Using Approximate Bayesian Computation to Infer the Number of Populations from SNP Genotype Data

Fabian Bergmann 372918

Technische Universität Berlin
Department of Electrical Engineering
and
Computer Science

July 18, 2019

A thesis presented for the degree of
Bachelor of Computer Science

Main Examiner: Prof. Dr. Manfred Opper

Parts of the thesis was written during an internship with the Mathematical and Computational Biology work group of the TIMC-IMAG laboratory at the Université polytechnique de Grenoble.

Special thanks go to Olivier François and Michael Blum for their advice and support.

Abstract

A great challenge in analysing population structure poses the estimation of the number of populations K in a genotype data set. Several techniques have been developed, such as a Bayesian modelling approaches with MCMC or sparse non-negative matrix factorisation with a cross entropy criterion, which however usually require elaborate and extensive computations. The following thesis will propose and investigate a fast and scalable method that constructs a generative model, which generates synthetic SNP data where the number of populations is known. Since the dimensionality of SNP data is substantially large and the task is to solely infer the single parameter K, a summary statistics is used that exploits the relationship between PCA and clustering. A synthetic training data set is consequently created and a supervised learning technique, in this case gradient boosted decision trees, employed. The supervised learning method intends to construct a generalisation between the pattern found in the magnitude of the eigenvalues of a given data set and the number of populations in that data set. The method is also evaluated for two real world data sets. In the light of the Ambiguity found in real world data sets, it yields satisfactory results. Further improvements are proposed.

Exposé

Eine der zentralen Herausforderungen in der Populationsgenetik besteht darin, die Anzahl der vorhandenen Populationen K in einem Datensatz, bestehend aus Stichproben von Genotypen einer Spezies, zu ermitteln. Viele derzeitige Methoden beruhen auf ausführlichen und aufwendigen Analysen des Datensatzes, wie beispielsweise Bayesian Modelle in Verbindung mit MCMC oder Sparse-Non-Negative-Matrix-Factorisation mit einem Cross-Entropy Kriterium. In der folgenden Arbeit wird eine skalierbare Methode vorgestellt, die die Anzahl der vorhandenen Populationen in einem SNP-Datensatz schätzt. Die Methode generiert vorerst synthetische SNP-Daten mittels eines in der Populationsgenetik weit verbreiteten generativen Models. Da SNP-Daten zumeist eine erhebliche Dimension aufweisen, und da zumal lediglich ein einzelner Parameter K geschätzt werden soll, wird eine Summary-Statistic angewandt, die auf Zusammenhängen zwischen Clustering und PCA beruht. Entsprechend wird ein synthetischer Trainingsdatensatz, bei dem die genauen Anzahlen der Populationen bekannt sind, zusammengestellt und damit eine Supervised-Learning Methode trainiert, in diesem Fall Gradient-Boosted-Decision-Trees. Die Supervised-Learning Methode versucht eine generalisierte Verbindung zwischen der Anzahl der Populationen in einem entsprechenden Datensatz und der Werte der Eigenwerte des Datensatzes zu lernen. Die Qualität der Schätzung des Modells wird untersucht, auch anhand zweier realen Datensätzen. Die Ergebnisse sind, angesichts der vorhandenen Unklarheiten in realen Datensätzen, zufriedenstellend, wobei Möglichkeiten der Verbesserung vorgeschlagen werden.

Contents

1 Introduction

A central objective in population genetics is to evaluate population structure. Genetic differences of member individuals are analysed to detect systematic genetic similarities and dissimilarities that could indicate the presence of various populations. The biological definition of a population generally follows the lines: a population is a "group of organisms of the same species living within a sufficiently restricted geographical area so that any member can potentially mate with any other member of the opposite sex" (Hartl, Clark, and Clark 1997). However, as only a theoretical setting is assumed without any further context, like geography, social hierarchy and other factors that could hinder gene flow, individuals will be associated with one another upon to what extent their genetic information is sufficiently homogeneous. In reality population structure usually possesses a hierarchical form, in the sense that within each group further groups can be distinguished until solely the individual remains. Therefore, every group of sufficiently genetic homogeneous individuals in the highest population structure level will subsequently be considered as a population.

An allele is a variant of a gene found at a specific location on a chromosome (locus), and consequently alleles are responsible for the appearance of genetic variation in a species. An allele frequency describes the probability $p(a|k)$ that an individual from population k has the allele a. Since individuals in a population possess similar genotypes (their genetic make-up), some alleles are encountered more frequently in these individuals than in others, which is what therefore in total sets the populations from one another apart. The allele frequencies of a population sufficiently summarise it.

The ampleness of information usually found in a genome poses a challenge dimensionality wise if all of it were to be captured. For tasks that solely involve the analysis of genetic variation between individuals and groups of individuals it suffices to rely on the evaluation of a limited amount of genes. Mostly genes are chosen that are known to be subject to genetic variation, these particular genes are called markers. The allele frequencies at markers will consequently be used to approximate the genetic variance found in a set of to be analysed individuals. Next, it is assumed that biallelic SNP markers are used, so there are two possible alleles at every marked locus.

A challenging task that arises with in the domain of assessing population structure, is to infer the number populations K. Along determining other assumptions about population structure, like what degree of similarity is necessary to be part of a population, great difficulty stems from the often encountered ambiguity of distinguishing the hierarchical levels of population structure from another.

Many models in population genetics demand the specification of the number of populations as hyperparameter in the given data. The assessability and comparability of models utilising K as hyperparameter is undermined should no realistic value for K be determined beforehand. For e.g. a model with faulty underlying model assumptions could not be easily rejected since a goodness of fit evaluation would be skewed for an unrealistically specified value of K.

So far no single best solution for determining K has been established, probably due to the manifold of existing population scenarios. However, one promising method that yielded reliable results was based on a generative model proposed by Pritchard, Stephens, and Donnelly 2000; Falush, Stephens, and Pritchard 2003. Although they implemented an MCMC algorithm in the software package STRUCTURE for determining all parameters of the generative model, their algorithm has also been widely applied for a reasonable estimation of the hyperparameter K. The most common approach involves an adhoc heuristic for detecting the "elbow" in the plotted likelihood outputs of the MCMC algorithm for different values of K. However, as the process of retrieving genetic information has been refined and thus the dimensions of genetic data have dramatically increased, the implemented MCMC algorithm has not proven to be scalable. Further more refined versions have been published (Patterson, Price, and Reich 2006), but also struggle with high dimensional data, which modern genotype data has become (data sets with millions of SNPs). Also running an MCMC algorithm for solely inferring the number of populations seems computationally a bit too elaborate.

Further propositions for calculating the number of populations involve the insight that population structure in a given data set is conveyed through a certain pattern in the eigenvalues of the corresponding covariance matrix. In summary, for K populations K distinguishable (linearly independent) clusters can be observed, that yield $K-1$ significantly larger eigenvalues. This insight will be used to construct a summary statistics.

Subsequently approximate Bayesian computation is used by constructing a summary statistics with the eigenvalues of the covariance matrix from SNP genotype data to efficiently handle the dimensional challenge. In addition, a generative model based on a model proposed by Pritchard, Stephens, and Donnelly 2000; Falush, Stephens, and Pritchard 2003 will be used to generate synthetic data, where the ground truth for the number of populations is known. Then a supervised learning method attempts to generalise a connection between the calculated summary statistics of the synthetic data instances and its ground truth values for K, to yield a model that estimates the number of populations for highly dimensional SNP genotype data with the same model assumptions as the model implemented in STRUCTURE. For the supervised learning method gradient boosting with decision trees will be used.

1.1 Reframing the Problem

With the established biological framework, the task can be formulated in a more specific manner: Given the SNP genotype data of individuals from the same species, each entry holds a value from $\{0,1,2\}$, organised in a matrix X where each row corresponds to an individual and each column to a locus, the task is to infer how many populations K are present in the data X. If the markers sufficiently captures the genetic variation of the sampled individuals, it is expected for members of the same population to cluster together in the feature space spanned by the parameterised alleles of the selected SNP markers. A cluster is a set of data points that are similar enough (by some means of evaluation) to be grouped together, which coincides with the given definition for a population. So for the K populations K different clusters should be observable in the feature space. Each cluster corresponding to a specific population, where the position of the cluster should be determined by the allele frequencies of the respective population.

1.2 Existing Approaches

Clustering problems have been well investigated and many methods for solving these problems have been proposed. The methods can predominantly be apportioned among

two groups.
One group compares the similarities between data points by the means of a distance measure. Structure is attempted to be recognized by evaluating the distance of every data point to every other data point. Examples of distance based clustering methods are centroid based clustering, hierarchical clustering, etc.
The other group contains methods that are based on statistical model assumptions. Every data point is considered to be a random draw of a probability distribution. The parameters of the distributions are inferred via common statistical methods, such as maximum likelihood or Bayesian methods.
Model based approaches, because they facilitate interpretability and flexibility by introducing explicit parameters, have been a popular choice in population genetics. A widely applied (Rosenberg, Pritchard, et al. 2002; Harter et al. 2004; Rosenberg, Burke, et al. 2001) implementation of a model based method is the software package STRUCTURE first developed by (Pritchard, Stephens, and Donnelly 2000; Falush, Stephens, and Pritchard 2003). In short, at its core lies the modelling of K unobserved populations by assigning them specific allele frequencies, where it is assumed that the genotype of an individual is drawn according to its population allele frequencies. Furtherly the whole is embedded into a Bayesian framework, where priors are set over the population membership of each individual and the allele frequencies of each population, in order to include scenario specific conditions like geography into the model and thus increase the model flexibility. Given some genotype data, the parameters of the model are inferred via a Markov Chain Monte Carlo (MCMC) algorithm with Gibbs sampling.
For the matter of determining the number of populations K present in the data the posterior $P(K|X)$ distribution is approximated for various K, where X is the given data. As only the maximum posterior of the selected K values is to be determined following reduced relation is received

$$P(K|X) \propto P(X|K)P(K)$$

(For further information on Bayes' theorem, see section about Approximate Bayesian Computation)
The likelihood $P(X|K)$ is estimated by making a Gaussian assumption about the distribution of the deviance D of the parameters model parameters conditioned on the data X. The assumption allows for determining the Gaussian distribution by only calculating the mean and variance of D with the MCMC samples of the stationary distribution. However, even Pritchard, Stephens, and Donnelly (2000) themselves describe this as a "dubious" assumption rendering the method as only a supportive argument for picking K.
Nonetheless, STRUCTURE is widely used for inferring the number of populations, however in a different way than originally intended. Most often the decision of K follows a heuristic based on the likelihood $P(X|K)$, which is calculated with the implemented Monte Carlo algorithm in STRUCTURE. The heuristic most commonly applied, investigated and proposed by Evanno, Regnaut, and Goudet 2005, relies on the search for the largest jump in the second derivative of a composed log likelihood function. The likelihood function $L(K)$ is calculated by averaging the log likelihood of each MCMC step and further subtracting half of the variance off that mean (a penalisation for unstable models). The second derivative is defined as $L''(K|X) = L'(K+1) - L'(K)$ with of course $L'(K) = L(K+1) - L(K)$. Several runs are completed and therefore several results for $L(K)$ received. The selection criteria then is $\Delta K = \frac{m(|L''(K)|)}{s(L(K))}$, where m is the mean and s

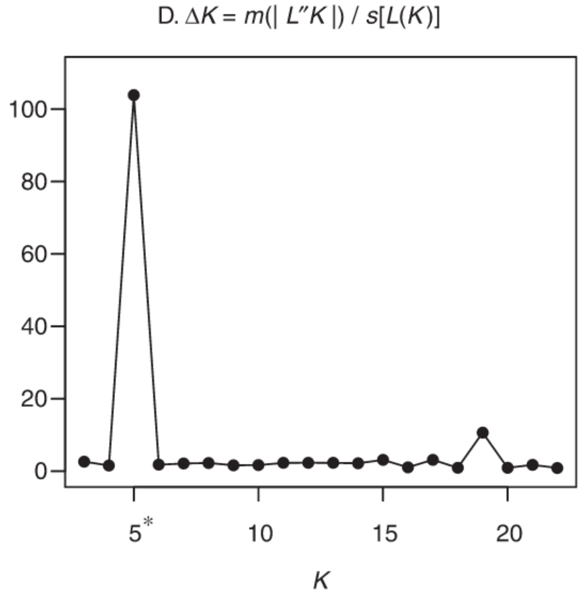


Figure 1: Visualisation of the Heurisitc for an example. Taken from Evanno, Regnaut, and Goudet 2005.

the standard deviation.

The method offers reasonable results, even for hierarchical clusters (the population structure in the highest hierarchy layer is the expected result), however it was conceived only as an ad-hoc solution strongly reliant on the results outputted by the STRUCTURE algorithm. Furthermore, multiple exhaustive rounds of the MCMC algorithm have to be completed, which tends to be at least computationally very time consuming (even impractical) as the dimension of genotype data has rapidly expanded with the introduction of more modern genome sequencing equipment.

Another method that also holds the underlying assumption that the genotypes of individuals can be described through the admixture of K populations uses an sNMF (sparse non-negative matrix factorization) framework (Frichot, Mathieu, et al. 2014). An approximate factorization of genotype data matrix X in to QG is sought, such that a least squares criterion $\| X - QG \|_F^2$ is minimized. Whereby Q should attain the admixture proportions of the populations for each individual and G the loci probabilities of each population. The idea is to represent the genotype of an individual as a linear combination of the population allele frequencies, with the constraint that the coefficients come from the probability Simplex. Setting the number of columns of Q and the number of rows of G to K enforces the number of populations. A regularization technique introduces sparsity, such that the algorithm is incentivised to pick a factorisation where individuals are apportioned to a single or few populations (many coefficients set to zero). The number of populations K is predicted via a cross-entropy criterion, where 5% of the genotype values of X are masked and then imputed by the algorithm. The value of K is chosen that minimizes the cross-entropy between the masked and the imputed values.

A more recent approach involves the insight that cluster structure is also resembled in a structured form in the spectrum of the covariance matrix of the respective data. The connection between the spectrum of a matrix and its clustering structure has been subject to research for a fair amount of time. It was first discovered in graph theory Donath and Hoffman 1973 Fiedler 1973 and later introduced into machine learning (Shi and Malik 2000; Meila and Shi 2001; Ng, Jordan, and Weiss 2002) for further information see Von Luxburg 2007. In general, the relevant insight states that: suppose K clusters can be observed in the data matrix X, then the first $k-1$ eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{k-1}$ of the covariance Matrix of X are significantly larger than the remaining eigenvalues. The number of clusters can thus be inferred by examining the eigenvalues of the data.

Inferring the number of populations by exploiting the occurrence of population structure in the eigenvalues was firstly applied by Patterson, Price, and Reich 2006. They used insights from random matrix theory, which state that (under various assumptions, like the covariance matrix has properties of a random Wishart matrix) the significant $K-1$ eigenvalues are approximately Tracy-Widom distributed. The Tracy-Widom distribution describes the distribution of the largest eigenvalue for various random Gaussian matrix ensembles, like the Gaussian orthogonal ensemble ($\beta = 1$) to which the Wishart-matrix belongs. A statistical test is constructed where the eigenvalues are checked with a chosen p-value if there is substantial evidence for the eigenvalue to be Tracy-Widom distributed.

Further research has been made with the use of random matrix theory (RMT) to concretise the behaviour of the first $k-1$ eigenvalues, including, under certain assumptions, a mathematically quantifiable threshold that distinguishes significantly larger eigenvalues from lower ones (K. Bryc, W. Bryc, and Silverstein 2013). However, explicit calculations tend to have problems when confronted with real world problem instances.

Out of these reasons, instead of relying on ad hoc heuristics or attempting to formulate specific mathematical methods with multiple underlying assumptions which in turn have trouble with real data, the use of a machine learning technique is employed. It attempts to generalise the connection between the number of populations present in genotype data and the correlating structure observed in the eigenvalues.

Patterson, Price, and Reich 2006 regards the eigenvalues approach as a black-box method, nonetheless it could also be categorised as distance method as it is dictated by the second central moment (variance), which is determined by the distance of the samples from the mean. The following presented supervised learning approach, using a model to generate synthetic training data and then summarising it by its eigenvalues, can therefore be viewed as a synthesis of model based and distance based cluster analysis.

2 The Generative Model

The subsequently presented model resembles the Bayesian model implemented in STRUCTURE. A first layer, referred to subsequently as the F-layer, originates from Falush, Stephens, and Pritchard 2003 to introduce the possibility of adjusting the correlation of allele frequencies between populations. The F-layer is responsible for determining the allele frequencies of the populations. The second layer, from now on referred to as the admixture layer, stems from Pritchard, Stephens, and Donnelly 2000 and allows individuals to have admixed genotypes. In other words, the frequencies of several populations are partially contributing to the pool from which the genotype of an admixed individual is sampled. Thus, the admixture layer samples the genotypes of the individuals, whereby controlling the proportions each population contributes to the genotype of an individual.

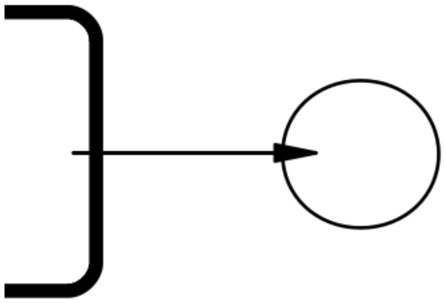


Figure 2: Schematic abstraction. On the mainland the ancestral population resides. As individuals from the mainland migrate to an island their mating choices are secluded and their genotypes drift and start differing from the ancestral population.

The SNP markers are biallelic, since the SNP markers locate a mutation in a single nucleotide base pair. That a base pair is effected by several mutations and these mutations assert themselves in the population is, because of the low chances of single nucleotide being affected by a mutation, very unlikely and therefore the possibility of more than two alleles occurring at an SNP marker is neglected. This allows for a notation simplification of the allele frequencies. Let subsequently p_{kl} determine the probability that an individual from population k has a mutation at locus l. Of course, the existence of mutant variants is always relative to some reference genome from which over sufficiently long amount of time mutations formed and were able to gather some share in the allele frequencies of a population.

2.1 The F-Layer

The motivation for introducing correlation between allele frequencies of populations is that populations often originate from a common ancestral population, so each population stems from the same starting allele frequencies. Reasons for the emergence of new populations are numerous, ranging from geographical divides to social structure affecting the genetic make-up, the component common to all is that a group of individuals excludes themselves sufficiently from mating with the rest of the population and thus practices inbreeding. Schematically in population genetics these scenarios of fragmentation are often abstracted as a set of K secluded islands to which some individuals of an ancestral population A migrated and interbreeding on each island occurs.

The formation of different populations on the islands, described by their distinct allele frequencies, is due to a phenomenon called random genetic drift. From a simplified view point, excluding all the biological idiosyncrasies and only looking at the alleles at a single locus, random genetic drift can be seen as an urn model. Each allele receives a different colored ball and the proportions of the balls lying in the urn should represent the allele frequencies of the population. The alleles that the offspring receives are sampled from the urn (with replacement). However, since the amount of individuals in the offspring is finite, the offspring will most likely not possess exactly the same allele frequencies as the parents did (not have the same ball proportions). This causes the allele frequencies

of a population to vary until one allele achieves fixation. Fixation occurs when by chance only one specific allele is sampled from the urn (only one colour), thus eliminating all its competing alleles. For more information consult (Hartl, Clark, and Clark 1997; Gillespie 2004)
Repeated over time and expanded to the continuous case, genetic drift can be formulated mathematically as a stochastic process over the change of allele frequencies, where the total dominance of an allele is an absorbing state (Crow, Kimura, et al. 1970). Therefore, after a sufficient amount of time, a decrease in genetic diversity will be observed in inbreeding populations as more alleles tend to fixation. Wright's F-statistic is a widely used measure for evaluating the divergence of allele frequencies of populations from the mean. It consists of a single value F_{ST} and is calculated as $F_{ST} = Var(p_a)/\overline{p_a}(1-\overline{p_a})$, where $Var(p_a)$ is the variance of the allele frequencies for an allele a across the evaluated populations and $\overline{p_a}$ the mean. Once all alleles are fixed (considering an infinite amount of populations), it is expected for a proportionate amount according to $\overline{p_a}$ to have fixed the allele a. The variance would consequently conclude to $\overline{p_a}(1-\overline{p_a})$ making the equation equal to one and indicating total fixation. If the variance is near zero all populations would still most likely have not diverged far from the overall starting allele frequency.

The model will employ an analogous adoption of the F-statistic. The ancestral allele frequency p_{Al} acts as the mean from which all other frequencies originate. Instead of a single parameter F_{st}, a hyperparameter F_k for each population k is introduced that controls the magnitude of its divergence from the ancestral population.
The allele frequencies for a population k are sampled from a beta distribution, with following parametrisation:

$$p_{kl} = \beta\left(p_{Al}\frac{1-F_k}{F_k}, \quad (1-p_{Al})\frac{1-F_k}{F_k}\right)$$

The parametrisation arises when the mean of a the beta distribution is set equal to p_{Al} and the variance equal to $F_{ST}p_{Al}(1-p_{Al})$. The choice of using a beta distribution is motivated by the fact that it is the stationary distribution that is received for the stochastic process if a counter force working against allele fixation is added, such as accounting for a steady stream of migrants from the ancestral population or further mutations that sustain an allele (Balding 2003).
It is assumed that the loci are independent from one another, as a consequence factors like linkage disequilibrium can not be respected, however it allows for the independent sampling at each locus with the same F_k value.
Proceeding, the allele frequencies for a population k are joined together to a vector p_k and then merged with all other K populations to a matrix $\mathbf{F} = [p_1 p_2 \dots p_K]^T$ of size $K \times L$, where L is the number of loci. Each column of $\mathbf{F}$ gives the allele frequency for each population at a specific locus l.

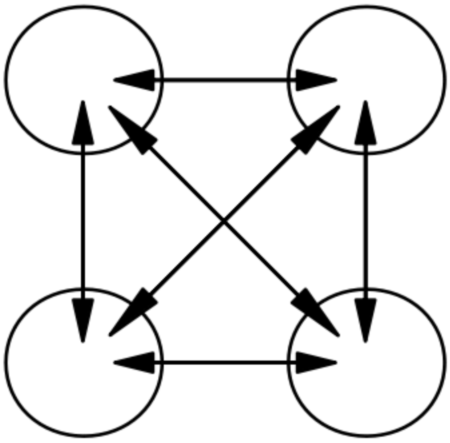


Figure 3: Abstraction of the admixture model. Each population is mating secluded on their own island, except for some migrants that arrive from other populations. The genotype of the offspring of the migrant will exhibit the admixed heritage.

2.2 The Admixture Layer

Apart from populations drifting from another apart, individuals of populations also migrate and mate with other individuals of other populations, which results in individuals having a genotype that exhibits the shared ancestry of different populations. The admixture layer introduces the prospect of modelling diverse ancestry according to the admixture model presented in Pritchard, Stephens, and Donnelly 2000, whereby the flexibility is achieved by sampling the proportions of admixture from a Dirichlet distribution. Its hyperparameters allow for the fine-tuning of the probabilities over the potential admixing proportions of the populations. Section 3.6 provides a more thorough look of the modelling with the Dirichlet distribution.

Before translating the admixture proportions to the genotypes of individuals, another matrix $\mathbf{Q}$ is created that contains the admixture proportions of every individual. Mathematically, for an individual i the mixture weights q_i are sampled from a Dirichlet distribution with K, parameterised according to the number of populations with concentration hyperparameters $\alpha 1, \alpha 2, \ldots, \alpha_K$. A "non admixed" individual j also receives admixture proportions with the speciality that the only non-zero value is a one at the kth position (with a lenient notation), indicating that the individual belongs to population k, so $q_j = [0_1, \ldots, 1_k, \ldots, 0_K]^T$. All N individuals are combined to a mixing matrix $\mathbf{Q} = [q_1, q_2, \ldots q_M]^T$ of dimensions $N \times K$.

2.3 Combining the Layers

The admixture proportions of an individual act upon each locus by weighting the allele frequency of each population at the locus according to the respective proportions and subsequently summing them. Since this accords to a linear operation, the matrix multiplication with both established matrices $\mathbf{P} = \mathbf{QF}$ yields a matrix $\mathbf{P}$ of dimension $N \times L$ that holds the allele frequencies at each locus for all individual from which their genotypes are sampled. In a final step each entry of $\mathbf{P}$ is used to sample twice from a Bernoulli distribution (binomial - two tries), as it is assumed that individuals are diploidic. Each individual consequently has at each locus a value of $0,1,2$, according to how many mutations in total both of their chromosomes exhibit. Adapting to species with a different ploidity is easily possible.

Further assumptions that the model expresses include random mating and thus the absence of further lower level population structure within the populations and that the alleles captured by the markers are neutral, meaning that they have no impact on the fitness of an individual and are therefore (with the allele independence assumption) exempt from natural selection.

2.4 Model Summary

In summary, the generation of a new SNP data set for which the number of populations K is known proceeds as following:

1. Sample the ancestral allele frequencies $p_{Al} \sim Uniform(0,1)$
2. Determine the F-values F_k
3. For each of the K populations:
   (i) Sample $p_{kl} \sim \beta\left(p_{Al}\frac{1-F_k}{F_k},\ (1-p_{Al})\frac{1-F_k}{F_k}\right)$
   (ii) Combine allele probabilities into matrix $\mathbf{F}$
4. For each individual i:
   (i) Choose admixture coefficients $q_i \sim Dir(\alpha_1,\ldots,\alpha_k)$
   (ii) Combine admixture coefficients into matrix $\mathbf{Q}$
5. Calculate admixture $\mathbf{P} = \mathbf{QF}$
6. Convert each value p of $\mathbf{P}$ by sampling $Bernoulli(2,p)$

2.5 Looking at Admixture

The admixture of an individual is determined by the Dirichlet distribution. The Dirichlet distribution is parametrised by K hyperparameters $\alpha_1,\alpha_2,\ldots,\alpha_k$. K corresponds to the desired dimension of the output. The probability density function is defined as following

$$f(x_1,\ldots,x_K,\alpha_1,\ldots,\alpha_K) = \frac{1}{B(\alpha)}\prod_{i=1}^{K} x_i^{\alpha_i-1}$$

where $\sum_{i=1}^{K} x_i = 1$ and all $x_i \geq 0$. So the Dirichlet distribution defines a probability density on the $K-1$-simplex and is therefore a natural choice for sampling admixture coefficients.

Of particular interest for the described model are the hyperparameters, also called concentration parameters, as they control the mode and the variance around it. For values

$a_i \geq 1$ the distribution has a single mode, whose coordinates at the maximum x is given by Bishop 2006:

$$x_i = \frac{\alpha_i - 1}{\sum_{k=1}^{K} a_k - K}$$

The mode moves therefore more towards those simplex vertices/populations that possess a relatively higher valued hyperparameter compared to the others. In addition, the variance σ, given by

$$\sigma_i = \frac{\alpha_i (\alpha_0 - \alpha_i)}{\alpha_0^2 (\alpha_0 + 1)}$$

where $\alpha_0 = \sum_{i=1}^{K} \alpha_i$, reveals that higher values of hyperparameters lead to a decrease of the variance, meaning a higher concentration around the mode.
These two properties can be exploited to control the probability of sampling certain admixture coefficients. Furthermore, by sampling from the same Dirichlet distribution one is able to simulate various population scenarios, such as a detached admixed cluster, which would correspond to a mode with high concentration parameters. Or a population that experienced migration originating from another population, which would coincide with a degenerated Dirichlet distribution that only has two non-zero concentration values for the two involved population (in the end a beta distribution).

2.6 Investigating the F-Value

The F-value, of which each population receives its own, determines the dispersion of the allele frequencies of a population from the ancestral allele frequencies. In general, low F-values lead to a strong correlation of the allele frequencies with the ancestral, a high value leads to a decoupling.
The effect of the F-value can be best evaluated through the visualization of some example beta distribution plots. Figures 4, 5, 6 depict the distribution of a single allele frequency originating from different ancestral allele frequencies $p_{Al} \in \{0.3, 0.5, 0.95\}$ for F-values 0.01, 0.1 and 0.5. Two main insights can be concluded: Firstly, the proposition is supported, that the higher the F-value the more probable a more different allele frequency than the ancestral one is chosen, expressed in the increase in variance. Therefore, it is to be expected that if the populations are assigned large enough F-values they will be more easily distinguishable. And secondly, as p_{Al} leans towards fixation, the sampled allele frequency will most likely fixate the same allele. This leads to the conclusion, that if the genotypes in the ancestral population are more homogenic then the offspring populations are probably similarly homogenic and harder to distinguish. Higher F-values also lead to a higher tendency of fixation. This coincides with the intention that a high F-value corresponds to for e.g. longer time in isolation, such that allele frequencies had time to drift and most likely fixate.

Figure 4: Beta plots for an F-value of 0.01 and different ancestral allele frequencies

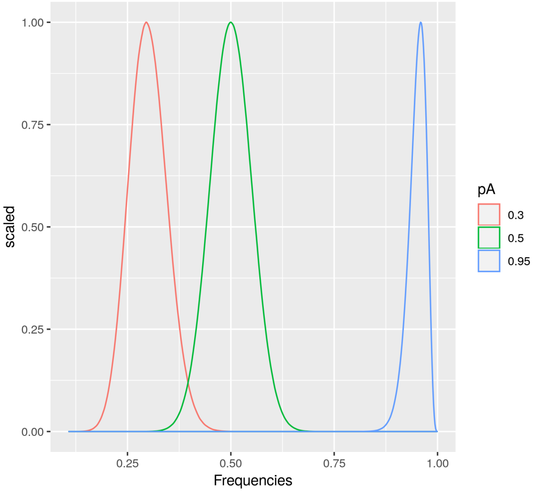


Figure 5: F-value of 0.1

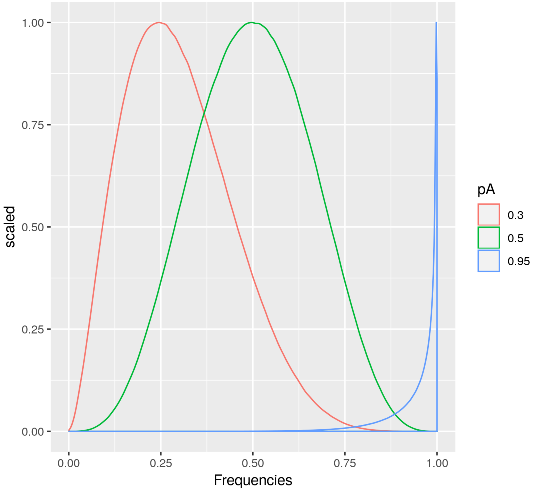
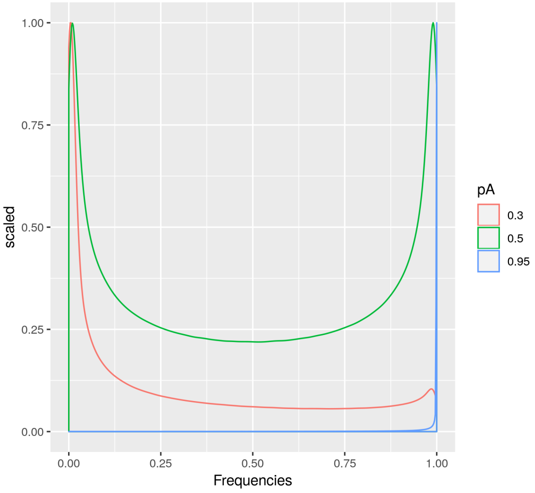


Figure 6: F-value of 0.5



2.7 Further Model Analysis

The allele probabilities of each individual from which its genotype is sampled lie within the $K-1$ probability simplex. As presented the allele frequencies, from which the genotype of an individual is sampled, are constructed by a linear combination of proportions of the population allele frequencies. This means that the allele probabilities of each individual is confined by an even smaller simplex within the $K-1$ simplex determined by the population allele frequencies which constitute the vertices. So the sampling space from which the allele frequencies for a locus l are determined corresponds to a simplex that moved the $K-1$ simplex vertices to the allele frequencies of the populations. From another perspective one can interpret the allele frequencies as constructing a change of basis transformation through the diagonal matrix $\mathbf{D} = diag(p_{1l}, \dots, p_{Kl})$ that squishes the unit vectors to the allele frequencies. Therefore, the allele frequencies from which the genotype is sampled could be determined by sampling a vector v from the $K-1$ simplex and transforming it into the allele frequency simplex through $\mathbf{D}v$. The transformation $\mathbf{FQ}$ given above generalises this idea for all L loci.
By sampling twice (Bernoulli(2)) the simplex is stretched by a factor of two, in which the genotypes of every individual lie. However, this act of discretization moves the genotypes out of the simplex of allele frequencies even if they also are scaled by a factor of two (special case: all alleles fixed). Nonetheless, the nature of distributed genotypes can be specified further by observing that the allele frequencies of each population scaled by two construct the cluster centroids $[2p_{k1}, \dots, 2p_{kL}]$ around which the individuals of a population k vary (definition Bernoulli mean).

For further assessment, a mean of quantifying the genetic dissimilarity between two individuals is necessary. As a measure of genetic distance between two individuals i and j a natural choice is to use a normalised Manhattan distance because the possible genotype values $0, 1, 2$ are discrete and already reflect dissimilarity appropriately. More concretely, let N be the number of loci used as genetic markers, then $\{0,1,2\}^N \subseteq \mathbb{R}^N$ is the set containing all possible values of genetic information. The measure of genetic distance is

$$D = \frac{1}{2N} \sum_{n=1}^{N} |l_n^i - l_n^j|$$

where l_n^i and l_n^j are the values of individual i and j respectively at locus l_n. The normalisation keeps the measure invariant to the number of loci used, as recovering more genetic information should not increase the genetic distance per se, as well as normalising out the diploidity. The measure ranges from 0, as two individuals are genetically similar, to 1, meaning genetic dissimilarity.
Suppose two individuals i and j are generated by the described model, so sampling from a Bernoulli distribution for each loci l with the respective allele frequencies $p_i(l)$ and $p_j(l)$. The expected genetic difference of both individuals then is:

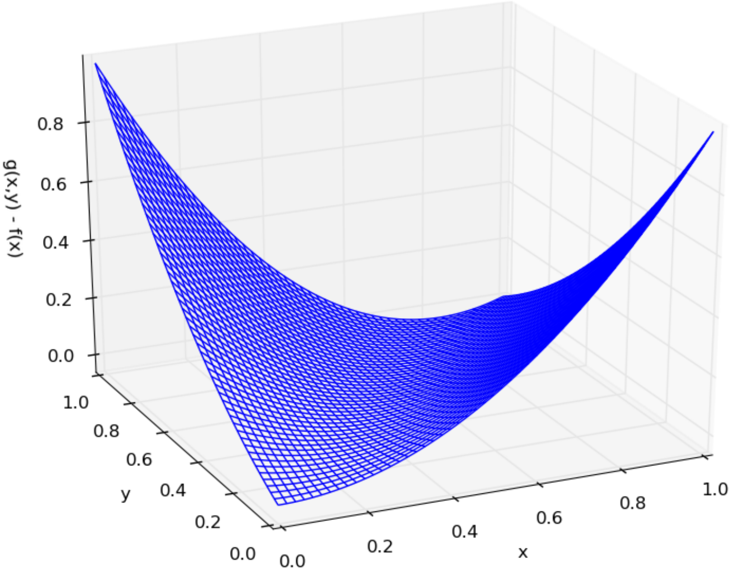


Figure 7: Depiction of the expected residual between the expected distance of two alleles sampled from the same frequency and sampled from different ones.

$$
\begin{aligned}
\mathbb{E}\left[D\right] &= \frac{1}{2N}\sum_{n=1}^{N}\mathbb{E}\left[\,|l_n^i - l_n^j|\,\right] \\
&= \frac{1}{2N}\sum_{n=1}^{N}\sum_{(x,y)\in\{(j,i),(i,j)\}} p_x(l)\left(1-p_x\right)\left(1-p_y(l)\right)^2 \\
&+ p_x(l)^2\,p_y(l)\left(1-p_x(l)\right) \\
&+ 2p_x(l)^2\left(1-p_y(l)\right)^2
\end{aligned}
$$

For individuals sampled from the same allele frequencies the expectation equates to:

$$
\begin{aligned}
&\frac{1}{N}\sum_{n=1}^{N} p(l)\left(1-p(l)\right)\left(p(l)+\left(1-p(l)\right)^2\right) \\
&= \frac{1}{N}\sum_{n=1}^{N} p(l)\left(1-p(l)\right)
\end{aligned}
$$

This lends further justification to the choice of the distance measure for diploid individuals. It corresponds to the expected empirical genetic variance found in a population without any further underlying substructure. A result also received calculating the F_{ST} value when no further populations are present.

Let the expected distance of two individuals with arbitrary allele frequencies at a locus l be given by the function $g(p_i, p_j)$ and for two individuals sampled from the same allele frequencies $f(p_i)$. It is expected, as figure 7 illustrates, that the genetic distance for individuals sampled from different allele frequencies possess a greater genetic dissimilarity, therefore it is moreover expected for individuals from the same population to cluster together. Furthermore, it should be expected that populations that have not diverged much from the ancestral population to be distinguished more difficulty.

Apart from establishing that individuals from different clusters are distinguishable, another clustering quality that would be advantageous to assess is the clustering density. As presented, individuals are sampled from probabilities located in the allele frequency simplex, however the resulting genotype could very much lie outside of the simplex due to the necessary discretisation and thus the consequently induced variance of the binomial sampling. By determining the variance of two individuals from the same population and therefore sampled from the same allele frequencies a better impression of the cluster (population) density can be obtained. In addition, the expected severity of individuals lying outside the simplex can be assessed.
However, before calculating the variance, for simplicity reasons the expectation of the genetic difference squared is calculated:

$$\begin{aligned}
\mathbb{E}\big[D^2\,|\,i,j\in k\big] &= \frac{1}{4N^2}\,\mathbb{E}\Big[\big(\sum_{n=1}^{N}|l_n^i - l_n^j|\big)^2\Big] \\
&= \frac{1}{4N^2}\sum_{n=1}^{N}\sum_{m=1}^{N}\mathbb{E}\big[|l_n^i - l_n^j|\,|l_m^i - l_m^j|\big] \\
&= \frac{1}{4N^2}\Bigg(\sum_{n=1}^{N}\mathbb{E}\big[|l_n^i - l_n^j|^2\big] + \sum_{n=1}^{N}\sum_{\substack{m=1\\ m\neq n}}^{N}\mathbb{E}\big[|l_n^i - l_n^j|\,|l_m^i - l_m^j|\big]\Bigg) \\
&= \frac{1}{4N^2}\Bigg(\sum_{n=1}^{N} 2p(l_n)^4 - 2p(l_n)^3 + p(l_n)^2 + p(l_n) \\
&+ \sum_{n=1}^{N}\sum_{\substack{m=1\\ m\neq n}}^{N} 4p(l_n)\big(1-p(l_n)\big)\,p(l_m)\,\big(1-p(l_m)\big)\Bigg)
\end{aligned}$$

and since

$$\mathbb{E}\big[D\,|\,i,j\in k\big]^2 = \frac{4}{4N^2}\Bigg(\sum_{n=1}^{N}\sum_{m=1}^{N} p(l_n)\,(1-p(l_n)\,p(l_m)\,\big(1-p(l_m))\Bigg)$$

The variance simplifies to

$$\begin{aligned}
Var(D) &= \mathbb{E}\big[D^2\big] - \mathbb{E}\big[D\big]^2 \\
&= \frac{1}{4N^2}\Bigg(\sum_{n=1}^{N} -p(l_n)^4 + 4p(l_n)^3 - 3p(l_n)^2 + p(l_n)\Bigg)
\end{aligned}$$

This reveals that the variance decreases significantly of the order $\mathcal{O}(N^{-2})$ with the number of loci. Therefore, less severe outliers and better observable clusters are to be expected the more loci are simulated. Furthermore, the terms in the summation possess a single maximum at 0.5, so a higher variance exists the further alleles are from fixation.

2.8 Relationship to LDA

The presented model is strongly related to a model more commonly known under the name latent Dirichlet allocation (LDA) Blei, Ng, and Jordan 2003. LDA uses in the setting of natural language processing (NLP) an ensemble of words that are probabilistically associated with certain topics, in order to determine which topics are exhibited by analysed documents, that preferably possess some associated words. Since not all existing words are associated with topics but only a selection, the selected words can be perceived as reasonable indicators of the topic context, as are the used genetic markers to determine population affiliation. A one hot encoding of whether a word is present in a document or not is the respective equivalent of whether an individual possess a gene variant at a genetic maker or not. The encoding of the selected words or respectively of the genetic markers span the feature space in which the topics/populations lie. The topics/populations then span a simplex in which the documents/individuals are mapped according to the admixture.

3 ABC and Summary Statistics

The prior described model will serve as generative model for constructing a training data set that will be used in a supervised learning scheme. Since the dimensionality of SNP genotype data poses as a challenge, approximate Bayesian computation (ABC) approaches will be employed. The core approximation is the construction of a summary statistics composed of the ordered eigenvalues of the covariance matrix from the genotype data. The chosen supervised learning technique, gradient boosting with decision trees, attempts to generalise the connection between the patterns found in the ordered eigenvalues to the numbers of populations in the respective original genotype data.

3.1 Approximate Bayesian Computation

At heart of approximate Bayesian computation (ABC) lies the inference of a desired parameter value θ for given data D by relating the conditional probability of that data given the parameter value $P(D|\theta)$ to the symmetric counterpart, the conditional probability of the parameter given the data $P(\theta|D)$. This is done by exploiting Bayes' rule:

$$P(\theta|D) = \frac{p(D|\theta)P(\theta)}{P(D)}$$

Where $P(D|\theta)$ is often called the likelihood, $P(\theta|D)$ the posterior, $P(\theta)$ the prior and $P(D)$ the evidence which is used in Bayes' rule solely for normalisation purposes.
For many problems the problem space is intractable or dimensionally too large to compute the likelihood. ABC intends to circumvent these problems.

3.1.1 Rejection Algorithm

The rejection algorithm is naturally derived from a certain perspective on conditional probability. Let A,B be a probability spaces and $P(x \subseteq A|y \subseteq B)$ the to be determined conditional probability. By infinitely sampling an event from A and from B and in addition always recording if the produced events correspond to x and y, then the desired conditional probability is computable by taking the past unsuccessful sampling iterations into consideration. Algorithmically this can be expressed as:

Algorithm 1 Conditional Probability $P(x \subseteq A|y \subseteq B)$

```
i ← 1
for ∞ do
    repeat
        $x_i$ ← event sampled from $A$
        $i \leftarrow i+1$
    until sample from $B$ is $y$
    output $x_i$
```

And the computation amounts to:

$$\begin{aligned} P\big(x \subseteq A \,|\, y \subseteq B\big) &= \sum_{i=1}^{\infty} P\big((x = x_i) \cap y\big) \, \big(1 - P(y)\big)^{i-1} \\ &= P\big(x \cap y\big) \sum_{i=1}^{\infty} \big(1 - P(y)\big)^{i-1} \\ &= \frac{P\big(x \cap y\big)}{P(y)} \end{aligned}$$

The rejection algorithm employs a basic approach for finding the posterior distribution of the desired parameter θ for specific data D. Given a known prior distribution of θ, the algorithm samples values $\hat{\theta}$ from the prior and then inputs $\hat{\theta}$ into an appropriate model to simulate some data $\hat{D}$. If the simulated data lies within a margin of error $\varepsilon \geq 0$ from data D for a chosen metric ρ , so $\rho(D,\hat{D}) \leq \varepsilon$, then the sampled prior value $\hat{\theta}$ is accepted by adding it to the final sample of parameter values for θ. The final sample should approximate the desired posterior. For further information and possible refinements such as using linear or non linear regression to counter a low acceptance rate or using Sequential Monte Carlo - ABC to sample from areas with higher posterior density the reader is referred to Csilléry et al. 2010.

If the task is for example to compare different models concerning a specific data set conventional ABC algorithms suffice. For tasks that only require a good point estimate of θ that fits well to the data D, like the maximum a posteriori (MAP), rejection algorithms might be too elaborate. Also, the algorithm demands to be run multiple times which is computationally costly.

3.1.2 ABC and Supervised Learning for Model Selection

A supervised learning method attempts to find a general connection between any given input data D and its desired output θ by training a malleable model. The model is instructed to infer the general connection by adapting itself in such a way that it minimises the empirical risk (for some chosen loss function) when solving a finite training set of size N, which is a data set with known optimal values $((D_1, \theta_1), \dots (D_N, \theta_N))$. From another perspective, a supervised learning algorithm attempts to forge a model in such a way that it perfects the approximation of the desired mapping from the input space described by D into the output space, which is desirably the correct value for θ. With a uniformly distributed training dataset a supervised learning model will approximate the maximum likelihood (maximise $P(D|\theta)$ w.r.t. θ). By changing the proportions of θ in the training data, a prior $P(\theta)$ can be implicitly set and the trained model will consequently attempt to approximate the maximum a posteriori value (MAP) (maximise $P(D|\theta)P(\theta)$ w.r.t. θ).

The task of choosing K can be viewed as a model selection problem. A rejection algorithm variant usually constitutes a reasonable choice for choosing a model for a particular data set, by for example using posterior ratios. Since the rejection algorithm produces an approximate posterior distribution, a broader choice for the selection criterion exists. However, the parameter space is, even with taking summary statistics into account, because of the large dimensionality of snp genotype data, rather daunting to tame computationally. Out of this reason one settles with the MAP value as sufficient selection criterion for the exchange of the posterior density calculation being waived. The supervised learning technique is subject to the same assumptions through the generative model and uses the same summary statistics just like a rejection algorithm.

3.1.3 Summary Statistics

Large dimensionality of a data set can undermine the practicability of an ABC-method. By summarising the data, one attempts to reduce dimensionality, while still sustaining a good approximation of the posterior. So if $S(D)$ is a summary statistics of some data D then the acceptance criterion for the rejection algorithm converts to $\rho(S(D), S(\hat{D})) \leq \varepsilon$, whereby $P(\theta|D) \approx P(\theta|S(D))$ holds sufficiently. The use of summary statistics is not confined to the rejection algorithm, rather it is a general tool that allows for a trade-off between reduction of dimensionality and the goodness of the approximation, as each summarisation usually forfeits some of the principal information. If no information is lost, so $P(\theta|D) = P(\theta|S(D))$ applies, then the summary statistics is called sufficient. A good informative choice of summary statistics is highly task and data set dependent Matthew A Nunes and Balding 2010. An overview of common heuristics and algorithms for choosing summary statistics can be found in Blum et al. 2013. To infer the number of populations K expressed in a given dataset X the conditional probability $P(K|X)$ with respect to K is maximised. Since the large dimensionality X poses substantial computational difficulties, the datasets are summarised in an effective manner, such that the approximation $P(K|X) \approx P(K|sum(X))$ is suitable for the intended inference. Bayes' theorem then yields

$$P\big(K \,|\, sum(X)\big) = \frac{P\big(sum(X) \,|\, K\big)\, P(K)}{P\big(sum(X)\big)}$$

3.2 Choosing the Summary Statistics

The choice of adequate summary statistics is essential to obtain viable results. Large dimensional data often times demands it to be summarised, so the intended methods

are reasonably applicable. In doing so, the manner of summary is of great importance because each summarisation usually forfeits some of the principal information. So one is confronted with the problem of how to effectively manage the trade-off between the practicability the method and the loss of information that could endanger desired results.
The entropy of a distribution measures the existing uncertainty about which event appears if one samples from the distribution. It is defined mathematically for a given continuous probability mass function $P(X)$ as

$$h(x) = -\int_{-\infty}^{\infty} P(x) \log\big(P(x)\big) dx$$

The principal of maximum entropy states that given some prior information about the underlying probability distribution, such as already drawn samples or a constraining property, the maximum entropy distribution that incorporates the prior information is the best distribution to respect the remaining uncertainty (Jaynes 1957). In other words, the maximum entropy distribution is the best distribution to fit the already obtained information if no further assumptions are to be added.
For a given mean μ and covariance Σ the multivariate continuous distribution that maximises the entropy is the multivariate Gaussian, for a proof the reader is referred to Cover and Thomas 2012. The entropy of the multivariate Gaussian is derived as following:

$$\begin{aligned}
h(x) &= -\int_{-\infty}^{\infty} \mathcal{N}\big(x\,|\,\mu,\Sigma\big)\, \ln\big(\mathcal{N}\big(x\,|\,\mu,\Sigma\big)\big)\, dx \\
&= \mathbb{E}\big[\, \ln(\mathcal{N}\big(x\,|\,\mu,\Sigma\big)\,\big] \\
&= \mathbb{E}\big[\, \ln\big(\det\big(2\pi\Sigma\big)^{-\frac{1}{2}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}\big)\,\big] \\
&= \frac{1}{2}\ln\big(\det(2\pi\Sigma)\big) + \frac{1}{2}\mathbb{E}\big[(x-\mu)^T\Sigma^{-1}(x-\mu)\big] \\
&= \frac{1}{2}\ln\big(\det(2\pi\Sigma)\big) + \frac{1}{2}\mathbb{E}\big[\,\mathrm{trace}\big(\Sigma^{-1}(x-\mu)^T(x-\mu)\big)\,\big] \\
&= \frac{1}{2}\ln\big(\det(2\pi\Sigma)\big) + \frac{1}{2}\mathbb{E}\big[\mathrm{trace}(I)\big] \\
&= \frac{1}{2}\ln\big(\det(2\pi e\Sigma)\big)
\end{aligned}$$

The only non-constant factor influencing the entropy of a multivariate Gaussian is the determinant of the respective covariance matrix. Since any real symmetric matrix is diagonalisable, $det(\Sigma)$ breaks down to $det(\Sigma) = det(\mathbf{Q}^{-1}) \cdot det(\Lambda) \cdot det(\mathbf{Q}) = \prod_{i=1} \lambda_i$, thus revealing that actually the eigenvalues of the covariance matrix are responsible for the magnitude of the entropy. In conclusion, by summarising the data by its covariance matrix, one implicitly approximates the data as being Gaussian and secondly it suffices for the summary to only take the eigenvalues into consideration.

3.2.1 Principal Component Analysis

Principle component analysis (PCA) is a statistical method that performs a basis transformation on a given data set, such that no linear correlations are present in the data along the novel axes. It can also be used to regress the data into a subspace that minimises the squared error. However, for the task at hand, another property, the preservation of variance, is of interest.

Since the direction of a linear correlation (if present) corresponds to the direction of the highest variance in a concerning subspace, a new axis must be aligned along that particular direction. This construction of the axes is done by requiring in an iterative manner each new axis to align with the direction that captures the most variance in the data, which has not been captured by previous axes.

The variance and the information found in a data set form a duality. Decreasing the variance in a data set by projecting it into a subspace decreases the variety in the data, thus it endangers distinguishability between data points. The amount of sustained variance after projecting the data into a subspace can therefore act as an indicator for how much information was retained. So by always maximizing the captured variance of a newly added orthogonal axis to the transformation the highest possible amount of information is retained (under the assumption that variance corresponds to information). Selecting K of these axes corresponds to a projection into a subspace of the rank K. The axes are the K largest eigenvectors of the empirical covariance matrix, as is subsequently shown.

Let $S = \frac{1}{N}XX^T - \overline{XX}^T$ denote the empirical covariance matrix of the data matrix X. Then the expression $u^T Su$ is the empirical variance of $u^T X$, which is the data X projected on to the vector u.

$$\begin{aligned} u^T Su &= \frac{1}{N} u^T XX^T u - u^T \overline{XX}^T u \\ &= \frac{1}{N} u^T X \left(u^T X\right)^T - \overline{u^T X} \left(\overline{u^T X}\right)^T \\ &= \frac{1}{N} \left(u^T X\right)^2 - \left(\overline{u^T X}\right)^2 \end{aligned}$$

The empirical variance is maximised with the restriction $\| u \| = 1$ because u is supposed to form a new standard basis. By using a Lagrange multiplier to add this restriction, the equation $\max\limits_u u^T \Sigma u - \lambda(u^T u - 1)$ is obtained.

$$\begin{aligned} \frac{\partial}{\partial u} \left(u^T \Sigma u - \lambda(u^T u - 1)\right) &= 0 \\ \Sigma u - \lambda u &= 0 \\ \Sigma u &= \lambda u \end{aligned}$$

The solution coincides with the definition of the Eigenvectors, where λ is the eigenvalue of u. Since u should be maximised, the overall solution is the eigenvector belonging to the largest eigenvalue. Solving the eigenproblem on a semi-definite matrix such as the covariance matrix Σ, yields the following factorisation:

$$\Sigma = Q \Lambda Q^T$$

Where Q is an orthogonal matrix that has the eigenvectors of Σ as its columns, and Λ a diagonal matrix with the corresponding eigenvalues on its diagonal.

3.2.2 In the Context of Cluster Analysis - An Intuitive Understanding

Subsequently, an argument will be explained that supports the connection between PCA and clustering for the given problem.

Beforehand, several clustering assumptions will be made: Clusters or rather cluster centroids are separated by a sufficient amount of distance. Clusters themselves are shaped of homogenous sizes and shapes (shapes are considered Gaussian like). Clusters are constituted by a reasonable amount of data points. As discussed, the previously described generative model generates clusters that fit the stated assumptions. The clusters shape is determined by the Bernoulli sampling around the given centroid specified by the allele frequencies of the population and the distance between clusters can be controlled by the F-values. Cluster density can also be controlled by specifying the number of individuals assigned to a population.

The term "within variance" will describe the highest possible variance received when a single cluster is projected on to a vector and the term "in-between variance" the highest possible variance received when multiple clusters are projected onto a vector.

Under the stated assumptions, the in-between variance of a set of clusters will be greater than the with-in variance of any of the considered clusters. The reasoning being, that the magnitude of the variance is dictated by the distance of the data points from the mean. In contrast to the with-in variance, the in-between variance is spread in general over a distance that is constituted by the width of the clusters and some captured distance in between them.

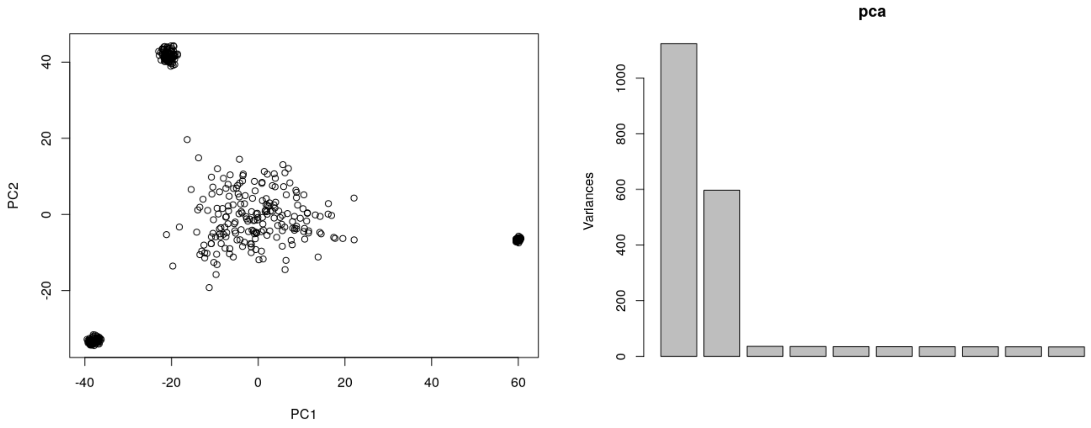
Since the eigenvectors align themselves along the biggest possible variance, it is to be expected that their orientation is greatly dominated by the in-between variance and thus their eigenvalues are significantly larger. The allele frequencies are from which genotypes are sampled stem from a simplex where the population allele frequencies compose the vertices. As discussed the simplex can be seen as the squishing of the $(K-1)$-simplex, that implements the constraint of all vectors having their values sum to 1. The constraint inhibits a degree of freedom, such that the simplex can be fully spanned by $K-1$ vectors. So it is to be expected that exactly $K-1$ eigenvectors will be needed to fully capture all the in-between variances of K population clusters. Therefore, the according eigenvalues of the first $K-1$ eigenvectors should be significantly greater than the remaining.

Another more intuitive argumentation could look like following. Consider the allele frequencies of one of the populations as the origin. Then $K-1$ linearly independent vectors are needed to be able to reach the allele frequencies of every other population. If less would be needed, then the allele frequencies would not be linearly independent and some population would not be considered a population, but a group of admixed individuals that are sampled from a linear combination from other population frequencies. The $K-1$ linearly independent vectors could consequently be composed of centroid vectors, each pointing to a centroids of one of the other populations. This implies that the space, the simplex with the population centroids as its vertices, is spanned by the centroid vectors and thus they capture approximately capture all the in-between variance (not to forget clustering assumptions include Gaussian like shape). The first $K-1$ eigenvectors orient themselves along the greatest variance and are linearly independent, therefore it is to be expected that they approximately span the space spanned by the centroid vectors. What remains is the smaller within variance found in the clusters, concluding that the first $K-1$ eigenvalues should be significantly greater than the remaining.

3.3 Examples

A synthetic problem instance generated by the model, could look like shown in figure 8, where three populations that span the simplex are observable. The populations have fairly distinct and large F-values, therefore the clusters are well separable from one another. Within the simplex is a cluster located of several admixed individuals. They were all sampled from the same Dirichlet distribution $Dir(8,8,8)$, with uniform hyperparameters, so they are concentrated around a central mode and all populations participate on average the same amount to the admixture.
Just by looking at the corresponding biggest eigenvalues, it is fairly easy, with the use of the previously established insights, to infer the number of populations. The first two eigenvalues are significantly larger than the rest, thus the number of populations should be three.

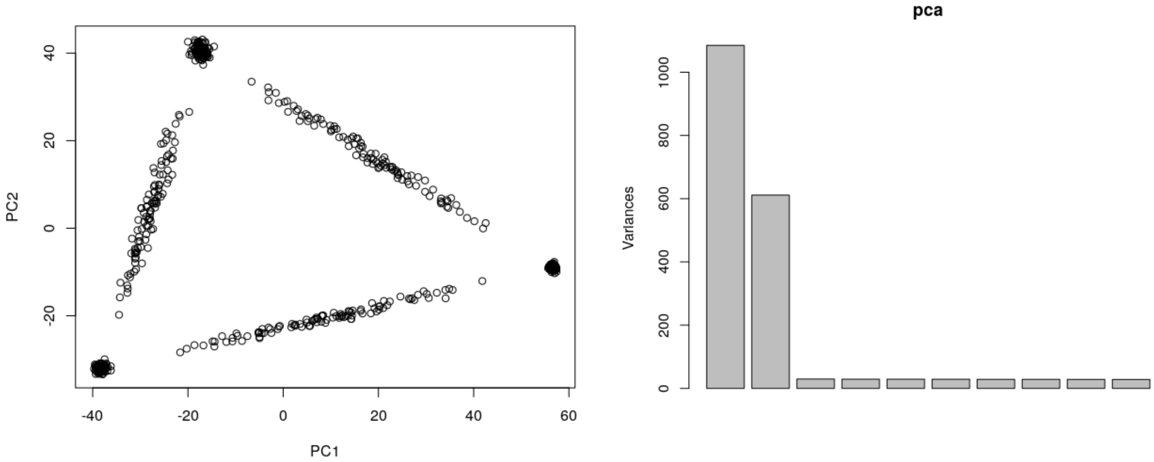
Figure 8: Projection of three populations and one admixed on to the first two PCs and corresponding eigenvalues



Three populations with F-values of $0.1, 0.5, 0.9$ and 100 individuals each were sampled. The admixed population derived the proportional weightings for its allele probabilities by sampling for each allele from a $Dir(8,8,8)$ distribution. 200 individuals were sampled for the admixed. For this simulation 10000 loci were simulated.

Figure 9 shows a similar scenario as Figure 8, whereby the only difference consists of a different admixture of the admixed individuals. In this example, admixed individuals are sampled from three different Dirichlet distributions. In turn one of the hyperparameters is set to zero, thus always one population does not partake in the admixture of an individual. As a consequence the individuals are spread along the edges of the simplex, also because the non-zero hyperparameters model a lower concentration than in Figure 1.
Again the eigenvalues feature two significant large eigenvalues, making the inference of the number of populations using solely the eigenvalues once again a simple task. The natural variance of the discretisation of the allele values through the Bernoulli distribution, which would allow individuals to lie outside the simplex, only has a neglectable marginal effect on the eigenvalues.

Figure 9: Projection of three populations and one admixed on to the first two PCs



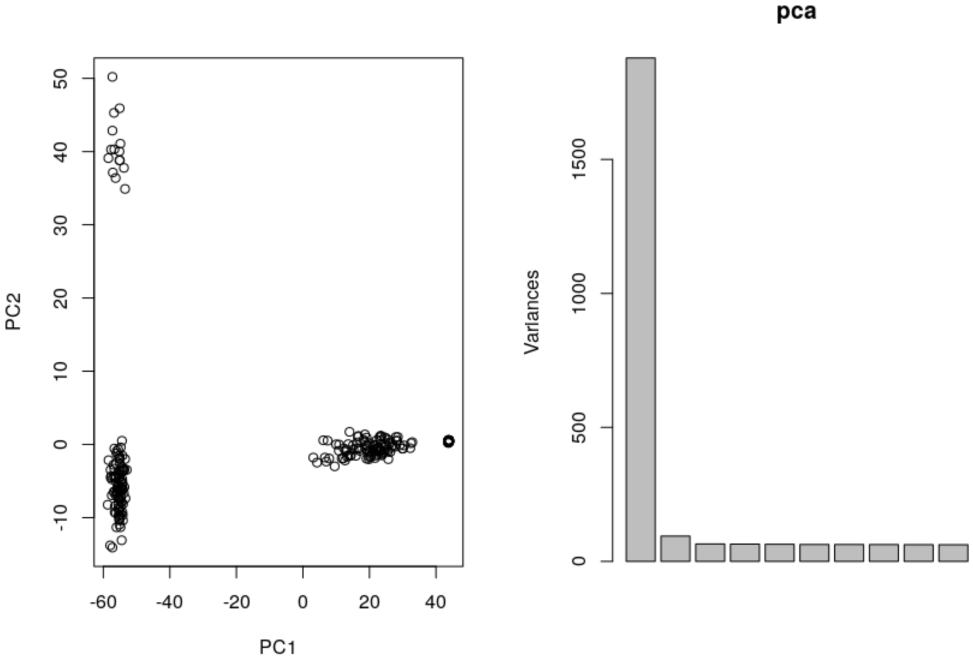
Three populations with F-values of $0.1, 0.5, 0.9$ and with 100 individuals each were sampled. Between each pair of population clusters lies an admixed population sampled from a Dirichlet with 5s and a 0 for not involved populations, which corresponds to a $\beta(5,5)$. Each admixed cluster holds 200 individuals. The simulation used again 10000 loci.

3.3.1 Possible Difficulties

The past examples only demonstrate fairly simple problem instances, for which even human recognition capabilities suffice. It is also possible to construct more difficult problem instances. Figure 10 is an example of such. The ploy is to simulate a setting where most of the variance is already captured by less than $k-1$ eigenvectors, hence making it more difficult to recognise the cut-off of significant and insignificant eigenvalues. In figure 10 there are two populations that have similar allele frequencies (F-values 0.05 and 0.1) and one population that has been subject to strong genetic drift and therefore is genetically completely distinct (F-value 0.99). Furthermore, most individuals are distributed over two of the populations that are far apart (have very distinct F-values). This urges the eigenvalues to capture the variance of the individuals of the two populations, since a bigger distance accounts for higher variance. Inversely, because the other population that holds a fewer amount of individuals is not very far away from the positioning of the first eigenvector, such that the second eigenvector, which has to orient itself perpendicular to the first, does not capture very much variance. Also, the low amount of individuals in the smaller population gives it a lower impact, it makes the orientation of the second eigenvector susceptible to the variance of the other populations or to any outliers and yields a lower variance.

The according graph with the biggest eigenvalues reveals the described dilemma. The first eigenvector accounts for almost all of the variance between the populations, rendering the other significant eigenvalue almost insubstantial and indiscernible from the other insignificant eigenvalues. In different scenarios the distance between the populations and the distribution of the individuals over the populations could be even more disadvantageous, further investigations of the F-value will be shown in the Results section.

Figure 10: Example of a difficult case



Three populations with F-values of $0.05, 0.01, 0.99$. The first population with the smallest F-value has 15 members, while the others have a 100 each. The mixture proportions of the admixed population were sampled from a $Dir(0, 10, 30)$. 10000 loci were simulated.

4 Boosting Decision Trees for Model Selection

Subsequently a concise overview of gradient boosting and decision trees is presented. For further details and idiosyncrasies of the methods the reader should consult more elaborate literature, like Trevor, Robert, and JH 2009.

4.1 Gradient Boosting

Gradient boosting is a supervised learning method for classification and regression, that iteratively builds up a linear combination of base learners to reduce an arbitrary differentiable loss function.

Let χ denote the input space. The task then is to approximate the function $f^*(x)$ that maps an arbitrary input $x \in \chi$ to the desired output $y \in \mathbb{R}$. An ensemble of M different basis learners $g_1, g_2, \ldots, g_M : \chi \to \mathbb{R}$ can be used to generalise over a training set $((x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N))$ of N training pairs in a linear fashion as following:

$$f(x) = \sum_{m=1}^{M} \phi_m g_m(x)$$

where $\phi_i \in \mathbb{R}$ are the weights for each basis learner. Linear models for a set of given basis learners can be fitted via conventional methods such as least squares, lasso, ridge

(Bishop 2006). $f(x)$ can be used as an approximation for various tasks like regression, classification (by for example using a threshold).

The model is limited by the explicit choice of the base learners. Instead an algorithm that finds the best base learners from a hypothesis space $\mathscr{H}$ would increase the adaptive potential of the model. Thus, for a given differentiable loss function $l(x, f(x))$, an algorithm should find the best M base learners $h_1, h_2, \dots, h_m \in \mathscr{H} : \chi \to \mathbb{R}$ and corresponding weights such that the empirical risk is minimised:

$$\underset{\phi_i, h_i}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} l\big(y_i,\ \sum_{m=1}^{M} \phi_m h_m(x_i)\big)$$

Solving this task is an optimisation problem usually beyond practicability. A common optimisation technique is too update the parameters through the use of gradient descent. Several problems arise in the case of differentiating over the hypothesis space:

- Firstly, the hypothesis space would need to be parametrised with finite dimensions. In the face of a countless manifold of possible base learners a rather daunting task. One is far better off by confining the possibilities of base learners, by choosing a subset $H \subset \mathscr{H}$ like decision trees or neural networks.
- Many modelling frameworks such as neural networks (e.g. number of hidden layers) and decision trees (e.g. tree depth) have to be parametrised at least partly discretely. Of course all discrete model features could be fixed to a constant, but that would greatly degrade the modelling capabilities.
- Lastly, some base learners do not model a differentiable functions. For e.g. decision trees model functions that possess jump discontinuities.

Gradient boosting algorithms, firstly developed and described by Freund and Schapire 1997; J. H. Friedman 2001; J. H. Friedman 2002, circumvent the necessity of differentiability by growing the ensemble of base learners iteratively to minimises the empirical risk.
The backbone of an gradient boosting algorithm is constituted by forward stagewise additive modelling, which works as following: Let $H \subset \mathscr{H}$ be the chosen set of possible base learners.

1. Initialise with constant like $f_0(x) = 0$
2. For each stage $m \in 1, \dots, M$:
   (i) solve
   $$\underset{\phi_i, h_i}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} l\big(y_i,\ f_{m-1}(x_i) + \phi_m h_m(x_i)\big)$$
   (ii) Set $f_m(x) = f_{m-1}(x) + \phi_m h_m(x)$

The optimisation step only possess closed form solution for a limited amount of loss functions, like L2-loss or exponential loss. More information is provided in J. Friedman, Hastie, Tibshirani, et al. 2000. To expand the optimisation step to any arbitrary differentiable loss function, the numerical optimisation via gradient boosting is used.

The optimisation procedure fixates at stage m the current estimates made by $f_{m-1}(x)$ of the training data in a vector $\mathbf{f}_m = [f_{m-1}(x_1), f_{m-1}(x_2), \dots f_{m-1}(x_N)]^T$.

The loss function then can be reformulated as:

$$L(\mathbf{f}) = \sum_{i=1}^{N} l(y_i, \mathbf{f}_i)$$

Then the gradient of the loss function is calculated w.r.t $\mathbf{f}$.

$$\begin{aligned}\hat{h}_m &= \nabla_{\mathbf{f}_m} L(\mathbf{f}_m) \\ &= \left[\partial_{\mathbf{f}_1} l(y_1, \mathbf{f}_1), \dots, \partial_{\mathbf{f}_N} l(y_N, \mathbf{f}_N)\right]^T\end{aligned}$$

Like in conventional gradient descent algorithms the model is adjusted in a manner that the empirical risk is minimised along the direction of steepest descent. The direction of steepest descent corresponds to the negative of the gradient, which is $-\hat{h}_m$. Unlike differentiable base learners, that can adjust their parameters by gradient descent through the chain rule, non-differentiable have to resort to a different strategy.
The embedding in the forward additive model allows for the negative gradient $-\hat{h}_m$ to be approximated directly by a new base learner. The objective of the new base learner $h_m(x)$ consequently concludes therein to approximate

$$h_m(x) \approx -\hat{h}_m$$

as well as possible. A possible approach would be to train a base learner on the training data, but where the labels are exchanged by $-\hat{h}_m$.
Intuitively, a base learner $h_m(x)$ should be considered as an approximate step in the direction of steepest descent of the empirical risk. Following this setup, the weight $\phi_m > 0$ can be considered as the corresponding step size to be adjusted to one's taste. As a conclusion the iterative construction of the final linear model with gradient boosting

$$f(x) = \sum_{m=1}^{M} f_{m-1}(x) + \phi_m h_m(x)$$

is a sequence of approximated gradient descent steps towards a minimum of the empirical risk.

4.2 Decision Trees

Typical decision trees are a supervised learning method that solve a regression or classification problem by segmenting the feature space in to distinct regions, whereby all data points lying in the same region are assigned the same value by the tree.

Let $(x_1,y_1),(x_2,y_2),\dots,(x_N,y_N)$ be the training data. The input dimension of a point x_i is D.

Suppose a tree T partitions the featurespace $X_1,X_2,\dots,X_D$ into M regions $R_1,R_2,\dots,R_M$. The response function outputed by T is given by:

$$f_T(x) = \sum_{m=1}^{M} c_m I\big(x \in R_m\big)$$

where I is an indicator function signalling if the input x is part of a particular region. c_m is the response for a region R_m. The optimal value of c_m depends on a chosen loss function that should be minimised w.r.t c_m over all the training points assigned to region R_m. As an example, for square loss in a regression setting this would accord to the empirical average of the label values from the training points lying in the region R_m.

A far more challenging optimisation task is finding the optimal tree that partitions the feature space into M regions, for a given loss function l.

$$\underset{R_m,c_m}{\operatorname{argmin}} \sum_{i=1}^{N} l\big(y_i, \sum_{m=1}^{M} c_m I\big(x_i \in R_m\big)\big)$$

The possibilities of partitioning feature space grows exponentially with the number of features, rendering the encounter of an optimal tree in most cases computationally infeasible.

As an alternative, a greedy approximation approach can be used. The greedy algorithm chooses the best dimension X_d and splitting point s that minimises the loss for the two new regions that arise (w.l.o.g. only binary trees will be considered). The two new regions are defined as:

$$R_1(d,s) = \big\{X \,|\, X_d \leq s\big\} \qquad R_2(d,s) = \big\{X \,|\, X_d > s\big\}$$

The feature space is divided by a plane that is orthogonal to the axis corresponding to feature X_j. The plane cuts the value of s on that axis.
The optimisation objective thus reduces to the choice of positioning the best partitioning plane orthogonal to an axis:

$$\underset{s,d}{\operatorname{argmin}} \Big[\underset{c_1}{\operatorname{argmin}} \sum_{x_i \in R_1(d,s)} l\big(y_i,x_i\big) + \underset{c_2}{\operatorname{argmin}} \sum_{x_j \in R_2(d,s)} l\big(y_j,x_j\big)\Big]$$

The optimisation step yields, for a naive implementation that checks for every dimension the splitting value of a plane between every two neighbouring data points, a worst case running time of $\mathcal{O}(D \cdot Nlog(N))$ ($Nlog(N)$ for sorting feature values), which is computationally much more feasible. However, between two neighbouring data points x_1 and x_2 there are infinitely many positions (assuming a continuous variable) to place the splitting plane that evaluate to the same empirical risk. As a convention, the splitting value s is placed in the middle of x_1 and x_2, when they are projected on the axis X_d. The reasoning is that no further assumptions are to be added through the positioning of the plane and since the splitting of the feature space can be viewed as a Bernoulli experiment (data point either lies left or right of plane) the Bernoulli distribution with expected value of 0.5 is the maximum entropy Bernoulli.

After finding the best split, the procedure is repeated on the newly constructed regions until the desired depth of the tree is reached. Regression and classification trees differ from one another only through the selection of a different loss function. For regression standard loss functions like square loss would be reasonable choices. Classification trees have revealed better results for loss function that reward node purity, so to which degree does a node hold data points of a single class. Such loss functions are for example the Gini index or cross entropy loss for classification. From an information theory perspective, node purity leads to a greater reduction in entropy.

Growing a too big tree T_0 is susceptible to overfitting. A regularisation technique that aids in the construction of a well generalising tree is cost-complexity pruning. The goal of pruning is to find a sub-tree $T \subseteq T_0$, by conflating all hierarchically lower nodes that lead off an internal node into that internal nodes, that minimises:

$$C_\alpha(T) = \sum_{i=1}^{N} l\big(y_i, f_T(x_i)\big) + \alpha|T|$$

where $f_T(x)$ is the estimate of Tree T for input x and $|T|$ denotes the number of terminal nodes of T. The tuning parameter $\alpha \geq 0$ punishes larger more complex trees according to its value. It resembles the regularisation term of ridge regression.
The optimal sub-tree corresponding to a particular tuning parameter T_α can be found using weakest link pruning. Weakest link pruning conflates those terminal nodes into an internal node to which the terminal nodes are all adjacent, such that the empirical risk increases minimally. Continuing with this procedure until only the root stub remains gives a sequence of subtrees $T_1, T_2, \ldots, T_n$ in which with a probability of 1 the optimal subtree T_α can be found (Breiman et al. 1984). Cross validation can be used to find the optimal value for α.

Decision trees exhibit high variance, meaning that completely different splits occur and thus the output prediction rules change considerably when minor changes are added to the training data. The reason resides inherently with how the splits are chosen in a greedy fashion. For example, two promising features could reduce the loss almost equally much, but just the best of both is considered for the next split. Adjusting the training data slightly by for example adding new data points could possibly change the value of the loss function enough to choose a different feature for a split, the hierarchical nature consequently propagates the difference further down the tree. This behaviour reveals that the confinement to the greedy perspective when constructing a tree to some degree neglects the goal of generalisation in return for tractability.

Remedies that address model stability involve the introduction of bias. The bias-variance trade off possess an eminent role in machine learning as it is a principle that is prominent for many models. In summary, it describes the forfeit of expected accuracy in return for decreasing the variance of an estimated parameter when the training sample is being varied. Creating an ensemble of decision trees is a widely used and fruitful approach. Ensemble approaches for decision trees include bagging, random forests and as well gradient boosting. Several further refinements improve the quality of an ensemble tree models, including randomly masking different features and training data entries for each tree, as this generates differing trees that place their splits differently and therefore deliver more uncorrelated predictions.

5 Results

For a beginning, a training data set is generated composed of $30,000$ synthetical instances, with the population sizes evenly distributed ranging from $2-16$. The number of loci was uniformly randomly varied ranging from $2,000$ to $40,000$ as well as the sample size, which could include from 100 to $5,000$ individuals. The $F-values$ of the populations are sampled from a uniform distribution.
The eigenvalues $\lambda_1 \dots \lambda_K$ for the summary statistics are determined by first calculating the singular value decomposition of the simulated genotype matrix and then setting $\lambda_i = s_i^2/(n-1)$ where $s_1, \dots, s_K$ are the singular values.
Since it is only necessary to calculate a certain number of eigenvalues and the genotype matrix is typically sparse, an SVD-solver based on Lanczos Bidiagonaliziation provides a computationally efficient choice (Golub and Kahan 1965).
The boosted model trained with the generated data set is used first to investigate some of the inference limits of the model, as well as the implicit clustering assumptions of the generative model. For this purpose the boosted model tries to solve different synthetic data, that are generated with different F-values.
As expected table 1 reveals that the lower the F-value, which accounts for the distance from the ancestral allele frequencies, the closer the population clusters will end up, making them harder or even impossible to distinguish from one another. The trained model is able to detect structure in the eigenvalues from F-values starting around 0.005.

F-values	**0.003**	**0.005**	**0.01**	**0.02**	**0.05**	**0.1**	**0.2**	**0.5**
MSE	135.80	135.80	134.42	118.16	62.62	20.98	4.02	0.02

Table 1: The mean squared error of the model with uniform F-value prior for generated data with different F-values. In each data set all populations had the according same F-value. 50 data set with different values of K for each F-value were used.

Two problems instances, figure 11 and 12, exemplify these empirical results. Both are synthetically generated with each instance having the three populations with same population sizes and originating from the same ancestral allele frequencies (all frequencies 0.5). The data is projected on to the first two principal components, which should, according to the discussed theory, suffice to make out the clusters. As one can see, the clusters of the data with an F-value of 0.005 are less discernible than for the data with 0.01. Of course, other factors, such as sample size and dimension of the data set influence the results.

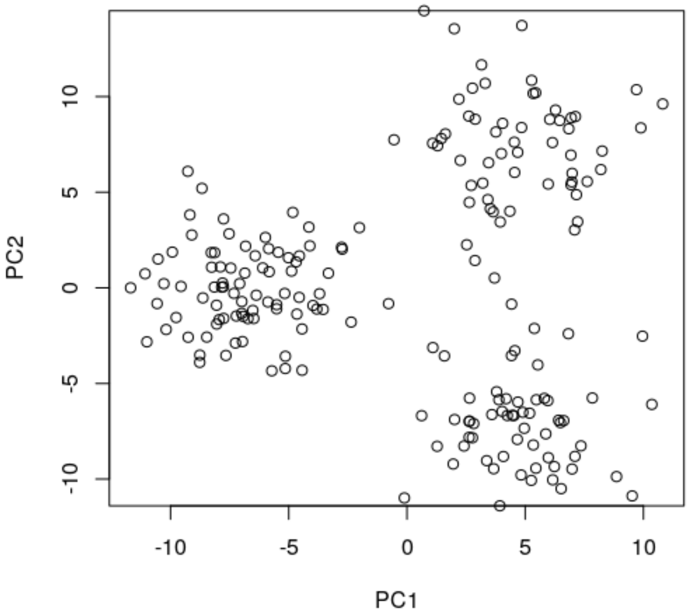


Figure 11: Possible generated population scenario for an F-value of 0.005. Datapoints projected on the first two PCs. Number of populations is supposed to be 3.

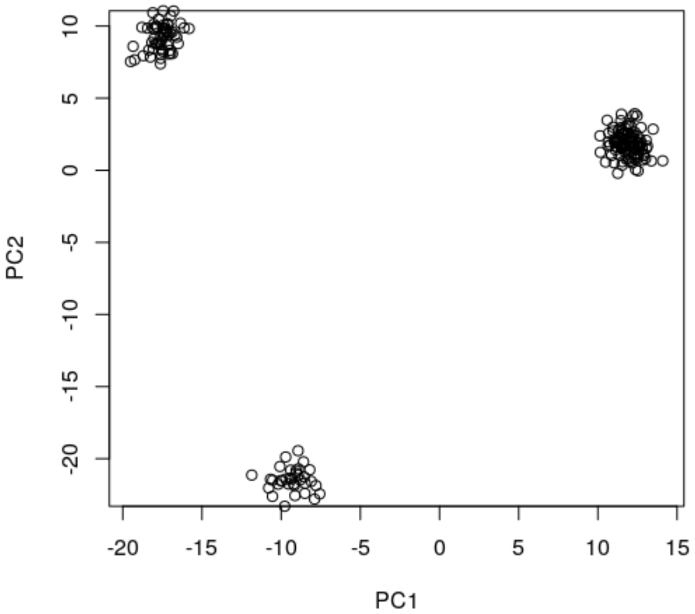


Figure 12: Almost same scenario settings, just the data was generated with an F-value of 0.01

Since a minimum F-value is necessary to generate detectable population structure, new synthetic data sets are generated that only allow an F-value higher than 0.003. One data set places a $\beta(1,3)$ prior and the other a $\beta(0.1,2)$ over the F-values. The priors will sample lower F-values, where by $\beta(0.1,2)$ does so the most extreme. Both have $30,000$ synthetic training samples. The intention is to emphasize trickier problem instances while training the model, with the hope that the models make more refined choices when confronted by more intricate problem instances.

5.1 Model Comparison

Subsequently three gradient boosting models with uniform, $\beta(1,3)$, $\beta(0.1,2)$ F-value priors, the Tracy-Widom p-value method, and the sNMF method are compared. The gradient boosting models were all trained with the same model parameter values and reached all about an average 96% error rate for a five fold cross validation with their respective data sets. The Tracy Widom p-value is set to 0.05. The sparse NMF algorithm is run five times (regularisation parameter α set to $1,000$) for a possible interval of K values and the K value with the lowest average cross entropy is chosen. For more information on the methods the reader is referred to section 1.2.
A first comparison is done with synthetic data generated by the prior described model. The F-value for the test data is chosen from a uniform distribution.

K	2	3	4	5	6	7	8
Uniform	0	0.1	0	0.1	4.9	0.1	0
$\beta(1,3)$	0	0	0	0.1	0	0.1	0.1
$\beta(0.1,2)$	0	0.1	0.4	0.1	1.6	0.1	0.1
TW	57.6	10	14.4	3.6	36.1	129.6	230.4
sNMF	0.1	1.6	0.9	0.4	0.9	0.4	0.4

Table 2: The mean squared of error for each method. For each K 10 data sets were generated. F-values were sampled from a uniform distribution.

To no surprise, as seen in table 2, the boosted models fare very well. The TW-method, despite the mathematical elegance of its origin, remains uncompetitive and inflexible until at least a heuristic for choosing the p-value for each problem instance independently is developed. The sNMF approach yielded also very competitive results, which is only mildly surprising, since it performs similar on the estimation of K as the likelihood method of Structure/Admixture that use a similar generative model as the one the data is generated from (Frichot, Mathieu, et al. 2014).

5.1.1 Real World Instances

To gain a better view of the inference decisions made by the boosted models, they are tested and compared on two real world data sets.
The data set originates from the Harvard Human Genome Diversity Project (`ftp://ftp.cephb.fr/hgdp_v3/`), it contains the information of $10,664$ SNPs from 418

human samples from Central- and East-Asia (Frichot, Schoville, et al. 2012). Figure 13 visualises a possible admixture attribution for $K = 4$.

From the display of the Eigenvalues (figure 14) one can observe that detectable population structure should be present. The question however remains how many populations are too be found on the highest clustering level. Inspecting the eigenvalues, most of the signal is captured by the first eigenvector, thus 2 would be a reasonable guess, but also $3, 4, 5$.

uniform	$\boldsymbol{\beta(1,3)}$	$\boldsymbol{\beta(0.1,2)}$	**sNMF**
14	8	5	3

Table 3: The estimates of each method for the real world Asia data set.

Although the true number of populations remains vague and dependable of what assumptions are made, the estimates of the boosted models seem, with regard of the prior described theory, too sensitive. Most of the signal was capture by the first to first three eigenvectors. Only the model with the $\beta(0.1, 2)$ prior outputs the reasonable result of 5. The estimate 3 of the sNMF model, seems more suitable. The results are depicted by table 3.

The other real world data contains samples of the plant species *Arabidopsis Thaliana*. $52,001$ SNP of the first chromosome were extracted from a larger data set, whereby $1,096$ across Europe where taken (François 2016). Figure 15 illustrates a possible admixture interpolation with the results of the sNMF method for $K = 5$. Population structure for *Arabidopsis thaliana*, as indicated in the Eigenvalues (figure 16), possess a more complex population structure. The species reproduces via seeds that gradually disperse over generations. Thus, the genetic differences changes are more of a continuous nature than clear cut clusters. The Eigenvalues decay accordingly also in a more continuous fashion. Making a prediction about the number of populations appears therefore quite intricate.

uniform	$\boldsymbol{\beta(1,3)}$	$\boldsymbol{\beta(0.1,2)}$	**sNMF**
7	4	3	10

Table 4: Estimates of the methods for the *A. Thaliana* data set.

The results are shown in table 4. From looking at the eigenvalues, various possible values of K seem reasonable. The value of 10 from the sNMF estimate appears high, but nonetheless possible as an even the lower eigenvalues display that their corresponding eigenvectors still capture some of the signal.
Considering the boosted models, it appears that the estimates with priors are more "conservative", where by the more "extreme" $\beta(0.1, 2)$ prior the most conservative is. A reason might be, as already pointed out, the models are more observant towards difficult cases where significant eigenvalues are not much larger than insignificant ones. As a consequence, the model will act more reserved when confronted with cases, where the decay of eigenvalues is of continuous nature.

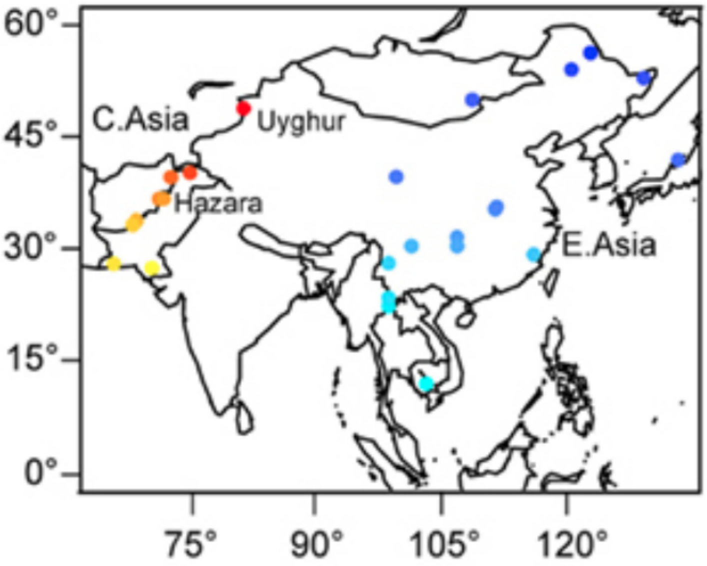


Figure 13: Possible admixture illustration for $K = 4$ for the real world Asia data set. Source: Frichot, Schoville, et al. 2012

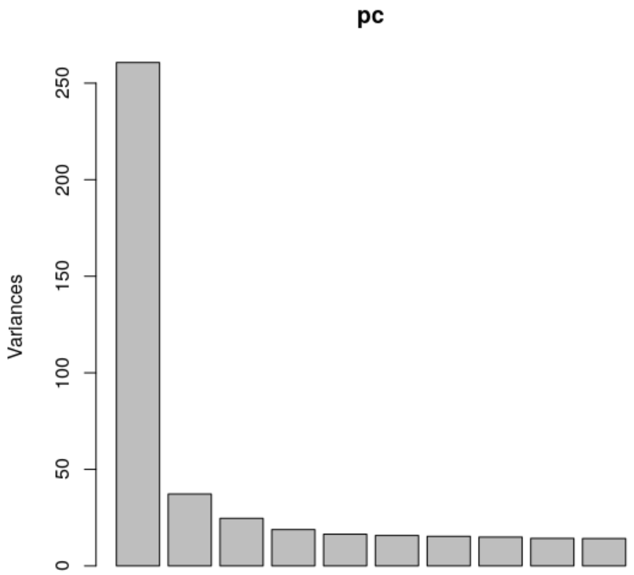


Figure 14: Eigenvalues of the real world Asia data set.

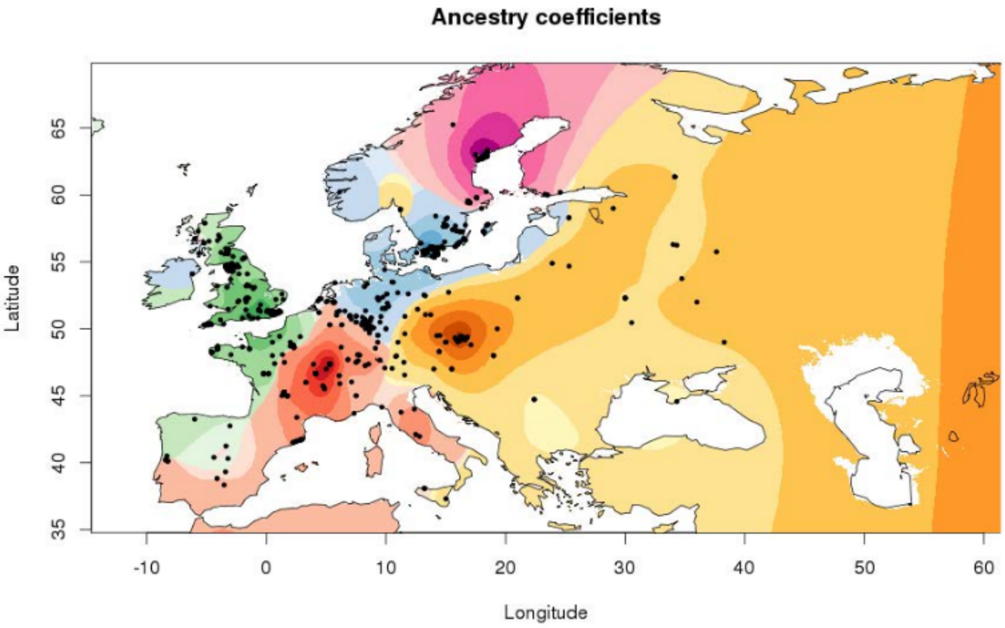


Figure 15: Interpolation for a possible population admixture assignment for $K = 5$ of the *A. Thaliana* data set. Source: François 2016

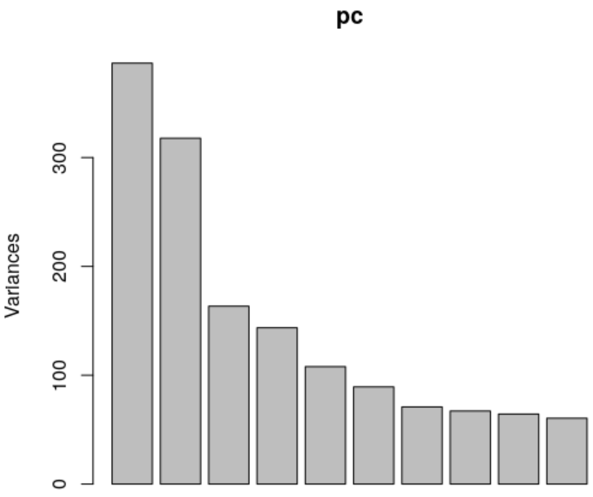


Figure 16: Eigenvalues of the *A. Thaliana*

6 Discussion

To further evaluate the quality of predictions made by the boosted models, more real world instances should be investigated. However, since the ground truth is not known and each real world problem instance needs to be investigated separately to at least contain the range of reasonable K values. This explains the difficulty of compiling a real world training set.

The proposed boosted model still displays some difficulties concerning real world instances, as in the case of the real world Asia data set. Setting appropriate priors is not a complete remedy for a profound structural problem within the model. I suspect that the main problem lies inherently in the simplicity of the generated training set. In particular, the assumption that de Finetti's Theorem holds, meaning individuals from the same population are exchangeable. This leads to the noise being simple and easily separable from the desired signal captured by the significant eigenvalues.

Real world instances on the other hand exhibit a more complex structure, especially in the sense that the population structure is usually hierarchical, possibly to a very low level, if one considers the possibility of sampling individuals from the same family (or close relatives). Separating the desired signal (population structure on the highest hierarchical level) becomes thus more challenging as the for the task insignificant eigenvalues also convey some structure. Exactly this difference is observable when comparing the eigenvalues of synthetic data sets and real world data sets. The insignificant eigenvalues of the synthetic data sets look flat and do not convey any more structure.

A possibility to make estimates more viable when dealing with real world instances could therefore be, to alter the generative model in a way that it produces a hierarchical population structure. This could be implemented by reapplying the algorithm recursively in such a fashion that for every population, further subpopulations (with decreasing F-values) are determined, until the single individual is reached. Of course further prior assumptions about the subpopulations would have to be made, however since the task is to solely provide some more complex substructure, which is to be exhibited in the decay of the lower eigenvalues, conservative assumptions should suffice.

Further factors that could influence the quality of predictions include the dimension ratio l/n (number of loci l, number of samples n) of typical genotype matrices. Usually $l/n >> 1$ holds, resulting in the standard covariance estimator being ill-suited (Ledoit and Wolf 2004; Schäfer and Strimmer 2005).

Also the generative model possesses various hyperparameters that can be tuned according to assumptions made about the nature of genotype data (F-value, minimum cluster size, number of samples, number of loci, etc.). Especially the assumptions made about the generated cluster should be investigated further. What clustering qualities are necessary such that the number of clusters on the highest hierarchical level is displayed by the significant eigenvalues?

As a conclusion, the boosted model has proven to be a scalable method, even for limited computational resources, to estimate the number of populations in SNP genotype data. Most other methods are very elaborate, for solely estimating the number of populations and require several extensive runs that might not be possible with limited computing power. The quality of the estimates have shown to be satisfactory, but could possibly be greatly improved by the points stated above.

References

Balding, David J (2003). “Likelihood-based inference for genetic correlation coefficients”. In: *Theoretical population biology* 63.3, pp. 221–230.

Bishop, Christopher M (2006). *Pattern recognition and machine learning*. springer.

Blei, David M, Andrew Y Ng, and Michael I Jordan (2003). “Latent dirichlet allocation”. In: *Journal of machine Learning research* 3.Jan, pp. 993–1022.

Blum, Michael GB et al. (2013). “A comparative review of dimension reduction methods in approximate Bayesian computation”. In: *Statistical Science* 28.2, pp. 189–208.

Breiman, Leo et al. (1984). “Classification and regression trees. Wadsworth Int”. In: *Group* 37.15, pp. 237–251.

Bryc, Katarzyna, Wlodek Bryc, and Jack W Silverstein (2013). “Separation of the largest eigenvalues in eigenanalysis of genotype data from discrete subpopulations”. In: *Theoretical population biology* 89, pp. 34–43.

Cover, Thomas M and Joy A Thomas (2012). *Elements of information theory*. John Wiley & Sons.

Crow, James F, Motoo Kimura, et al. (1970). “An introduction to population genetics theory.” In: *An introduction to population genetics theory.*

Csilléry, Katalin et al. (2010). “Approximate Bayesian computation (ABC) in practice”. In: *Trends in ecology & evolution* 25.7, pp. 410–418.

Donath, WE and AJ Hoffman (1973). “Lower bounds for the partitioning of graphs”. In: *IBM Journal of Research and Development* 17.5, pp. 420–425.

Evanno, Guillaume, Sebastien Regnaut, and Jérôme Goudet (2005). “Detecting the number of clusters of individuals using the software STRUCTURE: a simulation study”. In: *Molecular ecology* 14.8, pp. 2611–2620.

Falush, Daniel, Matthew Stephens, and Jonathan K Pritchard (2003). “Inference of population structure using multilocus genotype data: linked loci and correlated allele frequencies”. In: *Genetics* 164.4, pp. 1567–1587.

Fiedler, Miroslav (1973). “Algebraic connectivity of graphs”. In: *Czechoslovak mathematical journal* 23.2, pp. 298–305.

François, Olivier (2016). “Running structure-like population genetic analyses with R”. In: *R Tutorials in Population Genetics U Grenoble-Alpes*, pp. 1–9.

Freund, Yoav and Robert E Schapire (1997). “A decision-theoretic generalization of on-line learning and an application to boosting”. In: *Journal of computer and system sciences* 55.1, pp. 119–139.

Frichot, Eric, François Mathieu, et al. (2014). “Fast and efficient estimation of individual ancestry coefficients”. In: *Genetics* 196.4, pp. 973–983.

Frichot, Eric, Sean D Schoville, et al. (2012). “Correcting principal component maps for effects of spatial autocorrelation in population genetic data”. In: *Frontiers in genetics* 3, p. 254.

Friedman, Jerome H (2001). “Greedy function approximation: a gradient boosting machine”. In: *Annals of statistics*, pp. 1189–1232.

– (2002). “Stochastic gradient boosting”. In: *Computational statistics & data analysis* 38.4, pp. 367–378.

Friedman, Jerome, Trevor Hastie, Robert Tibshirani, et al. (2000). “Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)”. In: *The annals of statistics* 28.2, pp. 337–407.

Gillespie, John H (2004). *Population genetics: a concise guide*. JHU Press.

Golub, Gene and William Kahan (1965). “Calculating the singular values and pseudo-inverse of a matrix”. In: *Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis* 2.2, pp. 205–224.

Harter, Abigail V et al. (2004). “Origin of extant domesticated sunflowers in eastern North America”. In: *Nature* 430.6996, p. 201.

Hartl, Daniel L, Andrew G Clark, and Andrew G Clark (1997). *Principles of population genetics*. Vol. 116. Sinauer associates Sunderland.

Jaynes, Edwin T (1957). “Information theory and statistical mechanics”. In: *Physical review* 106.4, p. 620.

Ledoit, Olivier and Michael Wolf (2004). “A well-conditioned estimator for large-dimensional covariance matrices”. In: *Journal of multivariate analysis* 88.2, pp. 365–411.

Meila, Marina and Jianbo Shi (2001). “A random walks view of spectral segmentation”. In:

Ng, Andrew Y, Michael I Jordan, and Yair Weiss (2002). “On spectral clustering: Analysis and an algorithm”. In: *Advances in neural information processing systems*, pp. 849–856.

Nunes, Matthew A and David J Balding (2010). “On optimal selection of summary statistics for approximate Bayesian computation”. In: *Statistical applications in genetics and molecular biology* 9.1.

Patterson, Nick, Alkes L Price, and David Reich (2006). “Population structure and eigenanalysis”. In: *PLoS genetics* 2.12, e190.

Pritchard, Jonathan K, Matthew Stephens, and Peter Donnelly (2000). “Inference of population structure using multilocus genotype data”. In: *Genetics* 155.2, pp. 945–959.

Rosenberg, Noah A, Terry Burke, et al. (2001). “Empirical evaluation of genetic clustering methods using multilocus genotypes from 20 chicken breeds”. In: *Genetics* 159.2, pp. 699–713.

Rosenberg, Noah A, Jonathan K Pritchard, et al. (2002). “Genetic structure of human populations”. In: *science* 298.5602, pp. 2381–2385.

Schäfer, Juliane and Korbinian Strimmer (2005). “A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics”. In: *Statistical applications in genetics and molecular biology* 4.1.

Shi, Jianbo and Jitendra Malik (2000). “Normalized cuts and image segmentation”. In: *Departmental Papers (CIS)*, p. 107.

Trevor, Hastie, Tibshirani Robert, and Friedman JH (2009). *The elements of statistical learning: data mining, inference, and prediction*.

Von Luxburg, Ulrike (2007). “A tutorial on spectral clustering”. In: *Statistics and computing* 17.4, pp. 395–416.