographical distribution of these clusters can be seen in **Figure 4. 15**.

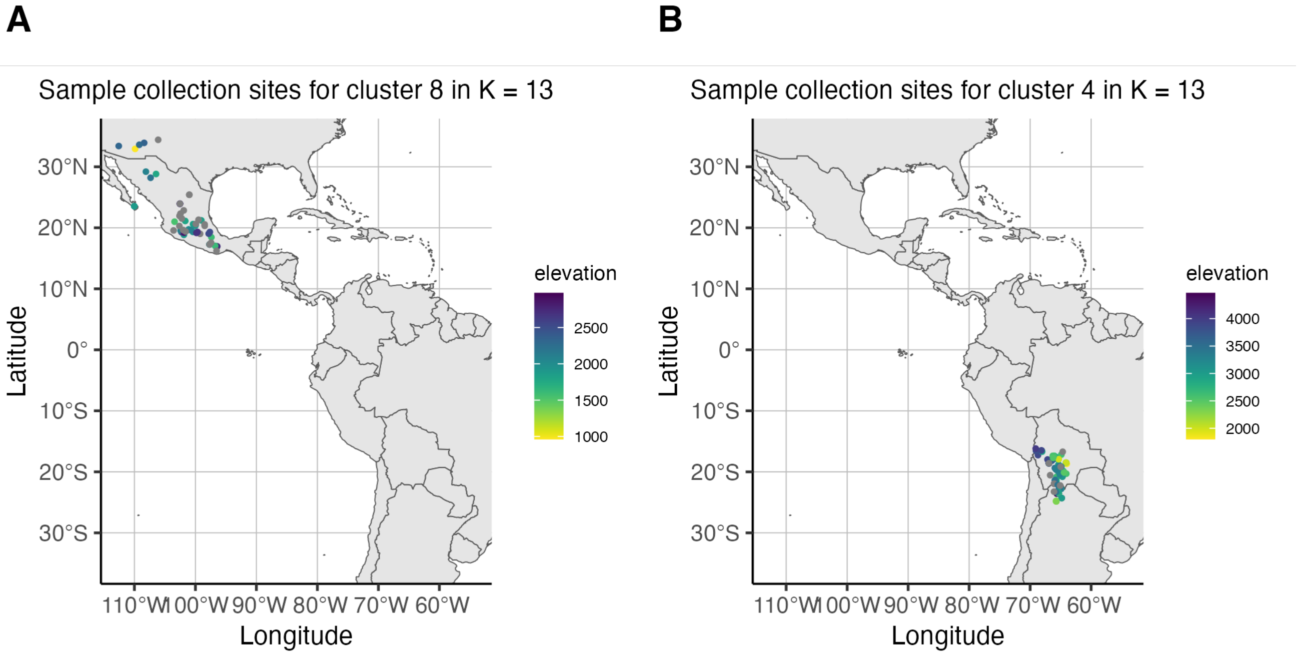


Figure 4. 15. Geographical distribution of individuals from cluster 8 and cluster 4 obtained with DAPC for K = 13

4.4.4. SVD with DA

The first step of SVD with DA was to carry out a SVD of the data. The variance explained by each singular value was plotted to choose the number of singular values that represented 80% of the variation in the original data. Thus, the number of singular values chosen was 50. This can be seen in **Figure 4. 16**.

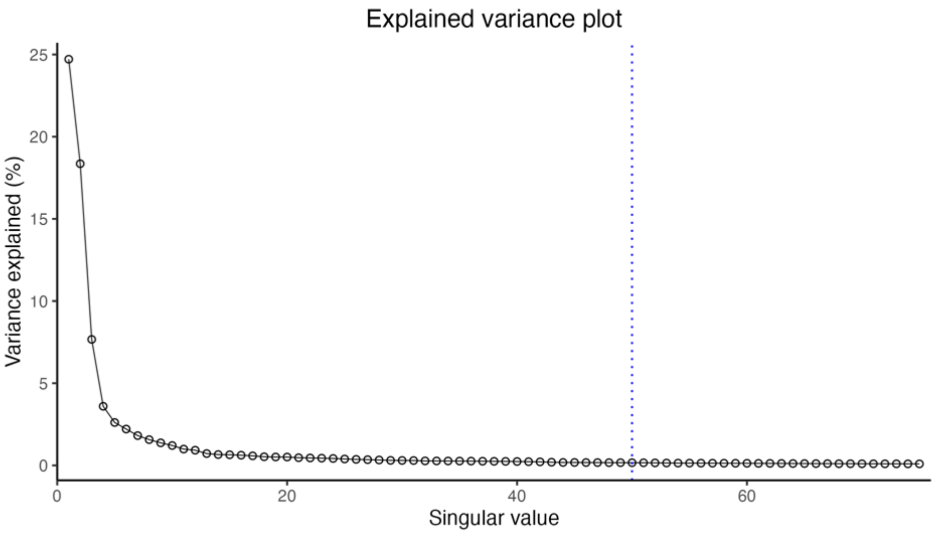


Figure 4. 16. Variance explained by each singular value

The data matrix was reduced using only the U component of the SVD and the 50 selected singular values. The optimum number of clusters was chosen by evaluating BIC values obtained through K-means clustering. The BIC plot can be seen in **Figure 4.17.** There seems to be an inflection point at around K = 9; however, considering the large groups identified in the dendrogram, the posterior DAs were carried out for an optimal K value range of 9 to 13. To represent 80% of the variation in the original data, 5 linear discriminants were retained for all K values. The variance explained by each linear discriminants for K = 13 can be seen in **Figure 4. 18**.

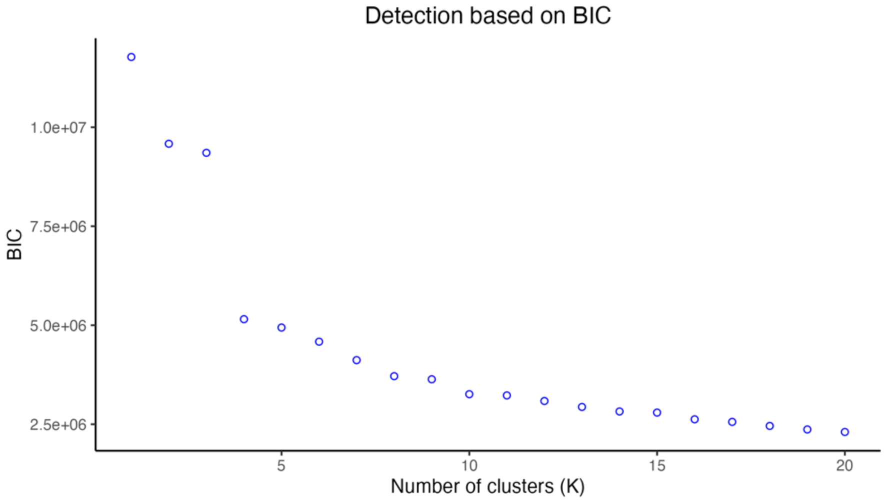


Figure 4. 17. Inference of number of clusters in SVD + DA according to BIC

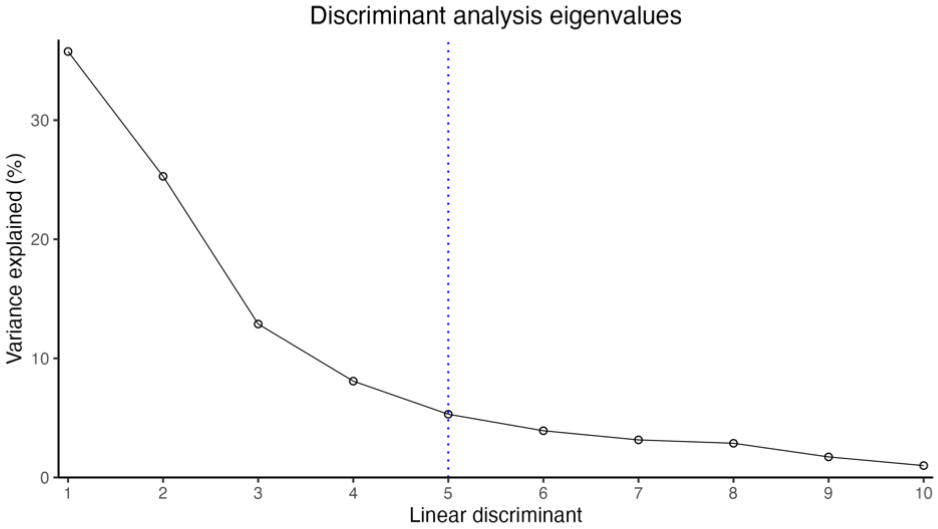


Figure 4. 18. Variance explained by linear discriminants in SVD + DA for K = 13

The assignment plot for K values 9 to 13 can be seen in **Figure 4.19.** Each bar represents an individual and the fill represents the membership probability for that cluster. Such as with DAPC, there seems to be higher single cluster membership probabilities for all individuals and much less admixture than in the fastSTRUCTURE results. Results for K = 13 and K = 14 show less admixture than results for K = 11 or K = 12.

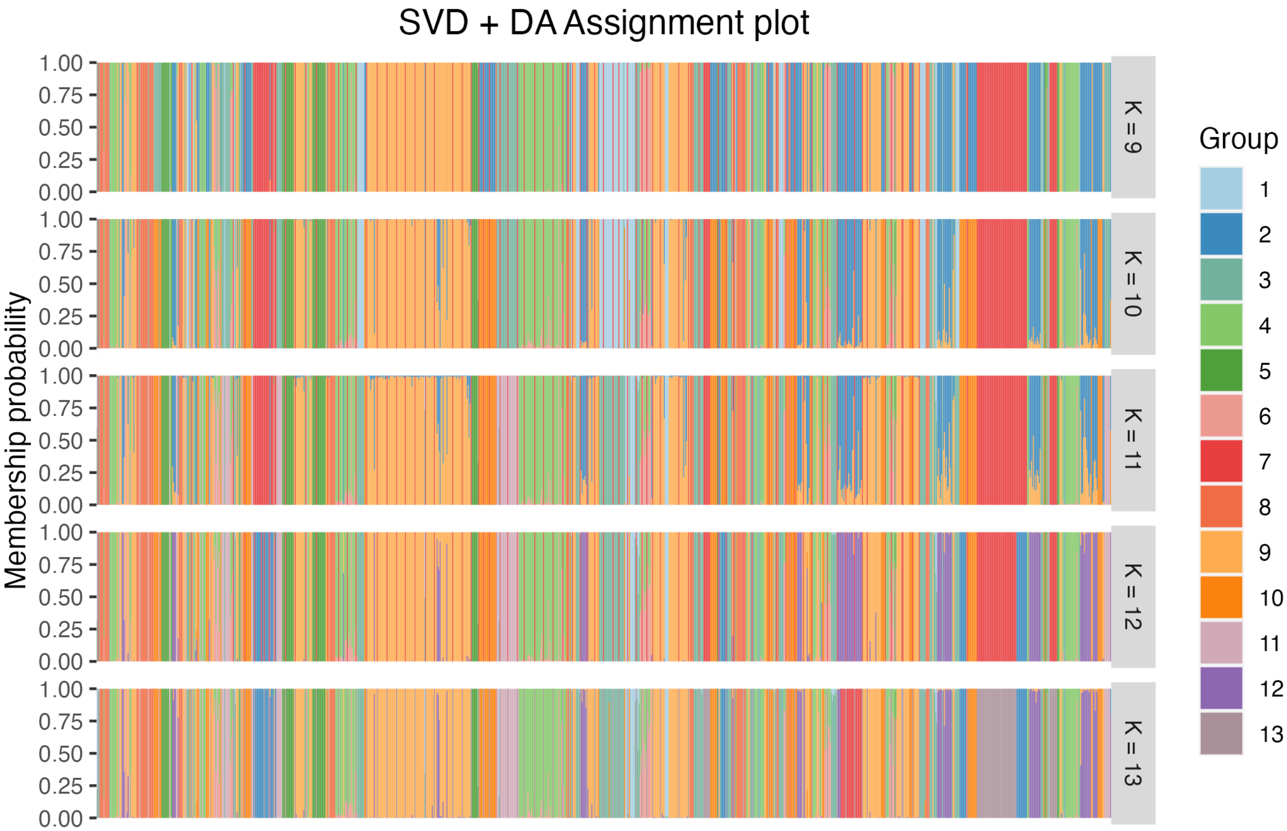


Figure 4. 19. Membership probabilities of individuals using SVD + DA for K values 9-13

For all K values there are certain individuals that tend to cluster together, such as those belonging to cluster 7 in K runs 9-12 and cluster 13 in K = 13. These belong to species acl, inc, dms and alb. In runs with K values 9-11 this cluster has around 164 individuals with a greater number of accessions from the alb species; however, in K = 13 this cluster has 135 individuals and a group of these alb individuals is segmented into its own individual cluster (cluster 2 in this case). Another group of individuals consistently clustered together are those represented by cluster 4 in all K runs. These individuals belong to species spl, tar, ugt, vid, avl, blv, brc, opl, hps, ifd, inm, lph, ast and *S. neorossii* Hawkes & Hjerting (nrs). In general, SVD with DA produced very similar results to those obtained by DAPC. This is important considering SVD with DA had not previously been used for population genetics. These results can promote its use in other population structure studies.

4.4.5. sNMF with ANLS

The sparse NMF was carried out on the formatted tetraploid data and the obtained cross-entropy results were plotted to define an optimal K value range. This plot can be seen in **Figure 4. 20**. There is not a particularly clear inflection point; however, the cross-entropy decreases by a negligible amount at K = 10 and the slope of the curve is the lowest at this point. Considering these results and the large groups found in the NJ dendrogram, the optimal K value range was chosen as 10 to 14.

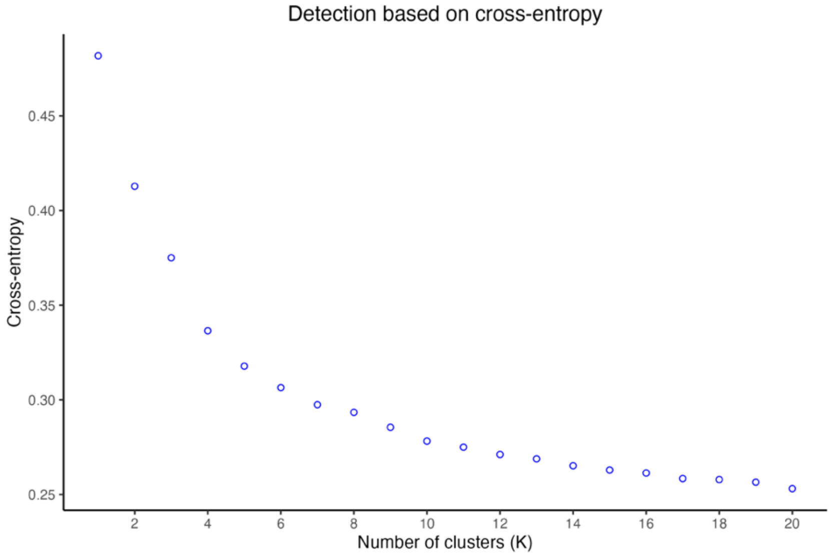


Figure 4. 20. Inference of number of clusters according to cross-entropy

The assignment plot for K values 10 to 14 can be seen in **Figure 4. 21**. Each bar represents an individual and the fill represents the membership probability for that cluster. The results show much more admixture than those obtained through any of the other methods.

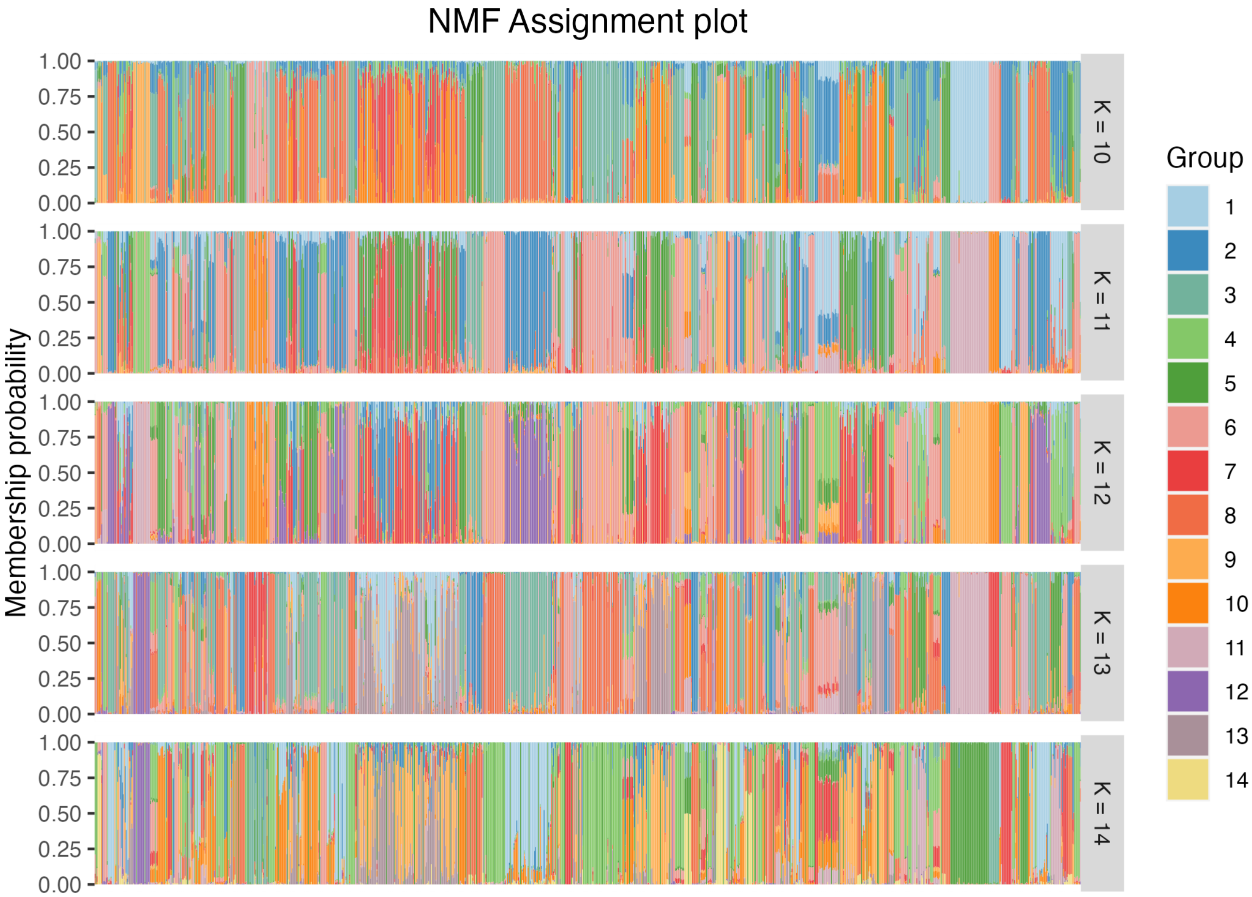


Figure 4. 21. Membership probabilities of individuals using sNMF for K values 10-14

Despite the high levels of admixture, once again there are certain individuals that tend to cluster together, such as those belonging to cluster 1 in K = 10, cluster 11 in K = 11, cluster 9 in K = 12, cluster 11 in K = 13 and cluster 5 in K = 14. These belong to species acl and inc. Individuals belonging to cluster 3 in K = 10, cluster 6 in K = 11, cluster 6 in K = 12, cluster 8 in K = 13 and cluster 4 in K = 14 are also consistently clustered together, although with lower membership probabilities in K = 13 and K = 14. These belong mostly to species *S. chomatophilum* Bitter (chm) and *S. multiinterruptum* Bitter (mtp), along with 20 other species from Peru.

4.4.6. Approach comparison for population structure analysis

Parametric and nonparametric methods were used for population structure analysis. These were compared in terms of the optimal K values identified, the assignment results, the obtained cluster characteristics, and the computational cost.

4.4.6.1. Optimal K values

The optimal K value range for each method was chosen according to the specific method's criteria. It is important to mention that there is generally no true value of K, and the appropriate choice of K must be interpreted while considering prior data about the data and type of sampling. This is especially true for model-based clustering, given samples in real populations rarely satisfy all the assumptions of the model [41]. This is why the taxonomic and geographical information of the individuals, and the large groups identified in the NJ dendrogram with Spooner's clades, were also considered to define this value range. The K value ranges for each method can be seen in **Table 4. 1**.

Method	Criteria	K value(s)	Chosen K value range
fastSTRUCTURE (simple prior)	Maximum likelihood	11 - 14	11 - 14
fastSTRUCTURE (logistic prior)	Maximum likelihood	1 - 9	11 - 14
DAPC	BIC	12	11 - 14
SVD + DA	BIC	9	9 - 13
sNMF + LS	Cross-entropy	10	10 - 14

Table 4. 1. K value range chosen for each analysis method

4.4.6.2. Assignment results

The assignment plot for each different method with K = 13 can be seen in **Figure 4. 22**. Despite the varying levels of admixture, especially when using NMF and fastSTRUCTURE, there are some individuals that are consistently clustered together across all methods with high membership probabilities of above 0.9. These also correspond to the individuals with highest membership probabilities across all K runs when evaluating each method's results. Individuals in cluster 11 for fastSTRUCTURE with simple prior, cluster 4 for fastSTRUCTURE with logistic prior, cluster 13 for DAPC and SVD with DA, and cluster 11 for sNMF are consistently grouped together. These correspond to species acl and inc. DAPC and SVD with DA also include individuals from dms and alb in this group.

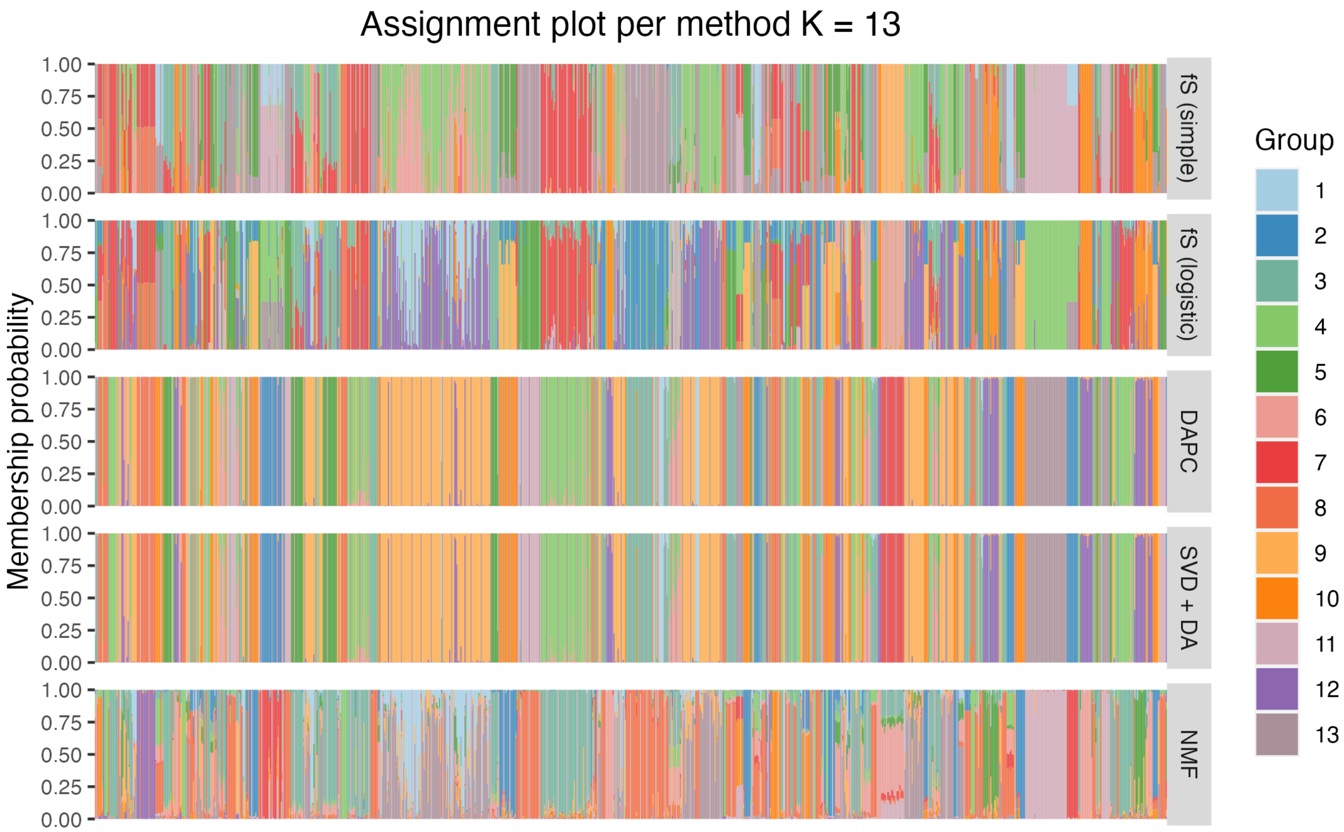


Figure 4. 22. Membership probabilities of individual for K = 13 using different methods

Another group of individuals consistently clustered together are those in cluster 7 for fastSTRUCTURE with simple and logistic priors, cluster 4 for DAPC and SVD with DA, and cluster 3 for sNMF. Membership probabilities for this group are considerably higher in

DAPC and SVD with DA and include individuals from more species; however, individuals from species spl, vid, brc, opl, ifd and lph are consistently assigned to this group. These individuals are all from Bolivia, Peru and Argentina and their geographical distribution can be observed in **Figure 4. 15**.

Some individuals consistently cluster together in the nonparametric methods with membership probabilities of above 0.9, but show membership probabilities of only around 0.5 in the parametric fastSTRUCTURE methods. One of these groups is composed by cluster 5 in fastSTRUCTURE with simple prior, cluster 9 in fastSTRUCTURE with logistic prior, cluster 10 in DAPC and SVD with DA, and cluster 2 in sNMF. These individuals belong to a range of species, mostly *S. colombianum* Dunal (col). Another of these groups is composed by cluster 8 in fastSTRUCTURE with simple and logistic priors, cluster 8 in DAPC and SVD with DA, and cluster 12 in sNMF. All of these individuals belong to the USA and Mexico, most belonging to the *S. stoloniferum* Schlechtdal (sto) species. Their geographical distribution can be observed in **Figure 4. 15**. Another of these groups is composed by cluster 11 in fastSTRUCTURE with simple prior, cluster 4 in fastSTRUCTURE with logistic prior, cluster 2 in DAPC and SVD with DA, and cluster 7 in sNMF. This group is mainly composed of individuals from *S. albicans* Ochoa (alb) species.

There are some groups that are consistent in the fastSTRUCTURE methods and the DAPC and SVD with DA methods, but not in sNMF. This is the case with cluster 10 in fastSTRUCTURE with both simple and logistic priors, cluster 12 with DAPC and SVD with DA, and cluster 6 in sNMF. This group is composed of individuals mainly from *S. medians var. autumnale* Ochoa (aut) and *S. raphanifolium* Cárdenas & Hawkes (rap); however, sNMF only includes individuals from the aut species, and separates those from rap into cluster 5.

There are other groups consistent in the fastSTRUCTURE methods and NMF, but different in DAPC and SVD with DA in terms of levels of admixture. This can be seen in cluster 9 in DAPC and SVD with DA, and the corresponding groups composed by both cluster 4 and 6 in fastSTRUCTURE with simple prior, cluster 1 and 12 in fastSTRUCTURE with logistic prior, and cluster 1 and 13 in sNMF. These belong to a range of species, mainly S. *bukasovil* Juzepzuck, which is present in both clusters in fastSTRUCTURE and sNMF.

The consistent division of this species group suggests it should be further explored in order to identify subspecies or varieties.

sNMF produced the results with the highest levels of admixture, followed by the fastSTRUCTURE methods. This was to be expected given fastSTRUCTURE works under an admixed subpopulations model assumption [41], and sNMF was proposed specifically to model ancestry proportions in admixed subpopulations [51]. Both the DAPC and SVD with DA produced very high levels of membership probabilities and gave little information regarding admixture between different groups of individuals, which relates to the use of K-means clustering for group assignment [43], [149]. All of these methods are exploratory and should be used in conjunction to explore the data, such that both the overall structure and admixture between subpopulations can be evaluated.

4.4.6.3. Cluster characteristics

Besides the assignment results comparison, the methods were also compared in terms of cluster characteristics. **Figure 4. 23** shows cluster characteristics for each method with K = 13, in terms of their composition and probability values. The boxplot shows the interquartile range, median and mean probability values for each cluster, excluding individuals with probability values below 0.01.

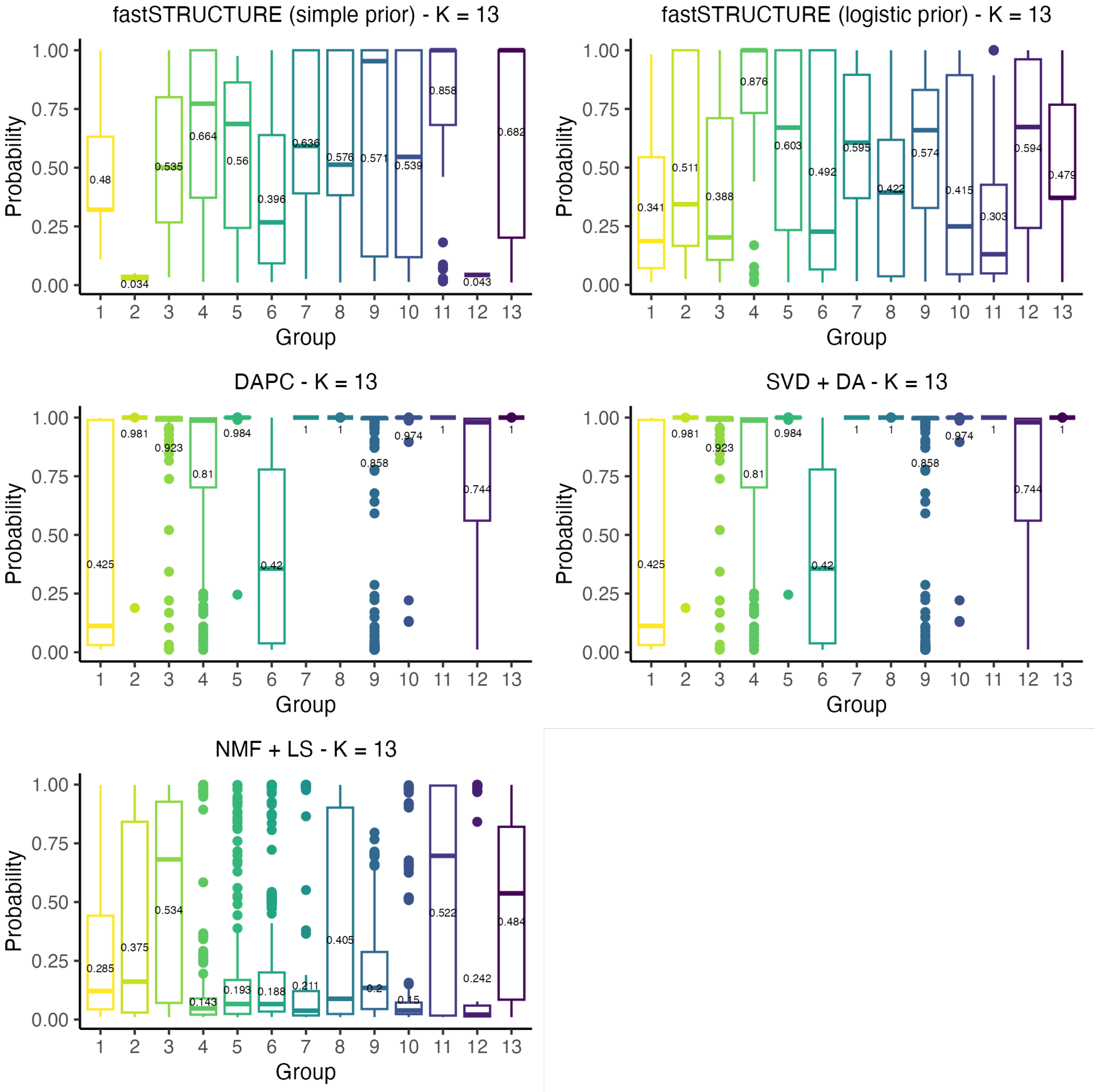


Figure 4. 23. Cluster characteristics for K = 13 in each analysis method

DAPC and SVD with DA have the highest membership probability values among each cluster, particularly in clusters 2, 3, 5, 7, 8, 10, 11 and 13. This was to be expected considering that as opposed to fastSTRUCTURE and NMF, the K means clustering used in DAPC and SVD with DA focuses on cluster assignment and not in modeling probabilities [43]. The cluster composition results for sNMF validate it as the method with the highest levels of admixture, with all clusters having mean and median probability values below 0.75.

On the other hand, fastSTRUCTURE results present mean and median values between 0.3 and 0.8 for all clusters, except for clusters 2, 11 and 12 when using the simple prior, and clusters 4 when using the logistic prior. Cluster 11 in fastSTRUCTURE with simple prior corresponds to cluster 4 in fastSTRUCTURE with logistic prior and cluster 11 in sNMF. The fact that these clusters have the highest mean and median cluster probability values for each method corresponds to the fact this is the most consistent group found across all methods and all K values, composed by individuals from acl and inc species.

The methods were also compared in terms of the number of individuals with membership probability values above 0.9. **Table 4.2** shows the results by each method. As expected, DAPC and SVD with DA have the highest number of individuals with such high membership probabilities, followed by fastSTRUCTURE with simple prior, fastSTRUCTURE with logistic prior, and sNMF.

Method	Number of individuals with membership probability > 0.9
fastSTRUCTURE (simple prior)	647
fastSTRUCTURE (logistic prior)	577
DAPC	981
SVD + DA	981
sNMF + LS	559

Table 4. 2. Number of individuals with membership probabilities > 0.9 for each method

4.4.6.4. Computational cost

Finally, the methods were compared in terms of their computational cost as execution time, which is an important criterion related to the computer power available for research organizations, since not all research organizations have powerful computing infrastructures available. All analysis methods were tested using the same standard computer with 8 GB of RAM. **Table 4. 3** shows the execution time for each method.

Method	Step	Time per step (s)	Total time (s)
fastSTRUCTURE (simple prior) K values 1-15	K = 1 (10 iterations)	21.7802	5526.2205
	K = 2 (20 iterations)	57.0727	
	K = 3 (30 iterations)	80.5322	
	K = 4 (50 iterations)	126.3130	
	K = 5 (50 iterations)	167.9194	
	K = 6 (60 iterations)	200.1893	
	K = 7 (60 iterations)	250.5735	
	K = 8 (110 iterations)	404.5261	
	K = 9 (60 iterations)	318.0575	
	K = 10 (60 iterations)	357.1579	
	K = 11 (50 iterations)	477.2411	
	K = 12 (100 iterations)	780.3237	
	K = 13 (60 iterations)	686.7493	
	K = 14 (80 iterations)	805.0135	
	K = 15 (60 iterations)	792.7711	
fastSTRUCTURE (logistic prior) K values 1-14	K = 1 (10 iterations)	69.2214	2046066.41
	K = 2 (12060 iterations)	23088.0102	
	K = 3 (8090 iterations)	23126.2106	
	K = 4 (1700 iterations)	11113.9243	
	K = 5 (2540 iterations)	21679.9817	
	K = 6 (30 iterations)	35017.6918	
	K = 7 (70 iterations)	70451.9279	
	K = 8 (140 iterations)	156073.9790	
	K = 9 (260 iterations)	277933.9986	
	K = 10 (180 iterations)	234240.3034	
	K = 11 (210 iterations)	306529.2953	

	K = 12 (100 iterations)	166476.9894	
	K = 13 (40 iterations)	118100.6670	
	K = 14 (360 iterations)	602164.2091	
DAPC	PCA	2563.068	2579.958779
	K-means clustering	4.807419	
	DA	12.08336	
SVD + DA	SVD	324.04212	325.4296251
	K-means clustering	0.60584	
	DA	0.7816651	
sNMF + LS	sparse NMF	810.8184	810.8184

Table 4. 3. Computational cost as execution time in seconds (s) for the parametric and nonparametric population structure analysis methods tested

The results show fastSTRUCTURE with logistic prior took a considerably longer time to carry out all K runs, taking a total of 2046066.41 seconds or 23.68 days to run. Even more, K = 15 could not be run using this method due to time constraints, and the maximum likelihood results could not be used to select an optimal K value range. fastSTRUCTURE with a simple prior had the second longest execution time, taking a total of 5526.2205 seconds or 1.54 hours to run. It was to be expected the two parametric methods had the longest execution times; nonetheless, the difference between prior choice is quite significant. Even though the logistic prior is recommended when populations are difficult to resolve, the assignment results did not show a particularly large difference when compared to the simple prior results. Focusing only on the parametric methods, fastSTRUCTURE with a simple prior produced the best results in a much more efficient and practical manner.

Regarding the nonparametric results, on the one hand, DAPC had the longest execution time, taking a total of 2579.958779 seconds or 43 minutes. The PCA, dimensionality reduction step, took the longest time to run, while the K-means clustering and DA had running times of only a few seconds. On the other hand, SVD with DA had the shortest execution time out of all methods, taking a total of 325.4296251 seconds or 5.42 minutes. The dimensionality reduction step, SVD in this case, was also the step that took the longest to run; however, it took considerably less time than the PCA in DAPC. This makes sense given computing the covariance matrix in PCA and computing the eigenvalue decomposition of this matrix is a computationally intensive step [150], especially when applied on such a large data matrix. It is important to mention that the PCA allows us to retain the loading information of the variables, or the SNP markers in this case. SVD loses this information while reducing the original data matrix. sNMF had the second longest execution time out of the nonparametric methods, taking a total of 810.8184 seconds or 15.51 minutes. It produced results comparable, in terms of assignment and admixture estimates, to those obtained through fastSTRUCTURE.

All population structure analysis methods were carried out on the same standard 8GB RAM computer. Execution times could have been considerably faster if carried out on an HPC environment; however, part of the objectives of this investigation was to produce a practical and accessible analysis pipeline, therefore it was decided not to use such expensive equipment. The workflow aimed to be replicable by other researchers without equipment limitations. Nonetheless, all analysis methods tested except for fastSTRUCTURE with logistic prior had much shorter execution times than the regular STRUCTURE program, even when the latter is run on a multicore HPC environment with less markers and individuals. In 2017, Chhatre et al analyzed 30 replicates for a 11,533 SNP marker and 57 individuals dataset using STRUCTURE with 60 cores on an HPC environment, and it took them a total of 9 days, without including the time required to define MCMC parameters [136].

4.4.6.5. Overall performance

The different nonparametric techniques tested have proven to be versatile and efficient when analyzing large genetic data and produce comparable results to those obtained through the most popular parametric methods used for population structure analysis. The significant differences in computational cost are important considering the remaining 50% of the collection is yet to be analyzed. These methods do not rely on any population genetics model assumptions such as HWE, LE or consistent ploidy [17], [43], [44], [51]. Moreover, DAPC and SVD with DA can be used in conjunction with NMF to explore the different levels of structure in the data and examine admixture between subpopulations. Low-rank approximations such as SVD and sNMF are preferred when the number of variables in the data is much larger than the number of observations, such as in this investigation, given PCA on high-dimensionality settings can be inconsistent [151]. Sparse PCA is usually proposed as a better alternative for these cases [151].

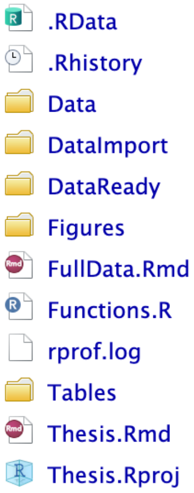
Furthermore, the population structure analysis methods were implemented such as they were first described in their first publication; however, there are certain modifications that could be done to produce better results, especially when analyzing high-dimensional and complex datasets. For example, different clustering techniques could be used to produce a better assignment of individuals in DAPC or SVD with DA [17], [148]. Dimensionality reduction methods such as sparse PCA could have been a better alternative for this type of data [150].

FastSTRUCTURE was chosen as the parametric method due to time constraints of the investigation. Nonetheless, STRUCTURE tends to be the preferred method in the field given it is the most robust parametric method of analysis of genetic structure, and despite also depending on model assumptions, allows for mixed ploidy populations [152]. Analyzing the data using STRUCTURE would allow the nonparametric techniques used to be further validated as comparable and efficient alternatives. Due to how computationally intensive STRUCTURE is, a core collection would first need to be defined in order to retain diversity representation while making the execution time more manageable. This would be done by choosing the markers with most distinguishing power, equivalent to approximately 10% of the total SNPs. The number of individuals would also be filtered considering clade, species and geographical region as important selection attributes. This can be done using core selector tools such as CoreSNP [153].

4.5. R Workflow

Part of the objectives of this thesis was to produce a scalable and replicable R workflow such that other researchers can use the code and carry out analyses on other sets of SNP data. The project was carried out as an R project and the distribution of directories can be seen in **Figure 4. 24**. The main project file is “Thesis.Rmd” and includes all the code for the genetic diversity and population structure analyses. The data filtering was carried out in the “FullData.Rmd” file. The code for all steps is included as annexes in the present document and is linked to each part of the methodology in its corresponding section. The code is written such that only the initial variables have to be defined for the rest of the code to run. Each part is correctly annotated; therefore, any changes or adaptations can be easily made, particularly concerning parameter choice, data formatting, or visualization. The code for all plots is also included in the Rmd file.

This thesis analyzed only 50% of the wild potato collection due to economic constraints. Nonetheless, due to the scalability and replicability of the workflow proposed, it can be used to analyze the remaining 50% of the collection, facilitating the analysis process. This remaining 50% will be genotyped and analyzed within the upcoming year. Additionally, the structure of smaller groups based on taxonomic clades or species can also be explored using the same methods used in this study. This would allow for more subtle differences to be identified and linked to the accessions’ unique characteristics. Finally, this work could serve as the basis for the development of deep learning methods for geographic location or taxonomic prediction of accessions of unknown origin.



.RData
.Rhistory
Data
DataImport
DataReady
Figures
FullData.Rmd
Functions.R
rprof.log
Tables
Thesis.Rmd
Thesis.Rproj

Figure 4. 24. Project directory distribution

CONCLUSIONS

This study was able to develop a standardized R workflow of analysis for population genetics studies using SNP datasets to analyze the genetic identity, diversity, and population structure of CIP's wild potato germplasm collection, which will be available for use in similar population genetics studies.

1. The genetic identity of the genotyped accessions was defined by their most valuable SNP information according to the defined filtering criteria. Uninformative SNPs and poor-quality data were filtered to optimize computational cost while retaining most of the genetic variation. This information will eventually be made available through CIPs open access database to improve the efficiency of germplasm conservation in the genebank, as it will allow duplicates, mislabeled samples, and unrecognized variants to be identified. The genetic characterization of these accessions will reduce costs related to germplasm maintenance in the long term and ensure researchers receive the material they require.

2. The genetic diversity of this previously unexplored set of wild potato accessions was able to be determined through the estimation of key genetic diversity parameters such as heterozygosity. These parameters were also explored by ploidy and species characteristics. Having a better understanding of the genetic diversity of wild potato species will facilitate their use within plant breeding programs and other investigations such as genome wide association studies (GWAS), as these indices of diversity can be associated with phenotypic trait characteristics.

3. The population structure of the wild potato collection was explored through different methods and linked to the ploidy, taxonomy, and geographic characteristics of the accessions. The data is inherently complex due to the high levels of diversity within wild potatoes; thus, we did not expect fully conclusive results regarding the structure of the collection. Nonetheless, this investigation offers a first insight into the population

structure of wild potatoes and provides a genetic backing to the previously reported taxonomic and geographical relationships between certain species. The distance based nonparametric approaches revealed overall similarities in the genetic profiles of individuals from similar geographical regions, ploidy levels and taxonomic clade. The dimensionality reduction based nonparametric approaches and parametric analysis method confirmed these relationships and allowed to further explore the population architecture and admixture levels. Further population structure classification and inference must be done by a specialized taxonomist to fully relate the different characteristics of the accessions with their genetic profiles. This will allow for the application of focused identification of germplasm strategies (FIGS) to identify species carrying specific adaptative traits, enabling the use of these species for marker-assisted selection (MAS). Additionally, the combination of genetic diversity and population structure characterization will allow collection curators to identify genetic or geographical bias present within the collection, allowing them to then correct this bias.

4. Parametric and nonparametric population structure analysis methods were successfully implemented and compared in terms of clustering capabilities and computational cost. The distance based nonparametric techniques allowed the general distribution to be observed. Both the dimensionality reduction based nonparametric and parametric techniques were able to render clustering results consistent with the individuals' ploidy levels, taxonomy, and geographical characteristics. The dimensionality reduction based nonparametric techniques showed promising results regarding their ability to identify population structure, infer appropriate numbers of subpopulations, and assign individuals to each. As opposed to their parametric counterpart, these methods do not depend on any genetic assumptions or models. The parametric method used, fastSTRUCTURE, was evaluated using both simple and logistic prior distributions. The logistic distribution had been previously recommended for populations with complex structures; however, the results using both priors were very similar and the execution time for the logistic prior was over 370 times greater than that of the simple prior, making it impractical for research organizations without HPC facilities. Overall, the nonparametric methods tested required

considerably less computational cost and execution time than the parametric methods. The SVD with DA method, which had not been previously used for population genetics studies, produced very similar results to those obtained through DAPC, requiring considerably less execution time than any other method and only 17% of the time required by fastSTRUCTURE with a simple prior. sNMF produced comparable admixture results to those obtained through fastSTRUCTURE. sNMF presented mean and median cluster probability values below 0.75, very similar to fastSTRUCTURE, which has most cluster mean and median probability values below 0.8. The number of individuals with membership probabilities above 0.9 for sNMF and fastSTRUCTURE with simple and logistic priors were 559, 647 and 577 respectively. Therefore, these nonparametric methods provide a useful, practical, and less costly alternative for the population structure analysis of large genetic datasets, especially when used in conjunction, either DAPC or SVD with DA along with sNMF, such that overall structure and admixture of the data can be explored.

5. Finally, the workflow followed for this study, consisting of data preprocessing, genetic diversity analysis, population structure analysis and approach comparison, can be replicated and adapted for use in other population genetics studies using SNP data. The produced R workflow was written such that the analyses run automatically once the initial variables are defined. The convenience and effectiveness of this programming language for this type of investigations was confirmed due to the large amount of open access packages available, the flexibility of data structuring options and cross-platform compatibility. The analysis methods not readily available through R packages were successfully implemented from scratch. Moreover, the code includes data visualization sections, which facilitate posterior result analysis. This R workflow will be readily available for use in CIP and partner organizations. It will be included as part of CIPs institutional manual on analysis of crop SNP data; however, it can also be uploaded to online repositories such that other people can reproduce these types of analyses. This will encourage more of these population genetics studies to be carried out and investigate previously unexplored genetic datasets. This is an important contribution for research for

megadiverse countries, such as Peru, that hold large biological diversity and are just beginning to understand the meaning of that diversity.

BIBLIOGRAPHY

[1] "Biodiversity for the future program," International Potato Center. Accessed: Sep. 04, 2022. [Online]. Available: https://cipotato.org/research/biodiversity-future-program/

[2] "Genebanks," CGIAR Genebank Platform. Accessed: Sep. 04, 2022. [Online]. Available: https://www.genebanks.org/genebanks/

[3] T. G. Benton, C. Bieg, H. Harwatt, R. Pudasaini, and L. Wellesley, "Food system impacts on biodiversity loss," p. 75.

[4] S. H. Jansky *et al.*, "A Case for Crop Wild Relative Preservation and Use in Potato," *Crop Science*, vol. 53, no. 3, pp. 746–754, 2013, doi: 10.2135/cropsci2012.11.0627.

[5] N. L. Anglin *et al.*, "Genetic Identity, Diversity, and Population Structure of CIP's Sweetpotato (I. batatas) Germplasm Collection," *Frontiers in Plant Science*, vol. 12, 2021, Accessed: Sep. 04, 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpls.2021.660012

[6] M. J. Díez *et al.*, "Plant Genebanks: Present Situation and Proposals for Their Improvement. the Case of the Spanish Network," *Frontiers in Plant Science*, vol. 9, 2018, Accessed: Sep. 06, 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpls.2018.01794

[7] M. Lee, "Genome projects and gene pools: New germplasm for plant breeding?," *Proceedings of the National Academy of Sciences*, vol. 95, no. 5, pp. 2001–2004, Mar. 1998, doi: 10.1073/pnas.95.5.2001.

[8] CPAD, "Wild potatoes," International Potato Center. Accessed: Sep. 11, 2022. [Online]. Available: https://cipotato.org/outcomes/wild-potatoes/

[9] J. Berdugo-Cely, R. I. Valbuena, E. Sánchez-Betancourt, L. S. Barrero, and R. Yockteng, "Genetic diversity and association mapping in the Colombian Central Collection of Solanum tuberosum L. Andigenum group using SNPs markers," *PLOS ONE*, vol. 12, no. 3, p. e0173039, Mar. 2017, doi: 10.1371/journal.pone.0173039.

[10] D. Ellis *et al.*, “Genetic identity in genebanks: application of the SolCAP 12K SNP array in fingerprinting and diversity analysis in the global in trust potato collection,” *Genome*, vol. 61, no. 7, pp. 523–537, Jul. 2018, doi: 10.1139/gen-2017-0201.

[11] L. Alhusain and A. M. Hafez, “Nonparametric approaches for population structure analysis,” *Human Genomics*, vol. 12, no. 1, p. 25, May 2018, doi: 10.1186/s40246-018-0156-4.

[12] J. Novembre, “Pritchard, Stephens, and Donnelly on Population Structure,” *Genetics*, vol. 204, no. 2, pp. 391–393, Oct. 2016, doi: 10.1534/genetics.116.195164.

[13] M. Hamilton, *Population Genetics*. Wiley, 2009.

[14] J. K. Pritchard, M. Stephens, and P. Donnelly, “Inference of population structure using multilocus genotype data.,” *Genetics*, vol. 155, no. 2, pp. 945–959, Jun. 2000.

[15] L. Porras-Hurtado, Y. Ruiz, C. Santos, C. Phillips, Á. Carracedo, and M. Lareu, “An overview of STRUCTURE: applications, parameter settings, and supporting software,” *Frontiers in Genetics*, vol. 4, 2013, Accessed: Sep. 11, 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fgene.2013.00098

[16] N. Patterson, A. L. Price, and D. Reich, “Population Structure and Eigenanalysis,” *PLOS Genetics*, vol. 2, no. 12, p. e190, Dec. 2006, doi: 10.1371/journal.pgen.0020190.

[17] N. Liu and H. Zhao, “A non-parametric approach to population structure inference using multilocus genotypes,” *Hum Genomics*, vol. 2, no. 6, pp. 353–367, Jun. 2006, doi: 10.1186/1479-7364-2-6-353.

[18] J. Bergelson, E. S. Buckler, J. R. Ecker, M. Nordborg, and D. Weigel, “A Proposal Regarding Best Practices for Validating the Identity of Genetic Stocks and the Effects of Genetic Variants[OPEN],” *Plant Cell*, vol. 28, no. 3, pp. 606–609, Mar. 2016, doi: 10.1105/tpc.15.00502.

[19] A. E. Anastasio *et al.*, “Source verification of mis-identified Arabidopsis thaliana accessions,” *The Plant Journal*, vol. 67, no. 3, pp. 554–566, 2011, doi: 10.1111/j.1365-313X.2011.04606.x.

[20] J. Bergelson, E. S. Buckler, J. R. Ecker, M. Nordborg, and D. Weigel, “A Proposal Regarding Best Practices for Validating the Identity of Genetic Stocks and the Effects of Genetic Variants,” *Plant Cell*, vol. 28, no. 3, pp. 606–609, Mar. 2016, doi: 10.1105/tpc.15.00502.

[21] P. Kellar and J. Pires, “Biodiversity assessment: State-of-the-art techniques in phylogenomics and species identification,” *American journal of botany*, vol. 98, pp. 415–25, Mar. 2011, doi: 10.3732/ajb.1000296.

[22] M. A. Hardigan, J. Bamberg, C. R. Buell, and D. S. Douches, “Taxonomy and Genetic Differentiation among Wild and Cultivated Germplasm of Solanum sect. Petota,” *Plant Genome*, vol. 8, no. 1, p. eplantgenome2014.06.0025, Mar. 2015, doi: 10.3835/plantgenome2014.06.0025.

[23] A. M. D. Ron and A. P. Rodiño, “Analysis of Crop Genetic and Germplasm Diversity,” *Agronomy*, vol. 12, no. 1, Art. no. 1, Jan. 2022, doi: 10.3390/agronomy12010091.

[24] M. D. Hayward and E. L. Breese, “Population structure and variability,” in *Plant Breeding: Principles and prospects*, M. D. Hayward, N. O. Bosemark, I. Romagosa, and M. Cerezo, Eds., in Plant Breeding Series. , Dordrecht: Springer Netherlands, 1993, pp. 16–29. doi: 10.1007/978-94-011-1524-7_3.

[25] M. Berhe *et al.*, “Genome-wide association study and its applications in the non-model crop Sesamum indicum,” *BMC Plant Biology*, vol. 21, no. 1, p. 283, Jun. 2021, doi: 10.1186/s12870-021-03046-x.

[26] A. Bohra *et al.*, “Reap the crop wild relatives for breeding future crops,” *Trends in Biotechnology*, vol. 40, no. 4, pp. 412–431, Apr. 2022, doi: 10.1016/j.tibtech.2021.08.009.

[27] M. M. Jacobs, M. J. Smulders, R. G. van den Berg, and B. Vosman, “What’s in a name; Genetic structure in Solanum section Petota studied using population-genetic tools,” *BMC Evolutionary Biology*, vol. 11, no. 1, p. 42, Feb. 2011, doi: 10.1186/1471-2148-11-42.

[28] B. Huang, H. Ruess, Q. Liang, C. Colleoni, and D. M. Spooner, “Analyses of 202 plastid genomes elucidate the phylogeny of Solanum section Petota,” *Sci Rep*, vol. 9, no. 1, Art. no. 1, Mar. 2019, doi: 10.1038/s41598-019-40790-5.

[29] D. Tang *et al.*, “Genome evolution and diversity of wild and cultivated potatoes,” *Nature*, vol. 606, no. 7914, Art. no. 7914, Jun. 2022, doi: 10.1038/s41586-022-04822-x.

[30] M. A. Hardigan *et al.*, “Genome diversity of tuber-bearing Solanum uncovers complex evolutionary history and targets of domestication in the cultivated potato,” *Proceedings of the National Academy of Sciences*, vol. 114, no. 46, pp. E9999–E10008, Nov. 2017, doi: 10.1073/pnas.1714380114.

[31] Y. Li *et al.*, “Genomic Analyses Yield Markers for Identifying Agronomically Important Genes in Potato,” *Molecular Plant*, vol. 11, no. 3, pp. 473–484, Mar. 2018, doi: 10.1016/j.molp.2018.01.009.

[32] B. Huang, D. M. Spooner, and Q. Liang, “Genome diversity of the potato,” *Proceedings of the National Academy of Sciences*, vol. 115, no. 28, pp. E6392–E6393, Jul. 2018, doi: 10.1073/pnas.1805917115.

[33] “GGP Potato Arrays.” Accessed: Oct. 02, 2022. [Online]. Available: https://www.illumina.com/products/by-type/microarray-kits/ggp-potato.html

[34] J. Pandey *et al.*, “Genetic diversity and population structure of advanced clones selected over forty years by a potato breeding program in the USA,” *Sci Rep*, vol. 11, no. 1, Art. no. 1, Apr. 2021, doi: 10.1038/s41598-021-87284-x.

[35] B. G. Díaz *et al.*, “Genome-wide SNP analysis to assess the genetic population structure and diversity of Acrocomia species,” *PLOS ONE*, vol. 16, no. 7, p. e0241025, Jul. 2021, doi: 10.1371/journal.pone.0241025.

[36] J. M. Muñoz-Pérez, G. P. Cañas, L. López, and T. Arias, “Genome-wide diversity analysis to infer population structure and linkage disequilibrium among Colombian coconut germplasm,” *Sci Rep*, vol. 12, no. 1, Art. no. 1, Feb. 2022, doi: 10.1038/s41598-022-07013-w.

[37] D. H. Alexander, J. Novembre, and K. Lange, “Fast model-based estimation of ancestry in unrelated individuals,” *Genome Res*, vol. 19, no. 9, pp. 1655–1664, Sep. 2009, doi: 10.1101/gr.094052.109.

[38] J. Corander, P. Waldmann, P. Marttinen, and M. J. Sillanpää, “BAPS 2: enhanced possibilities for the analysis of genetic population structure,” *Bioinformatics*, vol. 20, no. 15, pp. 2363–2369, Oct. 2004, doi: 10.1093/bioinformatics/bth250.

[39] T. H, P. J, W. P, and R. Nj, “Estimation of individual admixture: analytical and study design considerations,” *Genetic epidemiology*, vol. 28, no. 4, May 2005, doi: 10.1002/gepi.20064.

[40] S. Purcell and P. Sham, “Properties of Structured Association Approaches to Detecting Population Stratification,” *HHE*, vol. 58, no. 2, pp. 93–107, 2004, doi: 10.1159/000083030.

[41] A. Raj, M. Stephens, and J. K. Pritchard, “fastSTRUCTURE: Variational Inference of Population Structure in Large SNP Data Sets,” *Genetics*, vol. 197, no. 2, pp. 573–589, Jun. 2014, doi: 10.1534/genetics.114.164350.

[42] T. Limpiti *et al.*, “Study of large and highly stratified population datasets by combining iterative pruning principal component analysis and structure,” *BMC Bioinformatics*, vol. 12, p. 255, Jun. 2011, doi: 10.1186/1471-2105-12-255.

[43] T. Jombart, S. Devillard, and F. Balloux, “Discriminant analysis of principal components: a new method for the analysis of genetically structured populations,” *BMC Genetics*, vol. 11, no. 1, p. 94, Oct. 2010, doi: 10.1186/1471-2156-11-94.

[44] N. Elhadji Ille Gado, E. Grall-Maës, and M. Kharouf, “Linear Discriminant Analysis based on Fast Approximate SVD,” Feb. 2017. doi: 10.5220/0006148603590365.

[45] E. Frichot and O. François, “LEA: An R package for landscape and ecological association studies,” *Methods in Ecology and Evolution*, vol. 6, no. 8, pp. 925–929, 2015, doi: 10.1111/2041-210X.12382.

[46] 677 Huntington Avenue Boston and Ma 02115 +1495-1000, “Software,” Alkes Price’s Faculty Website. Accessed: Oct. 19, 2022. [Online]. Available: https://www.hsph.harvard.edu/alkes-price/software/

[47] T. Jombart, “adegenet: a R package for the multivariate analysis of genetic markers,” *Bioinformatics*, vol. 24, no. 11, pp. 1403–1405, Jun. 2008, doi: 10.1093/bioinformatics/btn129.

[48] C. Lee, A. Abdool, and C.-H. Huang, “PCA-based population structure inference with generic clustering algorithms,” *BMC Bioinformatics*, vol. 10, no. 1, p. S73, Jan. 2009, doi: 10.1186/1471-2105-10-S1-S73.

[49] A. Intarapanich *et al.*, “Iterative pruning PCA improves resolution of highly structured populations,” *BMC Bioinformatics*, vol. 10, no. 1, p. 382, Nov. 2009, doi: 10.1186/1471-2105-10-382.

[50] C. Amornbunchornvej, T. Limpiti, A. Assawamakin, A. Intarapanich, and S. Tongsima, “Improved iterative pruning principal component analysis with graph-theoretic hierarchical clustering,” in *2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, May 2012, pp. 1–4. doi: 10.1109/ECTICon.2012.6254120.

[51] E. Frichot, F. Mathieu, T. Trouillon, G. Bouchard, and O. François, “Fast and Efficient Estimation of Individual Ancestry Coefficients,” *Genetics*, vol. 196, no. 4, pp. 973–983, Apr. 2014, doi: 10.1534/genetics.113.160572.

[52] “Potato,” International Potato Center. Accessed: Oct. 05, 2022. [Online]. Available: https://cipotato.org/potato/

[53] “FAOSTAT.” Accessed: Oct. 05, 2022. [Online]. Available: https://www.fao.org/faostat/en/#data/QV/visualize

[54] “PBI Solanum: A worldwide treatment | Solanaceae Source.” Accessed: Oct. 05, 2022. [Online]. Available: https://solanaceaesource.myspecies.info/content/pbi-solanum-worldwide-treatment

[55] J. G. Hawkes, “The potato: evolution, biodiversity and genetic resources.,” *The potato: evolution, biodiversity and genetic resources.*, 1990, Accessed: Oct. 05, 2022. [Online]. Available: https://www.cabdirect.org/cabdirect/abstract/19901615687

[56] D. M. Spooner, M. Ghislain, R. Simon, S. H. Jansky, and T. Gavrilenko, “Systematics, Diversity, Genetics, and Evolution of Wild and Cultivated Potatoes,” *Bot. Rev.*, vol. 80, no. 4, pp. 283–383, Dec. 2014, doi: 10.1007/s12229-014-9146-y.

[57] D. M. Spooner, “The potato: Evolution, biodiversity and genetic resources. J.G. Hawkes,” *Am. J. Pot Res*, vol. 67, no. 10, pp. 733–735, Oct. 1990, doi: 10.1007/BF03044023.

[58] D. M. Spooner and R. J. Hijmans, “Potato systematics and germplasm collecting, 1989–2000,” *Amer J of Potato Res*, vol. 78, no. 4, p. 237, Jul. 2001, doi: 10.1007/BF02875691.

[59] C. M. Ochoa, *Las papas de Sudamérica: Perú*. International Potato Center, 1999.

[60] D. A. Sotomayor *et al.*, “Collecting wild potato species (Solanum sect. Petota) in Peru to enhance genetic representation and fill gaps in ex situ collections,” *Frontiers in Plant Science*, vol. 14, 2023, Accessed: Jun. 27, 2023. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpls.2023.1044718

[61] D. M. Spooner and W. L. A. Hetterscheid, “Origins, Evolution, and Group Classification of Cultivated Potatoes,” *Darwin’s harvest : new approaches to the origins, evolution, and conservation of crops / edited by Timothy J. Motley, Nyree Zerega, and Hugh Cross*, 2006, Accessed: Oct. 11, 2022. [Online]. Available: http://hdl.handle.net/10113/3260

[62] R. Machida-Hirano, “Diversity of potato genetic resources,” *Breed Sci*, vol. 65, no. 1, pp. 26–40, Mar. 2015, doi: 10.1270/jsbbs.65.26.

[63] J. E. Bradshaw and G. R. Mackay, “Potato genetics,” CAB INTERNATIONAL, 1994. Accessed: Oct. 05, 2022. [Online]. Available: https://scholar.google.com/scholar_lookup?title=Potato+genetics&author=Bradshaw%2C+J.E.&publication_year=1994

[64] R. E. Hanneman, “The potato germplasm resource,” *American Potato Journal*, vol. 66, no. 10, pp. 655–667, Oct. 1989, doi: 10.1007/BF02853985.

[65] R. K. Saini and Y.-S. Keum, “Significance of Genetic, Environmental, and Pre- and Postharvest Factors Affecting Carotenoid Contents in Crops: A Review,” *J Agric Food Chem*, vol. 66, no. 21, pp. 5310–5324, May 2018, doi: 10.1021/acs.jafc.8b01613.

[66] N. S. Brar, S. P. Sharma, P. Kaushik, N. S. Brar, S. P. Sharma, and P. Kaushik, *Visiting Potato from a Breeding Perspective: Accomplishments and Prospects*. IntechOpen, 2021. doi: 10.5772/intechopen.98519.

[67] K. Watanabe, M. Orrillo, M. Iwanaga, R. Ortiz, R. Freyre, and S. Perez, “Diploid potato germplasm derived from wild and land race genetic resources,” *American Potato Journal*, vol. 71, no. 9, pp. 599–604, Sep. 1994, doi: 10.1007/BF02851525.

[68] M. W. Bonierbale, W. R. Amoros, E. Salas, and W. de Jong, “Potato Breeding,” in *The Potato Crop: Its Agricultural, Nutritional and Social Contribution to Humankind*, H. Campos and O. Ortiz, Eds., Cham: Springer International Publishing, 2020, pp. 163–217. doi: 10.1007/978-3-030-28683-5_6.

[69] D. Charlesworth and J. H. Willis, “The genetics of inbreeding depression,” *Nat Rev Genet*, vol. 10, no. 11, Art. no. 11, Nov. 2009, doi: 10.1038/nrg2664.

[70] “How Potato Grows,” International Potato Center. Accessed: Oct. 10, 2022. [Online]. Available: https://cipotato.org/potato/how-potato-grows/

[71] “Potato Breeding Research,” National Institute of Food and Agriculture. Accessed: Oct. 05, 2022. [Online]. Available: http://www.nifa.usda.gov/grants/funding-opportunities/potato-breeding-research

[72] “Potato agri-food systems program,” International Potato Center. Accessed: Oct. 05, 2022. [Online]. Available: https://cipotato.org/research/potato-agri-food-systems-program/

[73] J. G. Hawkes, “Significance of wild species and primitive forms for potato breeding,” *Euphytica*, vol. 7, no. 3, pp. 257–270, Oct. 1958, doi: 10.1007/BF00025267.

[74] H. Ross, “The use of wild solanum species in German potato breeding of the past and today,” *American Potato Journal*, vol. 43, no. 3, pp. 63–80, Mar. 1966, doi: 10.1007/BF02861579.

[75] B. Flis, J. Hennig, D. Strzelczyk-Żyta, C. Gebhardt, and W. Marczewski, “The Ry-fstogene from Solanum stoloniferum for extreme resistant to Potato virus Y maps to potato chromosome XII and is diagnosed by PCR marker GP122718 in PVY resistant potato cultivars,” *Mol Breeding*, vol. 15, no. 1, pp. 95–101, Jan. 2005, doi: 10.1007/s11032-004-2736-3.

[76] C. Fowler and T. Hodgkin, “PLANT GENETIC RESOURCES FOR FOOD AND AGRICULTURE: Assessing Global Availability,” *Annual Review of Environment and Resources*, vol. 29, no. 1, pp. 143–179, 2004, doi: 10.1146/annurev.energy.29.062403.102203.

[77] “In Vitro Collection,” CIP Genebank. Accessed: Oct. 09, 2022. [Online]. Available: https://cipotato.org/genebankcip/genebankcip/process/active_collection/

[78] “International Potato Center (CIP),” CGIAR Genebank Platform. Accessed: Oct. 06, 2022. [Online]. Available: https://www.genebanks.org/genebanks/international-potato-centre/

[79] “Genebank,” CIP Genebank. Accessed: Sep. 06, 2022. [Online]. Available: https://cipotato.org/genebankcip/genebankcip/

[80] “GRIN-Global: A Data Management Tool for All Genebanks.” Accessed: Sep. 06, 2022. [Online]. Available: https://www.croptrust.org/news-events/news/grin-global-a-data-management-tool-for-all-genebanks/

[81] “Data Management,” CIP Genebank. Accessed: Sep. 06, 2022. [Online]. Available: https://cipotato.org/genebankcip/genebankcip/data-management/

[82] H. D. Upadhyaya and R. Ortiz, “A mini core subset for capturing diversity and promoting utilization of chickpea genetic resources in crop improvement,” *Theor Appl Genet*, vol. 102, no. 8, pp. 1292–1298, Jun. 2001, doi: 10.1007/s00122-001-0556-y.

[83] A. Bari, K. Street, M. Mackay, D. T. F. Endresen, E. De Pauw, and A. Amri, “Focused identification of germplasm strategy (FIGS) detects wheat stem rust resistance linked to environmental variables,” *Genet Resour Crop Evol*, vol. 59, no. 7, pp. 1465–1481, Oct. 2012, doi: 10.1007/s10722-011-9775-5.

[84] "Primer to Analysis of Genomic Data Using R | SpringerLink." Accessed: Sep. 20, 2022. [Online]. Available: https://link.springer.com/book/10.1007/978-3-319-14475-7

[85] H. R. Johnston, B. J. B. Keats, and S. L. Sherman, "12 - Population Genetics," in *Emery and Rimoin's Principles and Practice of Medical Genetics and Genomics (Seventh Edition)*, R. E. Pyeritz, B. R. Korf, and W. W. Grody, Eds., Academic Press, 2019, pp. 359–373. doi: 10.1016/B978-0-12-812537-3.00012-3.

[86] D. L. Hartl, *A Primer of Population Genetics and Genomics*. Oxford University Press, 2020. doi: 10.1093/oso/9780198862291.001.0001.

[87] "phenotype / phenotypes | Learn Science at Scitable." Accessed: Oct. 10, 2022. [Online]. Available: https://www.nature.com/scitable/definition/phenotype-phenotypes-35/

[88] "Encyclopedia of Biodiversity | ScienceDirect." Accessed: Oct. 09, 2022. [Online]. Available: https://www.sciencedirect.com/referencework/9780123847201/encyclopedia-of-biodiversity

[89] "Diploid," Genome.gov. Accessed: Oct. 09, 2022. [Online]. Available: https://www.genome.gov/genetics-glossary/Diploid

[90] "Polygenic Trait," Genome.gov. Accessed: Oct. 09, 2022. [Online]. Available: https://www.genome.gov/genetics-glossary/Polygenic-Trait

[91] J. Zhang, R. Chiodini, A. Badr, and G. Zhang, "The impact of next-generation sequencing on genomics," *J Genet Genomics*, vol. 38, no. 3, pp. 95–109, Mar. 2011, doi: 10.1016/j.jgg.2011.02.003.

[92] "Hardy-Weinberg equilibrium | Learn Science at Scitable." Accessed: Oct. 10, 2022. [Online]. Available: http://www.nature.com/scitable/definition/hardy-weinberg-equilibrium-122

[93] V. Douhovnikoff and M. Leventhal, "The use of Hardy–Weinberg Equilibrium in clonal plant systems," *Ecol Evol*, vol. 6, no. 4, pp. 1173–1180, Jan. 2016, doi: 10.1002/ece3.1946.

[94] D. L. Hartl and A. G. Clark, *Principles of Population Genetics*. Sinauer Associates, 1997.

[95] K. Ritland, “Estimators for pairwise relatedness and individual inbreeding coefficients,” *Genetics Research*, vol. 67, no. 2, pp. 175–185, Apr. 1996, doi: 10.1017/S0016672300033620.

[96] I. S. C. Granato, G. Galli, E. G. de Oliveira Couto, M. B. e Souza, L. F. Mendonça, and R. Fritsche-Neto, “snpReady: a tool to assist breeders in genomic analysis,” *Mol Breeding*, vol. 38, no. 8, p. 102, Jul. 2018, doi: 10.1007/s11032-018-0844-8.

[97] M. Nei, “Analysis of gene diversity in subdivided populations,” *Proc Natl Acad Sci U S A*, vol. 70, no. 12, pp. 3321–3323, Dec. 1973, doi: 10.1073/pnas.70.12.3321.

[98] F. W. Allendorf, G. H. Luikart, and S. N. Aitken, *Conservation and the Genetics of Populations*. John Wiley & Sons, 2012.

[99] C. P. Andam, L. Challagundla, T. Azarian, W. P. Hanage, and D. A. Robinson, “3 - Population Structure of Pathogenic Bacteria,” in *Genetics and Evolution of Infectious Diseases (Second Edition)*, M. Tibayrenc, Ed., London: Elsevier, 2017, pp. 51–70. doi: 10.1016/B978-0-12-799942-5.00003-2.

[100] D. J. Lawson and D. Falush, “Population Identification Using Genetic Data,” *Annual Review of Genomics and Human Genetics*, vol. 13, no. 1, pp. 337–361, 2012, doi: 10.1146/annurev-genom-082410-101510.

[101] M. C. Fusté and M. C. Fusté, *Studies in Population Genetics*. 2012. doi: 10.5772/2152.

[102] S. Wright, “ISOLATION BY DISTANCE,” *Genetics*, vol. 28, no. 2, pp. 114–138, Mar. 1943, doi: 10.1093/genetics/28.2.114.

[103] S. Yadav *et al.*, “A linkage disequilibrium-based approach to position unmapped SNPs in crop species,” *BMC Genomics*, vol. 22, no. 1, p. 773, Oct. 2021, doi: 10.1186/s12864-021-08116-w.

[104] M. D. Teare and J. H. Barrett, “Genetic linkage studies,” *The Lancet*, vol. 366, no. 9490, pp. 1036–1044, Sep. 2005, doi: 10.1016/S0140-6736(05)67382-5.

[105] A. P. Ramakrishnan, “Linkage Disequilibrium,” in *Brenner’s Encyclopedia of Genetics (Second Edition)*, S. Maloy and K. Hughes, Eds., San Diego: Academic Press, 2013, pp. 252–253. doi: 10.1016/B978-0-12-374984-0.00870-6.

[106] J. W. Lau and P. J. Green, “Bayesian Model-Based Clustering Procedures,” *Journal of Computational and Graphical Statistics*, vol. 16, no. 3, pp. 526–558, Sep. 2007, doi: 10.1198/106186007X238855.

[107] “Structure Software for Population Genetics Inference.” Accessed: Oct. 18, 2022. [Online]. Available: https://web.stanford.edu/group/pritchardlab/structure.html

[108] D. Shriner, “Overview of Admixture Mapping,” *Curr Protoc Hum Genet*, vol. CHAPTER, p. Unit1.23, Jan. 2013, doi: 10.1002/0471142905.hg0123s76.

[109] G. Evanno, S. Regnaut, and J. Goudet, “Detecting the number of clusters of individuals using the software STRUCTURE: a simulation study,” *Mol Ecol*, vol. 14, no. 8, pp. 2611–2620, Jul. 2005, doi: 10.1111/j.1365-294X.2005.02553.x.

[110] “What Is High-Performance Computing (HPC)? | IBM.” Accessed: Mar. 18, 2024. [Online]. Available: https://www.ibm.com/topics/hpc

[111] T. Jombart and I. Ahmed, “adegenet 1.3-1: new tools for the analysis of genome-wide SNP data,” *Bioinformatics*, vol. 27, no. 21, pp. 3070–3071, Nov. 2011, doi: 10.1093/bioinformatics/btr521.

[112] T. Jombart and C. Collins, “Analysing genome-wide SNP data using adegenet 2.0.0”.

[113] I. T. Jolliffe and J. Cadima, “Principal component analysis: a review and recent developments,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, Apr. 2016, doi: 10.1098/rsta.2015.0202.

[114] H. Abdi and L. J. Williams, “Principal component analysis,” *WIREs Computational Statistics*, vol. 2, no. 4, pp. 433–459, 2010, doi: 10.1002/wics.101.

[115] “Using SVD for Dimensionality Reduction.” Accessed: Jun. 29, 2023. [Online]. Available: https://blogs.oracle.com/machinelearning/post/using-svd-for-dimensionality-reduction

[116] A. F. M. Alkarkhi and W. A. A. Alqaraghuli, “Chapter 10 - Discriminant Analysis and Classification,” in *Easy Statistics for Food Science with R*, A. F. M. Alkarkhi and W. A.

A. Alqaraghuli, Eds., Academic Press, 2019, pp. 161–175. doi: 10.1016/B978-0-12-814262-2.00010-8.

[117] J. Kim and H. Park, “Fast Nonnegative Matrix Factorization: An Active-Set-Like Method and Comparisons,” *SIAM J. Sci. Comput.*, vol. 33, no. 6, pp. 3261–3281, Jan. 2011, doi: 10.1137/110821172.

[118] A. Subasi, “Chapter 2 - Data preprocessing,” in *Practical Machine Learning for Data Analysis Using Python*, A. Subasi, Ed., Academic Press, 2020, pp. 27–89. doi: 10.1016/B978-0-12-821379-7.00002-3.

[119] I. B. Djordjevic, “Chapter 12 - Quantum machine learning,” in *Quantum Communication, Quantum Networks, and Quantum Sensing*, I. B. Djordjevic, Ed., Academic Press, 2023, pp. 491–561. doi: 10.1016/B978-0-12-822942-2.00010-8.

[120] J. A. Hartigan and M. A. Wong, “Algorithm AS 136: A K-Means Clustering Algorithm,” *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979, doi: 10.2307/2346830.

[121] G. Schwarz, “Estimating the Dimension of a Model,” *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[122] L. K. Nakhleh, N. A. Rosenberg, and T. Warnow, “PHYLOGENOMICS AND POPULATION GENOMICS: MODDELS, ALGORITHMS, AND ANALYTICAL TOOLS,” in *Biocomputing 2013*, Kohala Coast, Hawaii, USA: WORLD SCIENTIFIC, Nov. 2012, pp. 247–249. doi: 10.1142/9789814447973_0024.

[123] M. Nei, “Genetic Distance between Populations,” *The American Naturalist*, vol. 106, no. 949, pp. 283–292, 1972.

[124] N. Saitou and M. Nei, “The neighbor-joining method: a new method for reconstructing phylogenetic trees,” *Mol Biol Evol*, vol. 4, no. 4, pp. 406–425, Jul. 1987, doi: 10.1093/oxfordjournals.molbev.a040454.

[125] J. A. Studier and K. J. Keppler, “A note on the neighbor-joining algorithm of Saitou and Nei,” *Mol Biol Evol*, vol. 5, no. 6, pp. 729–731, Nov. 1988, doi: 10.1093/oxfordjournals.molbev.a040527.

[126] "Neighbor-joining method." Accessed: Oct. 22, 2022. [Online]. Available: http://www.deduveinstitute.be/~opperd/private/neighbor.html

[127] T. Mailund, G. S. Brodal, R. Fagerberg, C. N. Pedersen, and D. Phillips, "Recrafting the neighbor-joining method," *BMC Bioinformatics*, vol. 7, no. 1, p. 29, Jan. 2006, doi: 10.1186/1471-2105-7-29.

[128] M. Dunn, N. Burenhult, N. Kruspe, S. Tufvesson, and N. Becker, "Aslian linguistic prehistory: A case study in computational phylogenetics," *Diachronica*, vol. 28, pp. 291–323, Oct. 2011, doi: 10.1075/dia.28.3.01dun.

[129] C. D. Michener and R. R. Sokal, "A QUANTITATIVE APPROACH TO A PROBLEM IN CLASSIFICATION," *Evolution*, vol. 11, no. 2, pp. 130–162, Jun. 1957, doi: 10.1111/j.1558-5646.1957.tb02884.x.

[130] C. D. Cruz, C. C. Salgado, and L. L. Bhering, "Chapter 3 - Biometrics Applied to Molecular Analysis in Genetic Diversity," in *Biotechnology and Plant Breeding*, A. Borem and R. Fritsche-Neto, Eds., San Diego: Academic Press, 2014, pp. 47–81. doi: 10.1016/B978-0-12-418672-9.00003-9.

[131] P. S. Soltis and D. E. Soltis, "Applying the Bootstrap in Phylogeny Reconstruction," *Statistical Science*, vol. 18, no. 2, pp. 256–267, 2003.

[132] K. K. Ojha, S. Mishra, and V. K. Singh, "Chapter 5 - Computational molecular phylogeny: concepts and applications," in *Bioinformatics*, D. B. Singh and R. K. Pathak, Eds., Academic Press, 2022, pp. 67–89. doi: 10.1016/B978-0-323-89775-4.00025-0.

[133] S. Dray and A.-B. Dufour, "The ade4 Package: Implementing the Duality Diagram for Ecologists," *Journal of Statistical Software*, vol. 22, pp. 1–20, Sep. 2007, doi: 10.18637/jss.v022.i04.

[134] K. R. Jo *et al.*, "Analysis of genetic diversity and population structure among cultivated potato clones from Korea and global breeding programs," *Sci Rep*, vol. 12, no. 1, Art. no. 1, Jun. 2022, doi: 10.1038/s41598-022-12874-2.

[135] E. Paradis and K. Schliep, “ape 5.0: an environment for modern phylogenetics and evolutionary analyses in R,” *Bioinformatics*, vol. 35, no. 3, pp. 526–528, Feb. 2019, doi: 10.1093/bioinformatics/bty633.

[136] V. E. Chhatre and K. J. Emerson, “StrAuto: automation and parallelization of STRUCTURE analysis,” *BMC Bioinformatics*, vol. 18, no. 1, p. 192, Mar. 2017, doi: 10.1186/s12859-017-1593-0.

[137] Z. N. Kamvar, J. F. Tabima, and N. J. Grünwald, “Poppr: an R package for genetic analysis of populations with clonal, partially clonal, and/or sexual reproduction,” *PeerJ*, vol. 2, p. e281, Mar. 2014, doi: 10.7717/peerj.281.

[138] I. Letunic and P. Bork, “Interactive Tree Of Life (iTOL) v5: an online tool for phylogenetic tree display and annotation,” *Nucleic Acids Research*, vol. 49, no. W1, pp. W293–W296, Jul. 2021, doi: 10.1093/nar/gkab301.

[139] A. Raj, “fastStructure.” May 15, 2023. Accessed: Jun. 07, 2023. [Online]. Available: https://github.com/rajanil/fastStructure

[140] L. Clark, “lvclark/R_genetics_conv: R_genetics_conv 1.1.” Zenodo, Aug. 22, 2017. doi: 10.5281/zenodo.846816.

[141] R. M. Francis, “pophelper: an R package and web app to analyse and visualize population structure,” *Mol Ecol Resour*, vol. 17, no. 1, pp. 27–32, Jan. 2017, doi: 10.1111/1755-0998.12509.

[142] W. N. Venables and B. D. Ripley, *Modern Applied Statistics with S*. in Statistics and Computing. New York, NY: Springer, 2002. doi: 10.1007/978-0-387-21706-2.

[143] O. Francois, “Running Structure-like Population Genetic Analyses with R,” p. 9.

[144] “The Dataverse Project - Dataverse.org.” Accessed: Mar. 18, 2024. [Online]. Available: https://dataverse.org/home

[145] “Lect 4. Heterozygosity.” Accessed: Jun. 29, 2023. [Online]. Available: https://www.uwyo.edu/dbmcd/molmark/lect04/lect4.html

[146] "Logistic Prior · Issue #20 · rajanil/fastStructure," GitHub. Accessed: Jul. 03, 2023. [Online]. Available: https://github.com/rajanil/fastStructure/issues/20

[147] P. R. Peres-Neto, D. A. Jackson, and K. M. Somers, "How many principal components? stopping rules for determining the number of non-trivial axes revisited," *Computational Statistics & Data Analysis*, vol. 49, no. 4, pp. 974–997, Jun. 2005, doi: 10.1016/j.csda.2004.06.015.

[148] C. Fraley and A. E. Raftery, "Model-Based Clustering, Discriminant Analysis, and Density Estimation," *Journal of the American Statistical Association*, vol. 97, no. 458, pp. 611–631, Jun. 2002, doi: 10.1198/016214502760047131.

[149] S. M. Funk *et al.*, "Major inconsistencies of inferred population genetic structure estimated in a large set of domestic horse breeds using microsatellites," *Ecol Evol*, vol. 10, no. 10, pp. 4261–4279, Apr. 2020, doi: 10.1002/ece3.6195.

[150] I. M. Johnstone and A. Y. Lu, "Sparse Principal Components Analysis".

[151] G. I. Allen and P. O. Perry, "Singular Value Decomposition and High-Dimensional Data".

[152] M. Stift, F. Kolář, and P. G. Meirmans, "STRUCTURE is more robust than other clustering methods in simulated mixed-ploidy populations," *Heredity (Edinb)*, vol. 123, no. 4, pp. 429–441, Oct. 2019, doi: 10.1038/s41437-019-0247-6.

[153] T. Dou, C. Wang, Y. Ma, Z. Chen, J. Zhang, and G. Guo, "CoreSNP: an efficient pipeline for core marker profile selection from genome-wide SNP datasets in crops," *BMC Plant Biology*, vol. 23, no. 1, p. 580, Nov. 2023, doi: 10.1186/s12870-023-04609-w.

ANNEXES

ANNEX 1: Referential framework

The present investigation does not include a specific chapter for information regarding the legal, geographic, and historical background, given the relevant information has already been included in the introduction and theoretical framework. The pertinent geographic information regarding potato crops has been included in section 2.1. With respect to CIP and its institutional organization, necessary information has been included in the introduction and in section 2.2.

ANNEX 2: Libraries

```
library(dartR) # genetic diversity
library(adegenet) # population structure (dimensionality reduction)
library(snpReady) # genetic diversity
library(seqinr)
library(ade4) # population structure (distance-based)
library(ape) # tree export
library(poppr) # population structure
library(pegas)
library(tibble)
library(purrr)
library(data.table)
library(writexl)
library(readxl)
library(itol.toolkit)
library(StAMPP)
library(LEA) # population structure (dimensionality reduction)
library(MASS) # population structure (dimensionality reduction)
library(RColorBrewer)
library(viridis)
library(mclust)
library(devtools)
library(pophelper) # fastSTRUCTURE results visualization
library(gtools)
library(reshape2)
library(dplyr)
library(gridExtra)
library(rnaturalearth)
library(rnaturalearthdata)
library(ggpubr)

source("Functions.R")
# colours
clist <- list(
"shiny"=c("#1D72F5","#DF0101","#77CE61",
"#FF9326","#A945FF","#0089B2","#FDF060","#FFA6B2","#BFF217","#60D5FD","#C
C1577","#F2B950","#7FB21D","#EC496F","#326397","#B26314","#027368","#A4A4
A4","#610B5E"),
"strong"=c("#11A4C8","#63C2C5","#1D4F9F","#0C516D","#2A2771","#396D35","#
80C342","#725DA8","#B62025","#ED2224","#ED1943","#ED3995","#7E277C","#F7E
C16","#F8941E","#8C2A1C","#808080"),
"funky" = c("#A6CEE3", "#3B8ABE", "#72B29C", "#84C868",
"#4F9F3B","#EC9A91", "#E93E3F", "#F06C45", "#FDAC4F",  "#FB820F",
"#D1AAB7", "#8C66AF", "#A99099", "#EEDB80", "#B15928"))
```

ANNEX 3. Functions

```
# Nei's genetic distance (poppr nei's gen dist)
nei.dist.alt <- function(x){
  mat <- x
  idmat <- mat %*% t(mat)
  vec <- sqrt(diag(idmat))
  idmat <- idmat/vec[col(idmat)]
  idmat <- idmat/vec[row(idmat)]
  D <- -log(idmat)
  if (any(D %in% Inf)){
    D <- infinite_vals_replacement(D, warning)
  }
  D <- as.dist(D)
  return(D)
}

# Euclidean distance by chunks
euc.dist.chunks <- function(x){
  l <- seploc(x, n.block=10)
  lD <- lapply(l, function(e) dist(as.matrix(e)))
  D <- Reduce("+", lD)
  return(D)
}

# LEA write geno mod
write.geno.mod <- function(R, output.file)
{
  if(missing(R))
    stop("'R' argument is missing.")
  else if (!(is.matrix(R) || is.data.frame(R) || is.vector(R)))
    stop("'R' argument has to be of type matrix, data.frame or vector.")
  else if (is.vector(R))
    R = matrix(R,ncol=1,nrow=length(R))
  else if (is.data.frame(R))
    R = as.matrix(R)
  output.file = test_character("output.file", output.file, NULL)

  R[which(is.na(R))] = 9
  R[which(is.nan(R))] = 9

  write.table(t(R), output.file, col.names=FALSE,row.names=FALSE,sep="");
  return(output.file);
}

test_character <- function(name, param, default)
{
  if(missing(param)) {
    if(is.null(default)) {
      p = paste("'",name,"' argument is missing.", sep="");
      stop(p)
    } else
      return(default);
  } else {
    if(!is.character(param)) {
      p = paste("'",name,"' argument has to be of type character.",
                sep="");
      stop(p);
    }
  }
  return(param)
}

# Adegenet palette creator

.palette_parser <- function(inPAL, npop, pnames)
  {
  PAL <- try(match.fun(inPAL, descend = FALSE), silent = TRUE)
  if ("try-error" %in% class(PAL)){
    if (all(pnames %in% names(inPAL))){
      color <- inPAL[pnames]
    } else if (npop == length(inPAL)){
      color <- stats::setNames(inPAL, pnames)
    } else if (npop < length(inPAL)){
      warning("Number of populations fewer than number of colors
supplied. Discarding extra colors.")
      color <- stats::setNames(inPAL[1:npop], pnames)
    } else {
      warning("insufficient color palette supplied. Using funky().")
      color <- stats::setNames(funky(npop), pnames)
    }
  } else {
    color   <- stats::setNames(PAL(npop), pnames)
  }
  return(color)
}

numeric2structure <- function(genmat,
                              indNames = dimnames(genmat)[[1]],
                              addtlColumns = NULL, ploidy = 2,
                              exportMarkerNames = TRUE){
  nInd <- dim(genmat)[1] # number of individuals
  if(length(indNames) != nInd){
    stop("Number of individuals does not match between indNames and
genmat.")
  }
  if(!is.null(addtlColumns) && dim(addtlColumns)[1] != nInd){
    stop("Number of individuals does not match between addtlColumns and
genmat.")
  }
  genmat <- as.matrix(genmat)
  if(!all(genmat %in% c(0:ploidy,NA))){
    stop("genmat must only contain 0, 1, 2... ploidy and NA")
  }
  if(length(ploidy) != 1 || !is.numeric(ploidy)){
    stop("ploidy must be a single number")
  }

  # make sets of possible genotypes
  G <- list()
  for(i in 0:ploidy){
    G[[i + 1]] <- c(rep(1, ploidy - i), rep(2, i))
  }
  G[[ploidy + 2]] <- rep(-9, ploidy) # for missing data

  # set up data frame for Structure
  StructTab <- data.frame(ind = rep(indNames, each = ploidy))
  # add any additional columns
  if(!is.null(addtlColumns)){
    for(i in 1:dim(addtlColumns)[2]){
      StructTab <- data.frame(rep(addtlColumns[,i], each = ploidy),
StructTab)
    }
  }
  colnames(StructTab)[1:dim(addtlColumns)[2]] <- "#"

  # add genetic data
  for(i in 1:dim(genmat)[2]){
    thesegen <- genmat[,i] + 1
    thesegen[is.na(thesegen)] <- ploidy + 2
    StructTab[[dimnames(genmat)[[2]][i]]] <- unlist(G[thesegen])
  }

  return(StructTab)
}

kmeansBIC <- function(clust){

  m = ncol(clust$centers)
  n = length(clust$cluster)
  k = nrow(clust$centers)
  D = clust$tot.withinss
  return(D + log(n) * m * k)
}
```

ANNEX 4. Data characteristics plots

```
# SNP Map plot
ggplot(data = SNPMap) +
  geom_density(aes(x = Position), bw = 150000, linewidth = 0.4,  color =
"dodgerblue4") +
  labs(title = "Location of SNPs", x = "SNP Position", y = "Density",
color = "Chromosome") +
  scale_x_continuous(expand = c(0, 0), breaks = waiver(), n.breaks = 10)
+
  scale_y_continuous(expand = c(0, 0)) +
  geom_point(aes(x = Position, y = rep(0, nrow(SNPMap)), color =
factor(Chrom)), size = 2, shape = 3)  +
  scale_color_viridis(discrete = TRUE, option = "D", direction = 1) +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5), legend.position =
"bottom", legend.key.size = unit(0.3, 'cm'), legend.title =
element_text(size=10), legend.text = element_text(size=8)) +
  guides(color = guide_legend(ncol = 8, bycol = TRUE))

# saving plot
ggsave(filename = "Figures/SNPMap.png", device = "png")

# Sample Region Map plot
ggplot(data = ne_countries(scale = "medium", returnclass = "sf")) +
  geom_sf() +
  geom_point(data = data.frame(longitude = SampleInfo[,"Longitude of
collecting site"], latitude = SampleInfo[,"Latitude of collecting site"],
elevation = SampleInfo[,"Elevation of collecting site"]), aes(x =
longitude, y = latitude, color = elevation), size = 1,
        shape = 20) +
  scale_color_viridis(discrete = FALSE, option = "D", direction = -1) +
  coord_sf(xlim = c(min(SampleInfo$`Longitude of collecting site`, na.rm
= TRUE), max(SampleInfo$`Longitude of collecting site`, na.rm = TRUE)),
ylim = c(min(SampleInfo$`Latitude of collecting site`, na.rm = TRUE),
max(SampleInfo$`Latitude of collecting site`, na.rm = TRUE)), expand =
TRUE) +
  theme_classic() +
  labs(title = "Sample collection sites", x = "Longitude", y =
"Latitude") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), axis.line =
element_line(color = "black", linewidth = 0.4), panel.grid.major =
element_line(color="grey", size=0.2), axis.title = element_text(size =
10), legend.key.size = unit(0.5, 'cm'), legend.title = element_text(size
= 10), legend.text = element_text(size = 8))

# saving plot
ggsave(filename = "Figures/SampleRegionMap.png", device = "png")
```

ANNEX 5. Data set filtering

```
# import full data file
fullrawSNP <- read.csv2("Data/FullDataRaw.csv", header=T, sep=",",
check.names = FALSE)

preSNP <- fullrawSNP[, which(colMeans(fullrawSNP!="NC") > 0.95)] #
filtering from call rate

preSNP[preSNP == "NC"] <- NA

preSNP <- preSNP[which(rowMeans(is.na(preSNP)) < 0.1),] # filtering from
missing data

# write filtered data file
write.csv(preSNP, "DataImport/FullData.csv", row.names = TRUE)
```

ANNEX 6. Mantel test

```
# import full data
fullrawSNP <- read.csv2("Data/FullDataRaw.csv", header=T, sep=",",
check.names = FALSE)

# convert to matrix
fullmatSNP <- as.matrix(fullrawSNP)
rownames(fullmatSNP) <- fullmatSNP[,1]
fullmatSNP <- fullmatSNP[,-1]

fullmatSNP <- ifelse(fullmatSNP == "AAAA", as.numeric(0),fullmatSNP) #
homozygote AA
fullmatSNP <- ifelse(fullmatSNP == "AAAB", as.numeric(1) ,fullmatSNP) #
heterozygote
fullmatSNP <- ifelse(fullmatSNP == "AABB", as.numeric(2) ,fullmatSNP) #
heterozygote
fullmatSNP <- ifelse(fullmatSNP == "ABBB", as.numeric(3) ,fullmatSNP) #
heterozygote
fullmatSNP <- ifelse(fullmatSNP == "BBBB", as.numeric(4) ,fullmatSNP) #
homozygote BB

fullmatSNP <- matrix(as.numeric(fullmatSNP), ncol = ncol(fullmatSNP),
dimnames = list(rownames(fullmatSNP),colnames(fullmatSNP)))

# save as RDS object
saveRDS(fullmatSNP, file = "DataReady/MatrixNumericRaw.rds")

# generation of euclidean distance matrix from genlight object for raw
data
fullmatSNP <- readRDS(file = "DataReady/MatrixNumericRaw.rds")
glSNPraw <- new("genlight",fullmatSNP, indNames = rownames(fullmatSNP),
locNames = colnames(fullmatSNP), parallel=FALSE)

eucDistRaw <- euc.dist.chunks(glSNPraw)
saveRDS(eucDistRaw, file = "DataReady/eucDistRaw.rds")
```

```
# generation of euclidean distance matrix from genlight object for
filtered data
matnumSNP <- readRDS(file = "DataReady/MatrixNumeric.rds")
glSNP <- new("genlight", matnumSNP, indNames = rownames(matnumSNP),
locNames = colnames(matnumSNP), parallel=FALSE)

eucDist <- euc.dist.chunks(glSNP)
saveRDS(eucDist, file = "DataReady/eucDist.rds")

# mantel test
mantel <- mantel.rtest(as.dist(eucDist), as.dist(eucDistRaw), nrepet =
500)
saveRDS(mantel, file = "DataReady/mantel.rds")
mantel

# null hypothesis = unrelated (not representative)
# we can reject the null hypothesis because p-value < 0.05
```

ANNEX 7. Data import

```
# Filtered file import (accessions vs markers)
preSNP <- read.csv2("DataImport/FullData.csv", header=T, sep=",",
check.names = FALSE)

# All Sample map import
allSampleMap <- read_excel("Data/resumen.xlsx", sheet = "All_Sample Map",
col_names = T)
allSampleMap <- allSampleMap[allSampleMap$Name %like% "wp",]
allSampleMap <- allSampleMap[,-1] # delete index column

# Passport data
Passport <- read_excel("Data/resumen.xlsx", sheet = "Data Pasaporte",
col_names = T)
colnames(Passport)[which(names(Passport) == "Accession number")] <-
"CIPnumber"

# Sample information merged
SampleInfo <- merge(allSampleMap, Passport[, c("CIPnumber", "Country of
Origin", "Administrative subdivision 1", "Elevation of collecting site",
"Latitude of collecting site", "Longitude of collecting site")],
by="CIPnumber", all.x=TRUE)

# Ploidy formatting
colnames(SampleInfo)[which(names(SampleInfo) == "Ploidy_FlowCitometry
Result")] <- "Ploidy"
SampleInfo$Ploidy <- ifelse(SampleInfo$Ploidy == "2x?", "2x",
SampleInfo$Ploidy)
SampleInfo$Ploidy <- ifelse(SampleInfo$Ploidy == "3x?", "3x",
SampleInfo$Ploidy)
SampleInfo$Ploidy <- ifelse(SampleInfo$Ploidy == "4x?", "4x",
SampleInfo$Ploidy)
SampleInfo$Ploidy <- ifelse(SampleInfo$Ploidy == "5x?", "5x",
SampleInfo$Ploidy)
SampleInfo$Ploidy <- ifelse(SampleInfo$Ploidy == "6x?", "6x",
SampleInfo$Ploidy)
```

```
SampleInfo$Ploidy <- as.factor(SampleInfo$Ploidy)

# SNP Map import
SNPMap <- read.table('Data/SNP_Map.txt', sep = '\t', header = TRUE)

# Chromosome formatting
SNPMap$Chrom <- SNPMap$Chromosome
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "0", "CH00", SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH00", "CH00",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH01", "CH01",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH02", "CH02",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH03", "CH03",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH04", "CH04",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH05", "CH05",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH06", "CH06",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH07", "CH07",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH08", "CH08",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH09", "CH09",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH10", "CH10",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH11", "CH11",
SNPMap$Chrom)
SNPMap$Chrom <- ifelse(SNPMap$Chrom == "ST4.03CH12", "CH12",
SNPMap$Chrom)
```

ANNEX 8. Data frame to matrix conversion

```
# convert to matrix
matSNP <- as.matrix(preSNP) # change object type
rownames(matSNP) <- matSNP[,1]
matSNP <- matSNP[,-1]

matnumSNP <- ifelse(matSNP == "AAAA", as.numeric(0),matSNP) # homozygote
AA
matnumSNP <- ifelse(matnumSNP == "AAAB", as.numeric(1) ,matnumSNP) #
heterozygote
matnumSNP <- ifelse(matnumSNP == "AABB", as.numeric(2) ,matnumSNP) #
heterozygote
matnumSNP <- ifelse(matnumSNP == "ABBB", as.numeric(3) ,matnumSNP) #
heterozygote
matnumSNP <- ifelse(matnumSNP == "BBBB", as.numeric(4) ,matnumSNP) #
homozygote BB

matnumSNP <- matrix(as.numeric(matnumSNP), ncol = ncol(matnumSNP))
rownames(matnumSNP) <- rownames(matSNP)
colnames(matnumSNP) <- colnames(matSNP)

saveRDS(matnumSNP, file = "DataReady/MatrixNumeric.rds")
```

ANNEX 9. Genetic diversity analysis

```
# genetic diversity parameter calculation with snpReady package
# change diploid calls for heterozygosity calculations
matdiSNP <- ifelse(matnumSNP == 1 | matnumSNP == 2 | matnumSNP == 3,
as.numeric(1),matnumSNP)
matdiSNP <- ifelse(matnumSNP == 4, as.numeric(2),matdiSNP)
saveRDS(matdiSNP, file="DataReady/MatrixNumericDiploid.rds")

# import diploid matrix from file
matdiSNP <- readRDS(file="DataReady/MatrixNumericDiploid.rds") # direct
import

# genetic diversity parameters
SNPstats <- popgen(matdiSNP, plot=FALSE)

# diversity parameters by markers
SNPmarkers <- SNPstats[["whole"]][["Markers"]]
SNPmarkers <- rownames_to_column(SNPmarkers,"Marker")
SNPmarkers
mean(SNPstats[["whole"]][["Markers"]]$He)
mean(SNPstats[["whole"]][["Markers"]]$Ho)
mean(SNPstats[["whole"]][["Markers"]]$GD)

# diversity parameters by accessions
SNPaccessionsHo <- as.data.frame(SNPstats[["whole"]][["Genotypes"]][,1])
SNPaccessionsHo <- rownames_to_column(SNPaccessionsHo, "Accession")
colnames(SNPaccessionsHo)[2] <- "Observed heterozygosity"
SNPaccessionsHo

# writing result table files
write_xlsx(SNPmarkers,"Tables/MarkerStats.xlsx")
write_xlsx(SNPaccessionsHo,"Tables/AccessionStats.xlsx")

# PIC per markers plot
ggplot(SNPmarkers, aes(x = PIC)) +
  geom_histogram(aes(y = ..count..), binwidth = 0.025, boundary = 0,
color = "black", fill = "#A6CEE3") +
  scale_x_continuous(breaks = waiver(), n.breaks = 10) +
  scale_y_continuous(breaks = waiver(), n.breaks = 7) +
  theme_classic() +
  labs(x = "Polymorphic Information Content (PIC)", y = "Number of SNPs")

# saving plot
ggsave(filename = "Figures/PICPlot.png", device = "png")

## Diversity by species
# formatting
speciesData <- SNPaccessionsHo[SNPaccessionsHo$Accession %in%
SampleInfo$ID[!is.na(SampleInfo$spp)],]
colnames(speciesData) <- c('ID', 'Observed heterozygosity')
speciesData <- merge(speciesData, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
speciesData$Species <- as.factor(speciesData$spp)

# plot
ggplot(speciesData, aes(x=spp, y=`Observed heterozygosity`, color=spp)) +
  geom_boxplot(outlier.size = 0.5) +
  coord_flip() +
  labs(title ="Heterozygosity of accessions", x = "Species") +
  scale_color_viridis(discrete = TRUE, option = "D", direction = -1) +
  theme_classic() +
  theme(legend.position = "none", plot.title = element_text(hjust = 0.5,
size = 12), axis.ticks.y = element_blank(), axis.text.y =
element_blank(), axis.title = element_text(size = 10))

# saving plot
ggsave(filename = "Figures/SpeciesHo.png", device = "png")

## Diversity by ploidy
# formatting
ploidyData <- SNPaccessionsHo[SNPaccessionsHo$Accession %in%
SampleInfo$ID[!is.na(SampleInfo$Ploidy)],]
```

```
colnames(ploidyData) <- c('ID', 'Observed heterozygosity')
ploidyData <- merge(ploidyData, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)

# plot
ggplot(ploidyData, aes(x=Ploidy, y=`Observed heterozygosity`,
color=Ploidy)) +
  geom_boxplot(outlier.color = “black”) +
  coord_flip() +
  labs(title="Heterozygosity of samples") +
  scale_color_viridis(discrete = TRUE, option = "D", direction = -1) +
  theme_classic() +
  theme(legend.position = "none", plot.title = element_text(hjust = 0.5))

# saving plot
ggsave(filename = "Figures/PloidyHo.png", device = "png")
```

ANNEX 10. Bulk genotyping validation

```
# wp-1_761164
dist.gene(matnumSNP[rownames(matnumSNP) %like% "761164",], method =
"percentage")

# wp-174_762070
dist.gene(matnumSNP[rownames(matnumSNP) %like% "762070",], method =
"percentage")

# wp-201_760642
dist.gene(matnumSNP[rownames(matnumSNP) %like% "760642",], method =
"percentage")

# wp-237_765994
dist.gene(matnumSNP[rownames(matnumSNP) %like% "765994",], method =
"percentage")

# wp-287_761748
dist.gene(matnumSNP[rownames(matnumSNP) %like% "761748",], method =
"percentage")

# wp1281_761156
dist.gene(matnumSNP[rownames(matnumSNP) %like% "761156",], method =
"percentage")

# wp-9_760212
dist.gene(matnumSNP[rownames(matnumSNP) %like% "760212",], method =
"percentage")

# wp-120_763923
dist.gene(matnumSNP[rownames(matnumSNP) %like% "763923",], method =
"percentage")

# wp1269_761143
dist.gene(matnumSNP[rownames(matnumSNP) %like% "761143",], method =
"percentage")
```

ANNEX 11. Object creation for distance-based population structure analysis

```
matnumSNP <- readRDS(file = "DataReady/MatrixNumeric.rds")

glSNP <- new("genlight", matnumSNP, indNames = rownames(matnumSNP),
locNames = colnames(matnumSNP), parallel=FALSE)
glSNP

alleleFreq <- tab(glSNP, freq = TRUE)
saveRDS(alleleFreq, file="DataReady/Dendrogram/alleleFreq.rds")
```

ANNEX 12. NJ dendrogram and annotations

```
# nei genetic distance (matrix)
alleleFreq <- readRDS(file = "DataReady/Dendrogram/alleleFreq.rds") #from
allele frequency data

tNJ <- aboot(alleleFreq, tree = "nj", distance = nei.dist.alt, sample =
500) #bootstrap = 500

ape::write.tree(tNJ, file="DataReady/Dendrogram/NJ/NeiTreeNJ.txt")

# annotations with itol.toolkit package
tree <- read.tree("DataReady/Dendrogram/NJ/NeiTreeNJ.txt")

# Ploidy
unitPloidy <- create_unit(data = SampleInfo %>% select("ID", "Ploidy"),
key = "Ploidy", type = "DATASET_COLORSTRIP", color = "npg", tree = tree)
write_unit(unitPloidy, file =
"DataReady/Dendrogram/NJ/PloidyAnnotations.txt")

# Country
unitCountry <- create_unit(data = SampleInfo %>% select("ID", "Country of
Origin"), key = "Country", type = "DATASET_COLORSTRIP", color = "jco",
tree = tree)
write_unit(unitCountry, file =
"DataReady/Dendrogram/NJ/CountryAnnotations.txt")

# Species + City
LabelsDendrogram <- data.frame(id = SampleInfo$ID, new_label =
paste(SampleInfo$ID, SampleInfo$spp, SampleInfo$`Administrative
subdivision 1`))
unitSppCity <- create_unit(data = LabelsDendrogram, key = "Spp + City",
type = "LABELS", tree = tree)
write_unit(unitSppCity, file =
"DataReady/Dendrogram/NJ/SppCityAnnotations.txt")
```

ANNEX 13. UPGMA dendrogram and annotations

```
# nei genetic distance (matrix)
alleleFreq <- readRDS(file = "DataReady/Dendrogram/alleleFreq.rds")

tUPGMA <- aboot(alleleFreq, tree = "upgma", distance = nei.dist.alt,
sample = 500) #bootstrap = 500

ape::write.tree(tUPGMA,
file="DataReady/Dendrogram/UPGMA/NeiTreeUPGMA.txt")
# annotations with itol.toolkit package
tree <- read.tree("DataReady/Dendrogram/UPGMA/NeiTreeUPGMA.txt")

# Ploidy
unitPloidy <- create_unit(data = SampleInfo %>% select("ID", "Ploidy"),
key = "Ploidy", type = "DATASET_COLORSTRIP", color = "npg", tree = tree)
write_unit(unitPloidy, file =
"DataReady/Dendrogram/UPGMA/PloidyAnnotations.txt")

# Country
unitCountry <- create_unit(data = SampleInfo %>% select("ID", "Country of
Origin"), key = "Country", type = "DATASET_COLORSTRIP", color = "jco",
tree = tree)
write_unit(unitCountry, file =
"DataReady/Dendrogram/UPGMA/CountryAnnotations.txt")

# Species + City
LabelsDendrogram <- data.frame(id = SampleInfo$ID, new_label =
paste(SampleInfo$ID, SampleInfo$spp, SampleInfo$`Administrative
subdivision 1`))
unitSppCity <- create_unit(data = LabelsDendrogram, key = "Spp + City",
type = "LABELS", tree = tree)
write_unit(unitSppCity, file =
"DataReady/Dendrogram/UPGMA/SppCityAnnotations.txt")
```

ANNEX 14. Structure file conversion

```
# fastStructure file format conversion
# from diploid matrix
matdiSNPrev <- readRDS(file = "DataReady/MatrixNumericDiploid.rds")
accessionsRev <- allSampleMap$ID[allSampleMap$`Seleccion
structure`=="si"]
matdiSNPrev <- matdiSNPrev[rownames(matdiSNPrev) %in% accessionsRev,]

# creating structure file
structureFile <- numeric2structure(matdiSNPrev, indNames =
rownames(matdiSNPrev), exportMarkerNames = TRUE, addtlColumns =
matrix("#", nrow = nrow(matdiSNPrev), ncol = 5,
dimnames=list(NULL,c("#","#","#","#","#"))))
colnames(structureFile)[1:5] <- "#"

# export all data
write.table(structureFile, file = "DataReady/SNPdata.str", row.names =
FALSE, col.names = FALSE, append = FALSE, sep = "\t", quote = FALSE)
```

ANNEX 15. fastSTRUCTURE Installation and Execution

```
# install dependencies
python --version
python -m pip --version
pip install numpy==1.16.0
pip install numpy==1.16.5
pip install cython==0.27.3
pip install scipy==1.2.1
wget ftp://ftp.gnu.org/gnu/gsl/gsl-1.9.tar.gz
tar -xf gsl-1.9.tar.gz
cd gsl-1.9
./configure --prefix=/Users/tamaraortiz/gsl-1.9
make
make check
make install

# building python extensions
export CPATH=/Users/tamaraortiz/gsl-1.9/include/
export LIBRARY_PATH=/Users/tamaraortiz/gsl-1.9/lib/
export LD_LIBRARY_PATH=/Users/tamaraortiz/gsl-1.9/lib/:$LD_LIBRARY_PATH
export CFLAGS=-I/Users/tamaraortiz/gsl-1.9/include

# get source code
pwd
cd /Users/tamaraortiz
mkdir fastStructure
cd fastStructure
wget --no-check-certificate
https://github.com/rajanil/fastStructure/archive/master.tar.gz
tar -xf master.tar.gz
cd /Users/tamaraortiz
cd ~/fastStructure/fastStructure-master/vars
python setup.py build_ext --inplace
cd ~/fastStructure/fastStructure-master/
python setup.py build_ext --inplace
```

```
# testing the code
python structure.py

python structure.py -K 3 --input=test/testdata --output=testoutput_simple
--full --seed=100
ls test/testoutput_simple*

# executing the code
for k in `seq 15`; do python structure.py -K $k --input=data/SNPdata --
output=data/output_log --full --seed=100 --prior=logistic --format=str;
done
for k in `seq 15`; do python structure.py -K $k --input=data/SNPdata --
output=data/output_sim --full --seed=100 --prior=simple --format=str;
done

# choosing optimum number of K
python chooseK.py --input=data/output_log
python chooseK.py –-input=data/output_sim
```

ANNEX 16. fastSTRUCTURE simple prior plots

```
# import files
fsfilesSim <- list.files(path="DataReady/fastSTRUCTURE/sim", full.names =
TRUE, all.files = FALSE, pattern = "meanQ")
fslistSim <- readQ(files = fsfilesSim)
fslistSim <- fslistSim[order(nchar(fslistSim), fslistSim)]
fslistSim <- lapply(fslistSim, "rownames<-", rownames(matnumSNPrev))

# plot with all K values
plotQ(alignK(fslistSim), exportpath = "DataReady/fastSTRUCTURE/Plots",
imgoutput = "join", returnplot = TRUE, exportplot = FALSE, clustercol =
clist$funky)

# data formatting for plot
Krange <- 11:14 # user has to define the subpopulation range
fslistFSim <- fslistSim[c(Krange)]

# DAPC data frame for plot
fsSimtemp <- as.data.frame(fslistFSim[[1]])
colnames(fsSimtemp) <- c(1:Krange[1])
fsSimtemp$K <- Krange[1]
fsSimtemp$ID <- rownames(fsSimtemp)
fsSimtemp <- merge(fsSimtemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
fsSimtemp <- merge(fsSimtemp, SampleInfo[, c("ID","Country of Origin")],
by="ID", all.x=TRUE)
fsSimtemp <- merge(fsSimtemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
fsSimtemp <- merge(fsSimtemp, SampleInfo[, c("ID","Administrative
subdivision 1")], by="ID", all.x=TRUE)
fsSimtemp <- melt(fsSimtemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
fsSimDF <- fsSimtemp

for(i in 2:length(fslistFSim)){
  fsSimtemp <- as.data.frame(fslistFSim[[i]])
  colnames(fsSimtemp) <- c(1:Krange[i])
  fsSimtemp$K <- Krange[i]
  fsSimtemp$ID <- rownames(fsSimtemp)
  fsSimtemp <- merge(fsSimtemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
  fsSimtemp <- merge(fsSimtemp, SampleInfo[, c("ID","Country of
Origin")], by="ID", all.x=TRUE)
  fsSimtemp <- merge(fsSimtemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
  fsSimtemp <- merge(fsSimtemp, SampleInfo[, c("ID","Administrative
subdivision 1")], by="ID", all.x=TRUE)
  fsSimtemp <- melt(fsSimtemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
  fsSimDF <- rbind(fsSimDF, fsSimtemp)
}

grp.labs <- paste("K =", Krange)
names(grp.labs) <- Krange

# Composition plot
ggplot(fsSimDF, aes(x = ID, y = Probability, fill = Group)) +
  geom_bar(stat = "identity") +
  facet_grid(rows = vars(K), scales = "free_x", space = "free", labeller
= labeller(K = grp.labs)) +
  scale_fill_manual(values=clist$funky) +
  labs(title = "fastSTRUCTURE Assignment plot w/ simple prior", y =
"Membership probability") +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x =
element_blank(), axis.title.x = element_blank(), axis.ticks.x =
element_blank(), axis.text.y = element_text(size = 8))

# saving plot
ggsave(filename = "Figures/fsSimK11-14.png", device = "png", width = 7,
height = 4.5)
```

ANNEX 17. fastSTRUCTURE logistic prior plots

```
# import files
fsfilesLog <- list.files(path="DataReady/fastSTRUCTURE/log", full.names =
TRUE, all.files = FALSE, pattern = "meanQ")
fslistLog <- readQ(files = fsfilesLog)
fslistLog <- fslistLog[order(nchar(fslistLog), fslistLog)]
fslistLog <- lapply(fslistLog, "rownames<-", rownames(matnumSNPrev))

# plot with all K values
plotQ(alignK(fslistFas), exportpath = "DataReady/fastSTRUCTURE/Plots",
imgoutput = "join", returnplot = TRUE, exportplot = FALSE, clustercol =
clist$funky)

# data formatting for plot
Krange <- 11:14 #user has to define the subpopulation range
fslistFLog <- fslistLog[c(Krange)]

# DAPC data frame for plot
fsLogtemp <- as.data.frame(fslistFLog[[1]])
colnames(fsLogtemp) <- c(1:Krange[1])
fsLogtemp$K <- Krange[1]
fsLogtemp$ID <- rownames(fsLogtemp)
fsLogtemp <- merge(fsLogtemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
fsLogtemp <- merge(fsLogtemp, SampleInfo[, c("ID","Country of Origin")],
by="ID", all.x=TRUE)
fsLogtemp <- merge(fsLogtemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
fsLogtemp <- merge(fsLogtemp, SampleInfo[, c("ID","Administrative
subdivision 1")], by="ID", all.x=TRUE)
fsLogtemp <- melt(fsLogtemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
fsLogDF <- fsLogtemp

for(i in 2:length(fslistFLog)){
  fsLogtemp <- as.data.frame(fslistFLog[[i]])
  colnames(fsLogtemp) <- c(1:Krange[i])
  fsLogtemp$K <- Krange[i]
  fsLogtemp$ID <- rownames(fsLogtemp)
  fsLogtemp <- merge(fsLogtemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
  fsLogtemp <- merge(fsLogtemp, SampleInfo[, c("ID","Country of
Origin")], by="ID", all.x=TRUE)
  fsLogtemp <- merge(fsLogtemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
  fsLogtemp <- merge(fsLogtemp, SampleInfo[, c("ID","Administrative
subdivision 1")], by="ID", all.x=TRUE)
  fsLogtemp <- melt(fsLogtemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
  fsLogDF <- rbind(fsLogDF, fsLogtemp)
}

grp.labs <- paste("K =", Krange)
names(grp.labs) <- Krange

# Composition plot
ggplot(fsLogDF, aes(x = ID, y = Probability, fill = Group)) +
  geom_bar(stat = "identity") +
  facet_grid(rows = vars(K), scales = "free_x", space = "free", labeller
= labeller(K = grp.labs)) +
  scale_fill_manual(values=clist$funky) +
  labs(title = "fastSTRUCTURE Assignment plot w/ logistic prior", y =
"Membership probability") +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x =
element_blank(), axis.title.x = element_blank(), axis.ticks.x =
element_blank(), axis.text.y = element_text(size = 8))

# saving plot
ggsave(filename = "Figures/fsLogK11-14.png", device = "png", width = 7,
height = 4.5)
```

ANNEX 18. DAPC

```
# PCA
# PCA from genlight object
PCA <- glPca(glSNPrev, parallel = TRUE, nf = NULL)
saveRDS(PCA, file = "DataReady/PCA.rds")

# PCA from saved object
PCA <- readRDS(file = "DataReady/PCA.rds")

# Explained variation by principal component
PCAvariation <- data.frame(PC = 1:length(PCA[["eig"]]), Variation =
(PCA[["eig"]]/sum(PCA[["eig"]])) * 100, CumulativeVar =
cumsum((PCA[["eig"]]/sum(PCA[["eig"]])) * 100))

# scree plot
ggplot(data = PCAvariance[1:75,], aes(x = PC, y = Variance)) +
  geom_line(size = 0.25) +
  geom_point(shape = 21) +
  geom_vline(xintercept = 50, linetype = "dotted", color="blue") +
  labs(title = "Scree plot", x = "Principal component", y = "Variance
explained (%)") +
  scale_x_continuous(expand = c(0, 1)) +
  scale_y_continuous(expand = c(0, 1)) +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5))

# saving plot
ggsave(filename = "Figures/DAPCPCAVar.png", device = "png", width = 7,
height = 4)

# K-means clustering
# run K-means
maxKclusters <- 20

DAPCBIC <- find.clusters(glSNPrev, n.pca = 50, stat = "BIC",
choose.n.clust = FALSE, max.n.clust = maxKclusters, glPca = PCA)
```

```
saveRDS(DAPCBIC, file = "DataReady/DAPC/KstatDAPC.rds")

# read K-means
DAPCBIC <- readRDS(file = "DataReady/DAPC/KstatDAPC.rds")

# BIC value plot
ggplot(data = as.data.frame(DAPCBIC$Kstat), aes(x = 1:20, y =
DAPCBIC$Kstat)) +
  geom_point(shape = 21, color = "blue") +
  theme_classic() +
  labs(title = "Detection based on BIC", x = "Number of clusters (K)", y
= "BIC") +
  theme(plot.title = element_text(hjust = 0.5))

# saving plot
ggsave(filename = "Figures/DAPCBIC.png", device = "png", width = 7,
height = 4)

# DA
# DAPC for K numbers 11 to 14
Krange <- 11:14 #user has to define the subpopulation range
DAPCgrouplist <- vector(mode = "list", length = length(Krange))
DAPC <- vector(mode = "list", length = length(Krange))

for(i in 1:length(DAPC)){
  set.seed(10)
  DAPCgrouplist[[i]] <- find.clusters(glSNPrev, n.pca = 50, n.clust =
Krange[i], glPca = PCA, n.iter = 1000)
  DAPC[[i]] <- dapc(glSNPrev, pop = DAPCgrouplist[[i]]$grp, n.pca = 50,
n.da = 5, glPca = PCA, parallel = TRUE, var.contrib = TRUE, var.loadings
= TRUE)
}

# Exploring how much variability is explained for each linear function
DAPCvar <- vector(mode = "list", length = length(DAPC))
```

```
for(i in 1:length(DAPCvar)){
  DAPCvar[[i]] <- data.frame(DA = 1:length(DAPC[[i]][["eig"]]), Variance
= (DAPC[[i]][["eig"]] / sum(DAPC[[i]][["eig"]])) * 100, CumulativeVar =
cumsum((DAPC[[i]][["eig"]] / sum(DAPC[[i]][["eig"]])) * 100))
}

# Plotting variability and linear functions
ggplot(data = DAPCvar[[3]], aes(x = DA, y = Variance)) +
  geom_line(size = 0.25) +
  geom_point(shape = 21) +
  geom_vline(xintercept = 5, linetype = "dotted", color="blue") +
  labs(title = "Discriminant analysis eigenvalues", x = "Linear
discriminant", y = "Variance explained (%)") +
  scale_x_continuous(expand = c(0, 0.1), breaks =
c(1,2,3,4,5,6,7,8,9,10)) +
  scale_y_continuous(expand = c(0, 1)) +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5))

# saving plot
ggsave(filename = "Figures/DAPCDAVar.png", device = "png", width = 7,
height = 4)

# DAPC data frame for plot
DAPCtemp <- as.data.frame(DAPC[[1]]$posterior)
DAPCtemp$K <- Krange[1]
DAPCtemp$ID <- rownames(DAPCtemp)
DAPCtemp <- merge(DAPCtemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
DAPCtemp <- merge(DAPCtemp, SampleInfo[, c("ID","Country of Origin")],
by="ID", all.x=TRUE)
DAPCtemp <- merge(DAPCtemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
DAPCtemp <- merge(DAPCtemp, SampleInfo[, c("ID","Administrative
subdivision 1")], by="ID", all.x=TRUE)
```

```
DAPCtemp <- melt(DAPCtemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
DAPCDF <- DAPCtemp

for(i in 2:length(DAPC)){
  DAPCtemp <- as.data.frame(DAPC[[i]]$posterior)
  DAPCtemp$K <- Krange[i]
  DAPCtemp$ID <- rownames(DAPCtemp)
  DAPCtemp <- merge(DAPCtemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
  DAPCtemp <- merge(DAPCtemp, SampleInfo[, c("ID","Country of Origin")],
by="ID", all.x=TRUE)
  DAPCtemp <- merge(DAPCtemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
  DAPCtemp <- merge(DAPCtemp, SampleInfo[, c("ID","Administrative
subdivision 1")], by="ID", all.x=TRUE)
  DAPCtemp <- melt(DAPCtemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
  DAPCDF <- rbind(DAPCDF, DAPCtemp)
}

grp.labs <- paste("K =", Krange)
names(grp.labs) <- Krange

# Composition plot
ggplot(DAPCDF, aes(x = ID, y = Probability, fill = Group)) +
  geom_bar(stat = "identity") +
  facet_grid(rows = vars(K), scales = "free_x", space = "free", labeller
= labeller(K = grp.labs)) +
  scale_fill_manual(values=clist$funky) +
  labs(title = "DAPC Assignment plot", y = "Membership probability") +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x =
element_blank(), axis.title.x = element_blank(), axis.ticks.x =
element_blank())
```

```
# saving plot
ggsave(filename = "Figures/DAPCK11-14.png", device = "png", width = 7,
height = 4.5)

# Map plots
# Sample Region Map plot K = 13 Group = 8
DAPCG8Map <- ggplot(data = ne_countries(scale = "medium", returnclass =
"sf")) +
  geom_sf() +
  geom_point(data = data.frame(longitude = SampleInfo[SampleInfo$ID %in%
DAPCDF$ID[DAPCDF$K == 13 & DAPCDF$Group == 8 & DAPCDF$Probability >
0.90],"Longitude of collecting site"], latitude =
SampleInfo[SampleInfo$ID %in% DAPCDF$ID[DAPCDF$K == 13 & DAPCDF$Group ==
8 & DAPCDF$Probability > 0.90], "Latitude of collecting site"], elevation
= SampleInfo[SampleInfo$ID %in% DAPCDF$ID[DAPCDF$K == 13 & DAPCDF$Group
== 8 & DAPCDF$Probability > 0.90], "Elevation of collecting site"]),
aes(x = longitude, y = latitude, color = elevation), size = 1,
        shape = 20) +
  scale_color_viridis(discrete = FALSE, option = "D", direction = -1) +
  coord_sf(xlim = c(min(SampleInfo$`Longitude of collecting site`, na.rm
= TRUE), max(SampleInfo$`Longitude of collecting site`, na.rm = TRUE)),
ylim = c(min(SampleInfo$`Latitude of collecting site`, na.rm = TRUE),
max(SampleInfo$`Latitude of collecting site`, na.rm = TRUE)), expand =
TRUE) +
  theme_classic() +
  labs(title = "Sample collection sites for cluster 8 in K = 13", x =
"Longitude", y = "Latitude") +
  theme(plot.title = element_text(hjust = 0.5, size = 10), axis.line =
element_line(color = "black", linewidth = 0.4), panel.grid.major =
element_line(color="grey", size=0.2), axis.title = element_text(size =
10), legend.key.size = unit(0.4, 'cm'), legend.title = element_text(size
= 8), legend.text = element_text(size = 6))

# Sample Region Map plot K = 13 Group = 4
DAPCG4Map <- ggplot(data = ne_countries(scale = "medium", returnclass =
"sf")) +
  geom_sf() +
  geom_point(data = data.frame(longitude = SampleInfo[SampleInfo$ID %in%
DAPCDF$ID[DAPCDF$K == 13 & DAPCDF$Group == 4 & DAPCDF$Probability >
0.90],"Longitude of collecting site"], latitude =
SampleInfo[SampleInfo$ID %in% DAPCDF$ID[DAPCDF$K == 13 & DAPCDF$Group ==
4 & DAPCDF$Probability > 0.90], "Latitude of collecting site"], elevation
= SampleInfo[SampleInfo$ID %in% DAPCDF$ID[DAPCDF$K == 13 & DAPCDF$Group
== 4 & DAPCDF$Probability > 0.90], "Elevation of collecting site"]),
aes(x = longitude, y = latitude, color = elevation), size = 1,
        shape = 20) +
  scale_color_viridis(discrete = FALSE, option = "D", direction = -1) +
  coord_sf(xlim = c(min(SampleInfo$`Longitude of collecting site`, na.rm
= TRUE), max(SampleInfo$`Longitude of collecting site`, na.rm = TRUE)),
ylim = c(min(SampleInfo$`Latitude of collecting site`, na.rm = TRUE),
max(SampleInfo$`Latitude of collecting site`, na.rm = TRUE)), expand =
TRUE) +
  theme_classic() +
  labs(title = "Sample collection sites for cluster 4 in K = 13", x =
"Longitude", y = "Latitude") +
  theme(plot.title = element_text(hjust = 0.5, size = 10), axis.line =
element_line(color = "black", linewidth = 0.4), panel.grid.major =
element_line(color="grey", size=0.2), axis.title = element_text(size =
10), legend.key.size = unit(0.4, 'cm'), legend.title = element_text(size
= 8), legend.text = element_text(size = 6))

#arranged plot
ggarrange(ggarrange(DAPCG8Map, DAPCG4Map, ncol = 2, labels = c("A",
"B")))

#saving plot
ggsave(filename = "Figures/DAPCG8G4.png", device = "png", width = 7,
height = 4)
```

ANNEX 19. SVD with DA

```
# SVD
# matrix centering
matrixCentered <- scale(matnumSNPrev, center = TRUE, scale = FALSE)

# mean imputation for NA values
for(i in 1:ncol(matrixCentered)) {
  matrixCentered[ , i][is.na(matrixCentered[ , i])] <-
mean(matrixCentered[ , i], na.rm = TRUE)
}

# SVD
SVD <- svd(matrixCentered)
saveRDS(SVD, file = "DataReady/SVD/SVD.rmd")

SVD <- readRDS(file = "DataReady/SVD/SVD.rmd")

SVDvariance <- data.frame(SV = 1:length(SVD$d), Variance =
SVD$d^2/sum(SVD$d^2) * 100, CumulativeVar = cumsum(SVD$d^2/sum(SVD$d^2))
* 100)

# singular value variance plot
ggplot(data = SVDvariance[1:75,], aes(x = SV, y = Variance)) +
  geom_line(size = 0.25) +
  geom_point(shape = 21) +
  geom_vline(xintercept = 50, linetype = "dotted", color="blue") +
  labs(title = "Explained variance plot", x = "Singular value", y =
"Variance explained (%)") +
  scale_x_continuous(expand = c(0, 1)) +
  scale_y_continuous(expand = c(0, 1)) +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5))

# saving plot
ggsave(filename = "Figures/SVDVar.png", device = "png", width = 7, height
= 4)
```

```
# reducing data size with 50 singular values
SVDrec <- (SVD$u[,1:50] %*% diag(SVD$d[1:50]))
rownames(SVDrec) <- rownames(matnumSNPrev)

# K-means clustering
SVDclusters <- vector(mode = "list", length = length(1:20))
KstatSVDMatrix <- data.frame(matrix(nrow = 20, ncol = 2))
colnames(KstatSVDMatrix) <- c("K", "BIC")

for(i in 1:20){
  set.seed(10)
  SVDclusters[[i]] <- kmeans(SVDrec, centers = i)
  KstatSVDMatrix$K[i] <- i
  KstatSVDMatrix$BIC[i] <- kmeansBIC(SVDclusters[[i]])
}

# BIC plot
ggplot(data = KstatSVDMatrix, aes(x = K, y = BIC)) +
  geom_point(shape = 21, color = "blue") +
  theme_classic() +
  labs(title = "Detection based on BIC", x = "Number of clusters (K)", y
= "BIC") +
  theme(plot.title = element_text(hjust = 0.5))

# saving plot
ggsave(filename = "Figures/SVDDABIC.png", device = "png", width = 7,
height = 4)

# DA
# DA for K numbers 9 to 13
Krange <- 9:13 # user has to define the subpopulation range
SVDDA <- vector(mode = "list", length = length(Krange))

for(i in 1:length(SVDDA)){
  set.seed(10)
  SVDDA[[i]] <- lda(SVDrec, grouping = SVDclusters[[Krange[i]]]$cluster)
  SVDDA[[i]]$posterior <- predict(SVDDA[[i]], dimen = 5)
  SVDDA[[i]]$eig <- SVDDA[[i]]$svd^2
}

# Exploring how much variability is explained for each linear function
SVDDAvar <- vector(mode = "list", length = length(SVDDA))

for(i in 1:length(SVDDAvar)){
  SVDDAvar[[i]] <- data.frame(DA = 1:length(SVDDA[[i]][["eig"]]),
Variance = (SVDDA[[i]][["eig"]] / sum(SVDDA[[i]][["eig"]])) * 100,
CumulativeVar = cumsum((SVDDA[[i]][["eig"]] / sum(SVDDA[[i]][["eig"]])) *
100))
}

# Plotting variability and linear functions
ggplot(data = SVDDAvar[[3]], aes(x = DA, y = Variance)) +
  geom_line(size = 0.25) +
  geom_point(shape = 21) +
  geom_vline(xintercept = 5, linetype = "dotted", color="blue") +
  labs(title = "Discriminant analysis eigenvalues", x = "Linear
discriminant", y = "Variance explained (%)") +
  scale_x_continuous(expand = c(0, 0.1), breaks =
c(1,2,3,4,5,6,7,8,9,10)) +
  scale_y_continuous(expand = c(0, 1)) +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5))

# saving plot
ggsave(filename = "Figures/SVDDAVar.png", device = "png", width = 7,
height = 4)

# data formatting for plot
Krange <- 9:13 #user has to define the subpopulation range

# SVDDA data frame for plot
SVDDAtemp <- as.data.frame(SVDDA[[1]]$posterior$posterior)
SVDDAtemp$K <- Krange[1]
SVDDAtemp$ID <- rownames(SVDDAtemp)
```

```
SVDDAtemp <- merge(SVDDAtemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
SVDDAtemp <- merge(SVDDAtemp, SampleInfo[, c("ID","Country of Origin")],
by="ID", all.x=TRUE)
SVDDAtemp <- merge(SVDDAtemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
SVDDAtemp <- merge(SVDDAtemp, SampleInfo[, c("ID","Administrative
subdivision 1")], by="ID", all.x=TRUE)
SVDDAtemp <- melt(SVDDAtemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
SVDDAdf <- SVDDAtemp

for(i in 2:length(SVDDA)){
  SVDDAtemp <- as.data.frame(SVDDA[[i]]$posterior$posterior)
  SVDDAtemp$K <- Krange[i]
  SVDDAtemp$ID <- rownames(SVDDAtemp)
  SVDDAtemp <- merge(SVDDAtemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
  SVDDAtemp <- merge(SVDDAtemp, SampleInfo[, c("ID","Country of
Origin")], by="ID", all.x=TRUE)
  SVDDAtemp <- merge(SVDDAtemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
  SVDDAtemp <- merge(SVDDAtemp, SampleInfo[, c("ID","Administrative
subdivision 1")], by="ID", all.x=TRUE)
  SVDDAtemp <- melt(SVDDAtemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
  SVDDAdf <- rbind(SVDDAdf, SVDDAtemp)
}

grp.labs <- paste("K =", Krange)
names(grp.labs) <- Krange

# Composition plot
ggplot(SVDDAdf, aes(x = ID, y = Probability, fill = Group)) +
  geom_bar(stat = "identity") +
  facet_grid(rows = vars(K), scales = "free_x", space = "free", labeller
= labeller(K = grp.labs)) +
  scale_fill_manual(values=clist$funky) +
  labs(title = "SVD + DA Assignment plot", y = "Membership probability")
+
  theme(plot.title = element_text(hjust = 0.5), axis.text.x =
element_blank(), axis.title.x = element_blank(), axis.ticks.x =
element_blank())

# saving plot
ggsave(filename = "Figures/SVDDAK9-13.png", device = "png", width = 7,
height = 4.5)
```

ANNEX 20. sNMF

```
# creating object
write.geno.mod(matnumSNPrev, "DataReady/LEA/LEAgeno.geno")

# snmf for 1-20 K
# import from file
snmfObjectKvar <- snmf("DataReady/LEA/LEAgeno.geno", K=1:20, ploidy = 4,
alpha = 100, entropy = TRUE, project = "new")
saveRDS(snmfObjectKvar, file="DataReady/LEA/snmfLEAKvar.rds")

# cross entropy data
snmfObjectKvar <- readRDS("DataReady/LEA/snmfLEAKvar.rds")
snmfCrossEntr <- data.frame(matrix(nrow = 20, ncol = 2)) #20 rows for the
20 Kmax clusters

for(i in 1:20){
  snmfCrossEntr[i,1] <- i
  snmfCrossEntr[i,2] <- snmfObjectKvar@runs[[i]]@crossEntropy
}

colnames(snmfCrossEntr) <- c("K", "Cross-entropy")

# plot
ggplot(data = snmfCrossEntr, aes(x = K, y = `Cross-entropy`)) +
  geom_point(shape = 21, color = "blue") +
  scale_x_continuous(breaks=seq(0, 20, 2)) +
  theme_classic() +
  labs(title = "Detection based on cross-entropy", x = "Number of
clusters (K)", y = "Cross-entropy") +
  theme(plot.title = element_text(hjust = 0.5))

# saving plot
ggsave(filename = "Figures/NMFCrossEnt.png", device = "png", width =
7.29, height = 4.91)

# Extracting data for K range
```

```
Krange <- 10:14 #user has to define the subpopulation range
NMFmatrix <- vector(mode = "list", length = length(Krange))

for(i in 1:length(NMFmatrix)){
  NMFmatrix[[i]] <- Q(snmfObjectKvar, K = Krange[i])
}

# NMF data frame
NMFTemp <- as.data.frame(NMFmatrix[[1]])
names(NMFTemp) <- 1:Krange[1]
NMFTemp$K <- Krange[1]
NMFTemp$ID <- rownames(matnumSNPrev)
NMFTemp <- merge(NMFTemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
NMFTemp <- merge(NMFTemp, SampleInfo[, c("ID","Country of Origin")],
by="ID", all.x=TRUE)
NMFTemp <- merge(NMFTemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
NMFTemp <- merge(NMFTemp, SampleInfo[, c("ID","Administrative subdivision
1")], by="ID", all.x=TRUE)
NMFTemp <- melt(NMFTemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
NMFdf <- NMFTemp

for(i in 2:length(NMFmatrix)){
  NMFTemp <- as.data.frame(NMFmatrix[[i]])
  names(NMFTemp) <- 1:Krange[i]
  NMFTemp$K <- Krange[i]
  NMFTemp$ID <- rownames(matnumSNPrev)
  NMFTemp <- merge(NMFTemp, SampleInfo[, c("ID","Ploidy")], by="ID",
all.x=TRUE)
  NMFTemp <- merge(NMFTemp, SampleInfo[, c("ID","Country of Origin")],
by="ID", all.x=TRUE)
  NMFTemp <- merge(NMFTemp, SampleInfo[, c("ID","spp")], by="ID",
all.x=TRUE)
```

```
  NMFTemp <- merge(NMFTemp, SampleInfo[, c("ID","Administrative
subdivision 1")], by="ID", all.x=TRUE)
  NMFTemp <- melt(NMFTemp, variable.name = "Group", value.name =
"Probability", id = c("ID", "K", "Ploidy", "Country of Origin", "spp",
"Administrative subdivision 1"))
  NMFdf <- rbind(NMFdf, NMFTemp)
}

grp.labs <- paste("K =", Krange)
names(grp.labs) <- Krange

# plot
ggplot(NMFdf, aes(x = ID, y = Probability, fill = Group)) +
  geom_bar(stat = "identity") +
  facet_grid(rows = vars(K), scales = "free_x", space = "free", labeller
= labeller(K = grp.labs)) +
  scale_fill_manual(values=clist$funky) +
  labs(title = "NMF Assignment plot", y = "Membership probability") +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x =
element_blank(), axis.title.x = element_blank(), axis.ticks.x =
element_blank())

# saving plot
ggsave(filename = "Figures/NMFK10-14.png", device = "png", width = 7,
height = 4)
```

ANNEX 21. Approach comparison

```
# Comparing cluster characteristics for K = 13
# fastStructure simple prior
fsSimClusterSum <- ggplot(fsSimDF[fsSimDF$K == 13 & fsSimDF$Probability >
0.01,], aes(x = Group, y = Probability, color = Group)) +
  geom_boxplot(show.legend = FALSE) +
  scale_color_viridis(discrete = TRUE, option = "D", direction = -1) +
  stat_summary(fun.y=mean, colour="black", geom="text", size = 1.8,
show_guide = FALSE, vjust = 1.4, aes(label = round(..y.., digits = 3))) +
  labs(title = "fastSTRUCTURE (simple prior) - K = 13") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5, size = 10), axis.title.x =
element_text(size = 10), axis.title.y = element_text(size = 10))

# fastStructure logistic prior
fsLogClusterSum <- ggplot(fsLogDF[fsLogDF$K == 13 & fsLogDF$Probability >
0.01,], aes(x = Group, y = Probability, color = Group)) +
  geom_boxplot(show.legend = FALSE) +
  scale_color_viridis(discrete = TRUE, option = "D", direction = -1) +
  stat_summary(fun.y=mean, colour="black", geom="text", size = 1.8,
show_guide = FALSE, vjust = 0.8, aes(label = round(..y.., digits = 3))) +
  labs(title = "fastSTRUCTURE (logistic prior) - K = 13") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5, size = 10), axis.title.x =
element_text(size = 10), axis.title.y = element_text(size = 10))

# DAPC
DAPCClusterSum <- ggplot(DAPCDF[DAPCDF$K == 13 & DAPCDF$Probability >
0.01,], aes(x = Group, y = Probability, color = Group)) +
  geom_boxplot(show.legend = FALSE) +
  scale_color_viridis(discrete = TRUE, option = "D", direction = -1) +
  stat_summary(fun.y=mean, colour="black", geom="text", size = 1.8,
show_guide = FALSE, vjust = 2, aes(label = round(..y.., digits = 3))) +
  labs(title = "DAPC - K = 13") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5, size = 10), axis.title.x =
element_text(size = 10), axis.title.y = element_text(size = 10))

# SVD + DA
SVDDAClusterSum <- ggplot(SVDDAdf[SVDDAdf$K == 13 & SVDDAdf$Probability >
0.01,], aes(x = Group, y = Probability, color = Group)) +
  geom_boxplot(show.legend = FALSE) +
  scale_color_viridis(discrete = TRUE, option = "D", direction = -1) +
  stat_summary(fun.y=mean, colour="black", geom="text", size = 1.8,
show_guide = FALSE, vjust = 2, aes(label = round(..y.., digits = 3))) +
  labs(title = "SVD + DA - K = 13") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5, size = 10), axis.title.x =
element_text(size = 10), axis.title.y = element_text(size = 10))

# NMF + LS
NMFClusterSum <- ggplot(NMFdf[NMFdf$K == 13 & NMFdf$Probability > 0.01,],
aes(x = Group, y = Probability, color = Group)) +
  geom_boxplot(show.legend = FALSE) +
  scale_color_viridis(discrete = TRUE, option = "D", direction = -1) +
  stat_summary(fun.y=mean, colour="black", geom="text", size = 1.8,
show_guide = FALSE, vjust = 2.5, aes(label = round(..y.., digits = 3))) +
  labs(title = "NMF + LS - K = 13") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5, size = 10), axis.title.x =
element_text(size = 10), axis.title.y = element_text(size = 10))

# plot
ggarrange(arrangeGrob(fsSimClusterSum, DAPCClusterSum, NMFClusterSum,
heights = c(0.3,0.3,0.3)), arrangeGrob(fsLogClusterSum, SVDDAClusterSum,
heights = c(0.3,0.3, 0.3)), ncol = 2)

# saving plot
ggsave(filename = "Figures/ClusterSummary.png", device = "png", width =
7, height = 7)

# Plotting results together
# bind data frames
K13df <- rbind(cbind(fsSimDF[fsSimDF$K == 13,], method = "fS (simple)"),
cbind(fsLogDF[fsLogDF$K == 13,], method = "fS (logistic)"),
cbind(DAPCDF[DAPCDF$K == 13,], method = "DAPC"), cbind(SVDDAdf[SVDDAdf$K
== 13,], method = "SVD + DA"), cbind(NMFdf[NMFdf$K == 13,], method =
"NMF"))
method.labs <- unique(K13df$method)
names(method.labs) <- method.labs

# plot
ggplot(K13df, aes(x = ID, y = Probability, fill = Group)) +
  geom_bar(stat = "identity") +
  facet_grid(factor(method, levels = names(method.labs)) ~ ., scales =
"free_x", space = "free", labeller = labeller(method = method.labs)) +
  scale_fill_manual(values = clist$funky) +
  labs(title = "Assignment plot per method K = 13", y = "Membership
probability") +
  theme(plot.title = element_text(hjust = 0.5), axis.text.x =
element_blank(), axis.title.x = element_blank(), axis.ticks.x =
element_blank(), axis.text.y = element_text(size = 8), strip.text.x =
element_text(size = 6))

# saving plot
ggsave(filename = "Figures/AssignmentPlotComparisonK13.png", device =
"png")
```

ANNEX 22. Data sharing agreement with CIP

https://shorturl.at/fsABT