A Comparison of Clustering Algorithms and Identification Methods on Genomic Variants to Compare Modern Dogs and Wolves

Bibiana Elisabeth Seng

April 2019

Abstract

An important issue that arises in population genetics and anthropology is the origin of dog domestication. Studies on this topic use Principal Component Analysis to de-correlate genetic data from their samples and identify groups based on the geographic region of origin. However, this method only uses the first 2 Principal Components and relies heavily on visual interpretation - furthermore, it does not allow for identification of similar subjects based on non-geographic criteria. This study discusses different clustering methods used, such as k-means and DAPC, and analyzes their pros and cons on a set of genetic variants from over 5,000 different dogs and 4 wolves from Azerbaijan. The goal of this study is to determine if there are additional insights that can be gleaned from using different clustering methods on PCA data that can either use additional Principal Components and/or algorithmically determine groups.

1 Introduction

The origin of dog domestication is a complicated topic. There have been many studies, especially within the last 10 years, that give conflicting analyses and results when trying to solve this problem. For example, a recent study in 2017 [2] concluded that dogs were domesticated between 20,000 and 40,000 years ago, though does not comment on where this occurred. However, a paper published just last year, in 2018 [11] postulated that concrete evidence for domestication puts the earliest known domesticated dog at 13,000 years ago. This is a significant temporal difference, as in the first case it could be argued that dogs were domesticated when humans were hunter-gatherers, to help with hunting; meanwhile, in the second case it could be argued that dogs were domesticated during the Neolithic era, when humans adopted agriculture and farming, to help herd other animals or to protect crops.

In addition to the temporal differences, there is debate on where precisely dogs were first domesticated. Some papers [23] conclude that dog domestication occurred in Central Asia, and that from there, domesticated dog breeds migrated to the rest of Eurasia. Meanwhile, others [10] suggest that, rather than a single domestication, dog domestication occurred twice: once in Western Eurasia (i.e. Europe), and once in Eastern Eurasia (i.e. the Middle East, China, and Russia). The Eastern Eurasian dogs then migrated over to Europe. Based on both anthropological and genetic evidence, there are four main geographic areas considered for dog domestication (which are discussed further in [2]):

1. Europe (i.e. Germany),
2. Central Asia (i.e. Kazakhstan),
3. The Middle East (i.e. Iran/Iraq),
4. East Asia (i.e. China)

From the conclusions drawn by the above studies, it is clear that, while there is an agreed-upon range of dates and locations, there is no consensus on the timescale nor *where* dogs were first domesticated.

This conflicting nature is due to the complex history of dog genetics, discussed further in the next section. Despite this challenge, it is imperative to figure out where dogs were first domesticated, because dogs were the very first animal to be domesticated [16]. This domestication paved the way for further advances in agriculture; through learning how to domesticate the dog, Neolithic (late Stone Age, approximately 12,000 years ago) humans were able to domesticate more animals such as cows and horses. This helped shape early farming communities in the Neolithic era of humanity.

While the original intent of this study was to work with a sample of a Neolithic dog genome [2] and compare it to modern dog genomes, this proved difficult to accomplish. This is because the format the Neolithic genome was stored in did not contain the necessary information on genetic variants (a copying error that occurs when genomes duplicate during cell replication) required to convert it to a variant file, and a proprietary sequence from Illumina (a sequencing company) would be required to find the genetic variants and accurately compare them to modern dogs.

Therefore, rather than use a sample from the Neolithic era, the study used samples obtained from wolves instead. Wolves, in particular gray wolves, are the closest living relatives to modern domesticated dogs. While there is evidence to support the hypothesis that the individual breed of gray wolves that dogs were domesticated from is now extinct, there is a clear branch between the ancestry of dogs and gray wolves, and the ancestry of other wolves that confirms dogs were domesticated from gray wolves [9]. Thus, modern gray wolves can still be used as a surrogate for ancient dogs when genomic data from ancient dogs is difficult to obtain. By using samples of gray wolves from Eurasia [21], the wolves' genetic data — in particular, their variants — can be compared to the

genetic variant data obtained from many modern domesticated dogs [23]. An example of how this genetic variant data is commonly stored can be found in the Appendix.

The question of how to actually compare genetic variants still remains, however. The first step biostatisticians take when comparing how genetic material varies across different subjects is to perform Principal Component Analysis (PCA) [19] on the genetic variant data to remove correlation between all of the variants and samples, and obtain a "birds-eye-view" of the data through the resulting principal component plots. However, current applications of PCA in published research rely on two-dimensional plots of the first two principal components and color-coding species based on their region of origin to visually identify clusters and outliers [23] [2]. This allows room for human error in erroneously identifying groups or patterns in the data where there are none, particularly when only taking into account the first 2 principal components. In addition, this does allow for a visualization of clusters based on geographic region of origin, but other aspects may be ignored in the process.

This study aims to compare different algorithms in clustering the PCA data to see if any algorithmic approaches could glean additional insights from the data. For example, some methods may be able to account for more variance in the entire dataset than simply the first 2 principal components, which may lead to grouping data together that wouldn't have been grouped together in a traditional 2-dimensional PCA plot. By projecting the genetic samples of wolves onto the existing data of modern dogs, relationships between the variants are identified and connections are established between wolves and dogs. Then, by implementing these new algorithms, outliers and points of interest are identified for use in future analysis without relying as much on visual identification of metrics of interest. In particular, this would be able to give insight as to how close or how far away the variant data in wolves would be compared to dogs. This would potentially allow for more data points to be used.

This paper is divided into the following sections:

- Section 2 is devoted to the history of dogs and wolves, and how they evolved to where they are today. It defines terms used in the rest of this paper.
- Section 3 contains a literature review on the main papers referenced in this study. It briefly discusses the original paper with the Neolithic sample, but the bulk of this section discusses the nature of the 2 datasets used in this study.
- Section 4 contains a literature review of the different statistical methods used. In particular, it discusses Principle Component Analysis (PCA). This section also discusses other algorithms that were chosen in this study.
- Section 5 contains the results obtained from using PCA, distance measures and other clustering data, as well as a discussion on those results.

- Section 6 contains the conclusion, summarizes the results found in Section 5, and where future work lies. It also contains potential uses for the results of this study.
- The Appendix contains additional information, such as comments on the code used and descriptions of file types.
- The References contain all the references I used and consulted when performing this study.

2 History of Dog Genetics

Before proceeding with the history of dogs, some terms need to be defined. Borrowing from Larson et. al. [17], the term *breed* refers only to modern, officially recognized pure-bred dog breeds, while *mutt* refers to cross-bred dogs from officially recognized breed dogs. Both terms refer to fully domesticated dogs. The term *village dog* refers to semi-domesticated dogs whose ancestors are indigenous to a given geographic area.

The history of dogs has been fraught in recent years, but was very steady at the beginning. Around 8,000 B.C., various domestic dog classes began to take shape. These dog classes included scouting dogs, giant dogs, and shepherding dogs, and helped humans by taking on various tasks relating to their dog class, such as herding other animals, and aiding in the development of domestication and training animals [17].

However, the genetic history of dogs became much more complicated to parse during the 1800s. In Ireland, for example, Irish Wolfhounds hunted wolves to extinction by 1786. With no more wolves to hunt, the need for Irish Wolfhounds dropped significantly; by 1840 the Irish Wolfhound was all but extinct. The breed was eventually restored, but they were saved via breeding the remaining few Irish Wolfhounds with aesthetically similar, but genetically different, dogs. This ensured common aesthetic similarities, such as body shape, among the new Irish Wolfhounds, but introduced new DNA and gave rise to variants that made these new Irish Wolfhounds genetically quite different from their ancestors. Thus, the modern genetic lineage of Irish Wolfhounds is quite complicated and contains different genetic material and mutations than a true, pre-1840s Irish Wolfhound.

This concept of recovering breeds for aesthetics was used quite often, particularly in the aftermath of World War 2, where many dog breeds only had 10 remaining population members [17]. During the aftermath, many breeders not only bred for aesthetics, but also in-bred the few remaining members of the breed. This in-breeding caused genetic mutations, or a permanent alteration in the genome, which give rise to many variants in the genetic data.

This method of breeding for artificial aesthetics without regard for ancestry, as

well as in-breeding the few remaining members, created a bottleneck and skew within purebred dog genetics. The rise of recent variants due to these methods results in difficulty identifying relationships between purebred dogs and other animals they are related to, such as wolves.

Using mutts for identifying variants gives rise to a new problem in excessive admixture. Admixture is defined as the presence of DNA from a genetically distant species, or interbreeding across different species. Because by definition a mutt is a cross-bred dog across two or more different breed dogs, mutts have excessive admixture which can potentially result in incorrect geographical areas identified and patterns that exist when they shouldn't.

To move past the issues of bottlenecked ancestry and admixture, many researchers use village dogs instead. Village dogs, as defined previously, are indigenous to a close given region. This is important, because present day village dogs are descendants of the area's founding population of domesticated dogs, and establish a link to the past for what dogs encompassed the original population of domesticated dogs. So long as admixture remains minimal, village dogs present a more accurate representation of the variants expected in dogs from a particular region of the world. However, while village dogs lack the bottlenecked ancestry of purebreds, they still have the potential for admixture, and they lack the information found from local breed dogs.

The history of dogs is fraught with complications — this is why the genetic history of dogs is such a contentious topic, and why many researchers observe different results and come to different conclusions based on their samples. Many papers [2] use genetic remains of ancient dogs from Neolithic dogs alongside modern day samples from either village dogs, mutts, purebred dogs, or a combination of the three. Some papers have even broached the topic of comparing village dogs with wolves, which is similar to comparing village dogs with ancient samples [20] as discussed previously.

3 Discussion of the Paper

The two data sets used in this paper were from Shannon et. al.'s [23] paper and Pilot et. al.'s [21] papers. The former data set contains variants identified in samples from 5,406 dogs. These dogs include mutts, breeds, and village dogs, as well as geographical information on where the village dogs were sampled from. Of these dogs, 2,662 are male and 2,744 are female. The samples were genotyped (i.e. compared to a reference genome to identify differences between the genomes) against a semi-custom version [23] of the Illumina CanineHD array to find 166,171 variants.

Geographic data is available from the village dogs from this dataset, and was used by the original authors in a PCA analysis to find clusters based on the samples' region of origin.

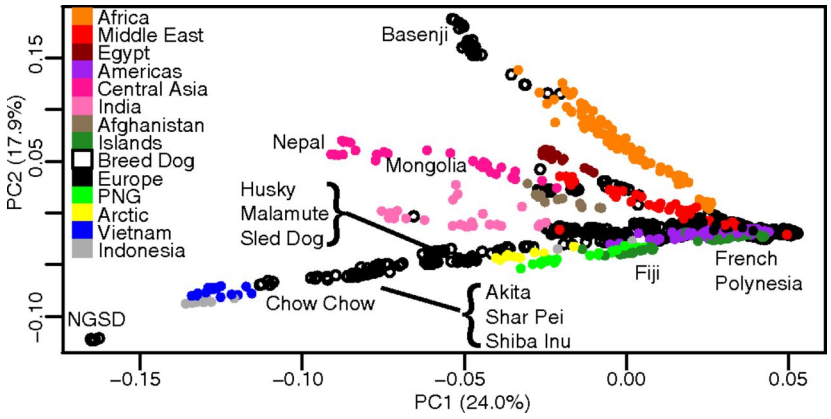


Figure 1: Geographical Clusters as Identified in [2]

The latter data set contains variants from a set containing 334 specimens sampled across Eurasia. From this dataset, 234 are village dogs across Eurasia, 96 are breed dogs, and 4 are gray wolves from Nagorno-Karabakh, an area in the central-southwest region of Azerbaijan. Two of these wolves are male and two of them are female. There are 173,662 variants in this data set. The wolves were genotyped against the Illumina CanineHD array.

Asian wolves were chosen because, despite the differences in the precise region, general consensus of dog domestication puts the origin somewhere in Eurasia. Thus, even accounting for large-scale migration, Asian wolves give the closest overall approximation to what dogs were like in the Neolithic era, before domestication became widespread. This allows us to use wolves from Azerbaijan as our sample. The combined data resulted in 5,410 unique specimens with 2,664 male and 2,746 female dogs and wolves, with 183,365 variants in total.

The discussion on the different file types that were used to store this data can be found in the Appendix.

3.1 Methods

Comparing entire genomes together is a difficult task to accomplish due to the size of a single genome, let alone comparing thousands of them. Thus, most studies perform variant analysis, which looks at SNPs - Single Nucleotide Polymorphisms. These occur when DNA is copied during cell replication. During this copying process, the nucleotides in DNA (**A**denine, **T**hymine, **C**ytosine and **G**uanine) are supposed to map to the same coordinates retaining the same order as in the original DNA strand. There is a chance during this copying process that there could be a single mistake, known as a mutation, that occurs. This mutation is known as a SNP, an example of which is documented in Figure 2 [1].

Figure 2: Notional Example of SNPs. Note that Adenine can only map to Thymine and Cytosine can only map to Guanine.

Information about these SNPs are stored in different files, the specifics of which are discussed further in the Appendix. The resulting files are processed using PLINK 1.9 [5], which is a C/C++ code that interfaces with other existing codes commonly used in research such as VCFtools and GATK. PLINK shares much of its methodology — and in some cases, its code — with GCTA (Genome-wide Complex Trait Analysis) [24], which is another free toolbox for genome analysis.

All data used in this study was processed in PLINK such that the major allele is used as the reference for the data set. This is because identifying the variants with respect to the most commonly occurring allele will give a better representation of where genetic outliers are (i.e. dogs with many SNPs). Essentially, if there is a dog with many variants, that dog will contain fewer common genes and potentially more mutations. Because the mutations occur during DNA replication, the resulting variants are much newer in origin than the existing DNA which acts as a baseline. This means that the dog not only has less in common with the average subject of the data set, but that the dog and its children will continue to deviate from the distribution of prior dogs [5]. Furthermore, because mutations arise in DNA replication and can be passed down as a trait across generations, the principle idea is that dogs with similar SNPs will cluster closer together than dogs with vastly different SNPs.

PLINK contains different statistical processing codes within its toolbox, including a method for calculating the Hamming Distances between subjects. Of particular note, it contains the native PCA method of finding clusters (described in the following subsection) as a port of the function from GTCA's PCA calculation [24].

There are actually two versions of PLINK commonly used: PLINK2 and PLINK1.9, which are somewhat different as PLINK2 is more modernized, updated code that is currently testing new parameters in its alpha [5]. However, PLINK1.9 is considered stable and has been for quite some time, and had more options to use when combining two different files into one cohesive file. To keep the calculations consistent, I used PLINK1.9 for all data combining, cleaning, and

PCA calculations.

Another software package used in this analysis is the R package *adegenet* [12], which contains, among the typical PCA methods, a variant called Discriminant Analysis of Principal Components [13]. It also includes a variant of PCA, known as spatial PCA (sPCA), which creates a graph of the distances between previously calculated Principal Components. The current implementation of the *adegenet* package does not allow for previously calculated Principal Components to be passed in as inputs. However, this can be fixed with some modifications to the source code, which is also contained in the Appendix.

3.2 Principal Component Analysis

Principal Component Analysis (PCA) [19] is a widely used statistical method that is used in multivariate data. Often, multivariate data is correlated between its data points. For example, measuring how closely related two dogs are can be affected by geographic location, who the parent and grand-parent dogs were, how many variants were found in comparing two SNPs, and so on. All of these different factors give rise to correlation between samples. This correlation between multiple variables makes accurate model building and analysis difficult.

PCA is an orthogonal transformation onto a multivariate data set such that the new variables, called principle components, are linearly uncorrelated. Consider a $n \times p$ data matrix $\mathbf{X}$, and a matrix of different weights assigned to the data, denoted as α, which is also of size $n \times p$. Each pth column in $\alpha\mathbf{X}$ is subject to the constraint that its variance is maximized while remaining uncorrelated with the rest of the columns [19].

The PCA algorithm that PLINK 1.9 uses is a port of the PCA algorithm found in GCTA [24]. PLINK1.9 calculates the genetic relationship matrix (GRM), denoted as A in the literature and equivalent to the weighting matrix α above. This relationship matrix A estimates the genetic relationships between 2 distinct individuals from analyzing their SNPs and looking for variants. This matrix is calculated using the equation:

$$A_{ij} = \frac{1}{N} \sum_{k=1}^{N} \frac{(x_{ki} - 2p_k)(x_{kj} - 2p_k)}{2p_k(1 - p_k)} \tag{1}$$

where p_k is the frequency of the reference allele for the sample, which is generally considered to be the most frequently occurring allele in a given population; and x_{ki} is the number of copies of the kth SNP of the ith individual within the sample set. N represents the number of SNPs in the data set.

From this relationship matrix $\mathbf{A}$, eigenvalues and eigenvectors are calculated for PCA.

3.3 Mahalanobis Distance

How the distances are calculated between points is significant because it establishes a baseline for relationships between points. They are just as important as summary statistics, such as averages. The Mahalanobis Distance is a form of measurement that requires linearly uncorrelated data to perform, but it can be very informative. This distance is a measurement of how far a data point is from the mean of a given distribution of data, provided that the distribution of data is spherical in nature. [6] This measurement is expressed as:

$$||x - y|| = \sqrt{det(M)^{1/n}(x - y)M^{-1}(x - y)'} \tag{2}$$

where x and y are the matrices containing different measurements, n is the number of dimensions in R^n, and M is the covariance matrix.

An important aspect about the Mahalanobis distance is that, if M is the identity matrix, I, then this equation becomes:

$$\begin{aligned} ||x - y|| &= \sqrt{det(I)^{1/n}(x - y)I^{-1}(x - y)'} \\ &= \sqrt{(x - y)(x - y)'} \\ &= \sqrt{(x^2 - y^2)} \end{aligned} \tag{3}$$

which is the equation for the *standardized* Euclidean Distance. If M is *not* the identity matrix I, then this becomes the *scaled* Euclidean distance:

$$||x - y|| = \sqrt{\sum_i \frac{(x_i - y_i)^2}{\sigma_i}} \tag{4}$$

where i represents the ith dimension in R^n [6].

It is important to note that the relationship matrix $\mathbf{A}$ is *not* diagonal — however, another key factor to consider is that, due to the transformation into PCA-space, the Euclidean distance calculated on the PCA transformed data is equivalent to the Mahalanobis distance calculated on the original data space [3].

Due to the fact that the data set is heterogeneous as defined in [6], given that it is the principal components of SNPs across several thousand examples, and that M is the relationship matrix $\mathbf{A}$ the study will proceed with the scaled Euclidean distance discussed above as equivalent to the Mahalanobis Distance.

3.4 Discriminant Analysis of Principal Components

Before discussing Discriminant Analysis of Principal Components (DAPC), a definition of Discriminant Analysis is required. Discriminant Analysis [15] (DA) is a method of sorting data. Essentially, DA summarizes differences between groups while overlooking minute differences within the groups. For example, DA would allow dogs to be grouped into classes, such as farming dogs or hunting dogs, without necessarily considering other characteristics such as their height. This allows for sorting individuals into groups to ensure that the groups are distinct from each other.

When sorting the data, it is assumed that the group priors (i.e. how the data was distributed) are unknown. This is because the assumptions made often rely on concepts such as population subdivision, which are difficult to model. In particular, modelling any subdivision of a given population would rely on existing assumptions about that population that may be untrue [13]. Given that the problem statement relies heavily on historical data and that identifying population subdivision is essentially the end goal, this means that methods without group priors need to be used.

DAPC, then, is a two-step process that combines PCA and DA. It requires a transformation of the variants via Principal Components to do an a priori analysis, and then sorting the transformed data via DA.

To perform DA, the number of prescribed clusters needs to be calculated. This is performed by the k-means [18] clustering algorithm, which is described in the next section.

Before choosing the optimal number of clusters, however, the optimal number of PCs must be retained. Too few PCs can result in a lack of information conveyed and variance accounted for in the total dataset. While this may seem to imply all of the PCs should be used, too many PCs can increase computational time and make the results hard to interpret. Thus, a happy medium must be found. While there are general "rules of thumb" to guide the selection of PCs [4], there is no set practice. Therefore, the method I used was similar to the method used in [13], which is simple to accomplish through plotting the number of PCs retained compared to the total percentage of variance accounted for.

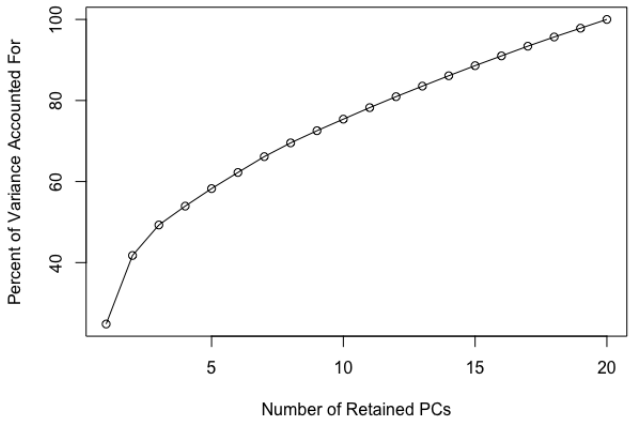


Figure 3: Plot of the number of PCs compared to the total accounted variance. Note how for high PC values, the graph takes on the shape of a square root function

Any number between 4 and 12 in the above figure would be valid.

To choose the optimal number of clusters, the Bayesian Information Criterion (BIC) [13] is used. The BIC is recommended in the paper as the optimal method for choosing the number of clusters over the Akaike Criterion (AIC) and other methods.

The BIC is calculated as:

$$BIC = n\log(||y||^2 + ||y - \hat{y}||^2)) + k\log(n) \tag{5}$$

where $||y||^2$ is the vector of predictions from the k-means algorithm, $||y - \hat{y}^2||$ is the vector of residuals from the k-means algorithm, k is the number of clusters chosen, and n is the sample size. The optimal value of k from the BIC criterion is located when the plot of k versus the BIC values creates an elbow shape, as shown below as a notional example in the left graph. However, note in Figure 4, which is an example of what the BIC plots looked like for my data set - the proposed elbow shape is much less steep, which makes picking an accurate number of clusters difficult. Within this range, anything from $k = 5$ to $k = 25$ would be a valid number of clusters.

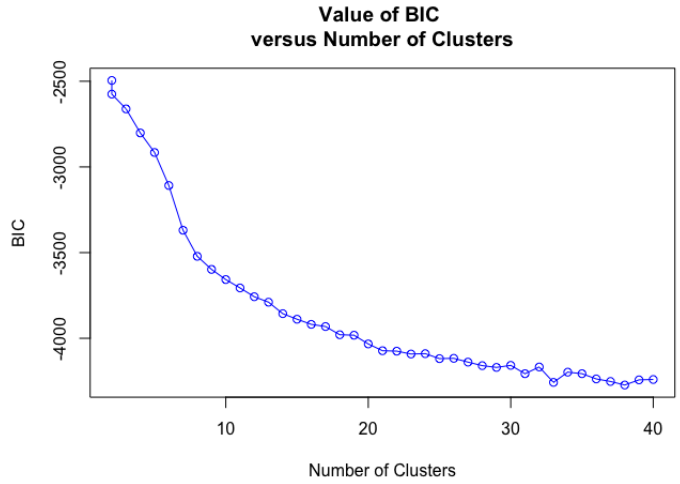


Figure 4: Plot of the BIC for the dataset of wolves projected onto all village dogs. 6 PCs were retained.

To prevent picking too large of a k value, another criterion is used that was not included in the original DAPC paper. This criterion is known as the Average Silhouette Method [14]. The Silhouette value is a ratio of how similar an object inside its own cluster is to how similar it is to objects outside its own cluster; the closer to +1 this ratio is, the more the clusters chosen are appropriate because the divide the groups into clusters accordingly. The measure of similarity is calculated using the Euclidean Distance.

Denoting $f(i)$ as the smallest average distance between points within a cluster, and $g(i)$ as the smallest average distance between a point within the cluster i and any point in any cluster that isn't i, then the silhouette coefficient is calculated using the following:

$$s(i) = \frac{g(i) - f(i)}{\max(f(i), g(i))} \tag{6}$$

Then, choosing the optimal method of clusters involves both analyzing the BIC plot and the Average Silhouette plot to see where the plot bends for the former, and where the largest value is for the latter.

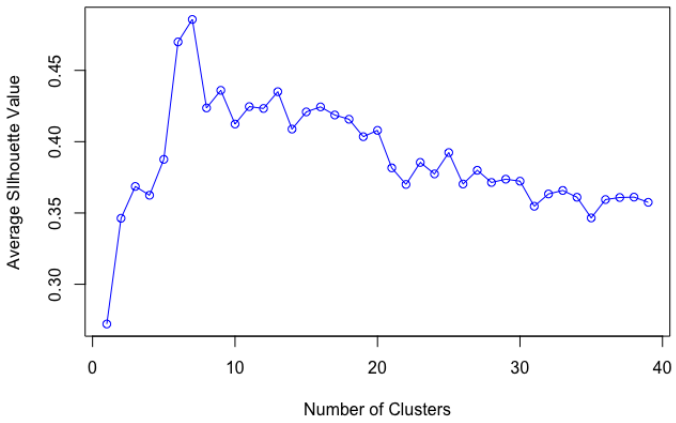


Figure 5: Example of the Silhouette Coefficient calculated for n = 2 up to n = 40 clusters.

As shown in Figure 4, the BIC plot denotes a range of values for k, and as shown in Figure 5, the Average Silhouette plot demonstrates a clear peak, or optimal number of clusters. This cluster value of k found from the Average Silhouette plot is then used as the optimal k value for both DAPC and the hierarchical clustering methods.

Then, once the optimal value of k is chosen by the above two methods, DA is performed on the principal components obtained from PLINK using the groups denoted in k-means as the appropriate model. Once these values are obtained, the results are plotted to show what groups they belong to.

4 Clustering Methods

To review, we have a methodology for finding the distance between a sample of variants and its transformed space, as well as a method of combining principal component analysis and discriminant analysis to form groups. One final question that needs to be addressed is how to group the principal component projections so that they aren't reliant on manual color coding. Furthermore, how can they be grouped so that they account for all the principal components?

Because the PCA plots can take on any shape, a clustering method chosen must be robust enough that it does not simply assume all clusters are spherical in nature. Furthermore, for it to be an optimal clustering method, it must be able to handle multidimensional inputs (i.e. to create groupings in R^d where $d \geq 2$.

All methods of cluster analysis can be divided into two different methods:

- Partitioning Clustering, which splits existing data into k groups that are distinct from each other, where k is often chosen by the user based on some summary statistics for the most optimal k
- Hiearchical Clustering, which identifies groups of similar observations. These groups are set up such that they all relate back to the entire set of data.

The k-means clustering algorithm [18] works to group together n data points into k clusters, where k is chosen by the number of clusters desired from the dataset, with the limitation that k cannot be larger than the number of data points x. It is an iterative algorithm that first assigns the k-means (the average or center value of a cluster k, denoted as m_k) to a random point within the dataset. The next step in the algorithm is called the assignment step, where each data point is assigned to a guess for a cluster k by minimizing the distance between m_k and x, i.e.

$$k_{guess} = argmin_k\{dist(m_k, x_n)\} \tag{7}$$

Once this is performed, a flag value f_n is denoted as 1 if k_{guess} is the closest mean to the datapoint x_n, and 0 otherwise. Each cluster k has a total flag value $F_{n,k} = \sum_n f_{n,k}$.

Then, the update step occurs, where the k-means values m_k are updated to reflect the sample averages: that is,

$$m_k = \frac{\sum_n f_n x_n}{F_n} \tag{8}$$

The above process is repeated until the value of m_k does not change - that is, all data points are assigned to their closest clusters.

As discussed in the prior section, DAPC uses k-means to group together principal components, which is a method of partitioning clustering. This is effective for identifying distinct groups within a given data set. However, partitioning clustering methods do not give any insight into how these clusters - the *parts* of the data set - relate back to the whole.

Hierarchical clustering, meanwhile, allows for the relation of a part to a whole. These are expressed in terms of dendrograms, which are tree-like graphs that relate the different samples to different smaller clusters that relate to the whole of the dataset. Figure 6 demonstrates the differences between partition clustering using k-means and hierarchical clustering on the notional *iris* dataset included previously in R.

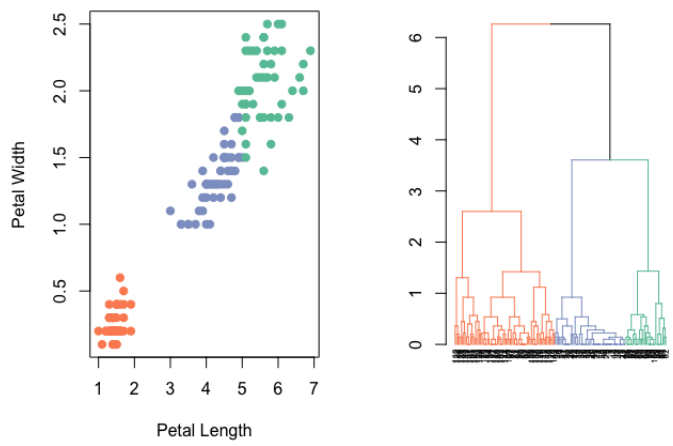


Figure 6: Visual differences between partitioning clustering (left) and hierarchical clustering (right)

To interpret the dendrogram, look at the height of any two objects. This height is a direct reference to how close, distance-wise, any two points are to each other. The smaller the height is, the closer points are to each other. Each branching point indicates a segment that is divided into a separate cluster, too — removing the top branch results in two clusters, removing the next branch results in three clusters and so on until each point is considered its own distinct cluster. An important note about dendrograms is that they cannot be used to determine the most optimal number of clusters, like how the optimal number of clusters for k-means was determined. It is simply a method of relating objects to each other via their distances, and categorizing how many objects are similar to each other.

It is important to remember that, while visually similar, dendrograms and phylograms are different graphs that demonstrate different ideas. A phylogram's branches demonstrate inferred evolutionary change between groups of points — the longer the branch, the more change which occurred. A dendrogram, meanwhile, is more general. In the case of this study, a dendrogram visualizes how close or how far away a datapoint is from its neighbors based on the PCs of its variants.

An issue that arises across some methods of partitioning clustering is the native assumption that the data — and thereby the clusters — are spherical in nature. [18] If the data fall into a straight line, or overlap each other like a sine wave (as shown in the below notional image), k-means will overcompensate and declare there to be more clusters than actually exist. Thus, one final proposed clustering

method is DBSCAN [7].

DBSCAN works on the assumption that clusters are groups of dense points, regardless of the shape of those groups. For each point a within a cluster, there has to be a distance greater than 0, denoted as ϵ_0, that contains a set number of neighbor points, denoted as *MinPts*: that is,

$$Neighborhood_{\epsilon_0}(a) = \{b_1, b_2, ...b_{MinPts} \in D | dist(a, b) < \epsilon_0\} \tag{9}$$

This works well for points within a dense region, but for points on the border of a cluster, there may not be enough points surrounding it to satisfy the ϵ_0 neighborhood condition. If the ϵ_0 distance condition was reduced to compensate for identifying border points, the border points could be identified alongside noise, which are observations that do not belong to any cluster.

There are two ways a border point can be defined with respect to the desired density achieved from the ϵ_0 factor specified. They can either be *density reachable* (a point a can be reached from $Neighborhood_{\epsilon_0}$ of point b that is more than one neighborhood away, but b cannot be reached from the point a), or they can be *density connected* (there is a point o such that the border points a and b can be reached from multiple neighborhoods from o). Figure 7 demonstrates this visually and was taken from [7].

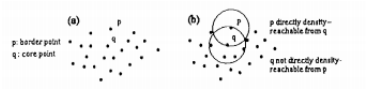


Figure 7: Illustrated example of core points (left) and border points (right).

There are some heuristic methods to choosing the parameters ϵ_0 and *MinPts*. The authors of the method recommend using $2 * dim$, where dim represents the dimension of the dataset, for the minimum number of points [22]. Thus, for a 2-dimensional plot, the recommended minimum number of points is 4. In the examples with the wolf data following, this has the benefit of ensuring that the wolves will all cluster together, as each wolf will only have 3 nearest neighbors, ensuring that the 4th nearest neighbor is a dog. This number is also used for the *MinPts* value for choosing the epsilon distance. Some articles [8] recommend using a *MinPts* value of $2 * dim - 1$. However, because the dataset used has 4 wolves and a variety of dogs, using a *MinPts* value of 3 as recommended would run the risk of only grouping wolves with wolves for recommended radii, whereas a value of *MinPts* equal to 4 would ensure one of the nearest neighbors to any wolf point would be a dog. This sort of analysis of data to choose the *MinPts* value is recommended in [22], as well as leaving one parameter open to variation. Because of this, values of *MinPts* equal to 3 and *MinPts* equal to 4 are used when choosing the epsilon value.

Determining the epsilon distance is somewhat difficult, as discussed above. A method similar to the BIC criterion discussed previously is employed, where the points are sorted by the distance they are from each other, and an elbow area is defined to be the optimal epsilon distance. The graph below demonstrates this case for the mutts and wolves dataset — the optimal epsilon distance, in this case, is $\epsilon_0 = 0.011$ [7].

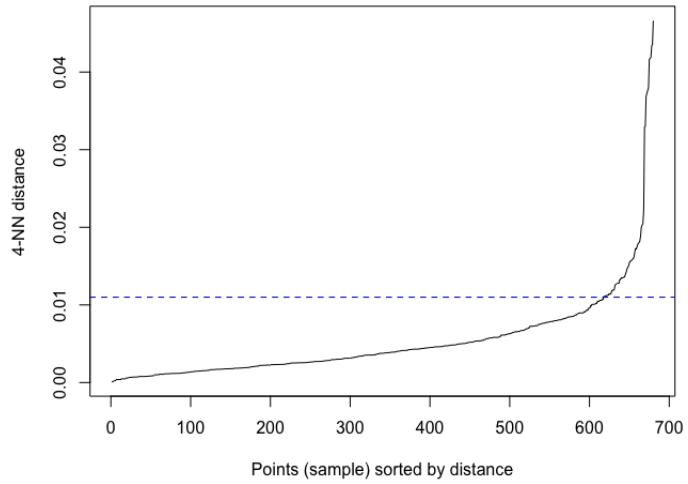


Figure 8: Epsilon Distance plot for the set of mutts and wolves. Note that a case can also be made for slightly larger distances.

5 Results

When constructing this experiment, the data was divided into four different sets:

- The entire set of dogs
- The set containing no village dogs (including breeds and mutts)
- The set containing only village dogs
- The set containing only mutts

Onto each of these subsets of the original data set, four selected wolves were added to the data set, and PCA was calculated on the subset with the additional four selected wolves. This resulted in a PCA-space which is then used for the

previously discussed clustering methods, the results of which are discussed in the following paragraphs.

The following graph shows the wolves chosen for this study based on the plot of their first two Principal Components. The selection was based on obtaining a sample from each of the outliers in the 3-pronged shape, as well as a sample relatively close to the origin where there are clusters of similar wolves.

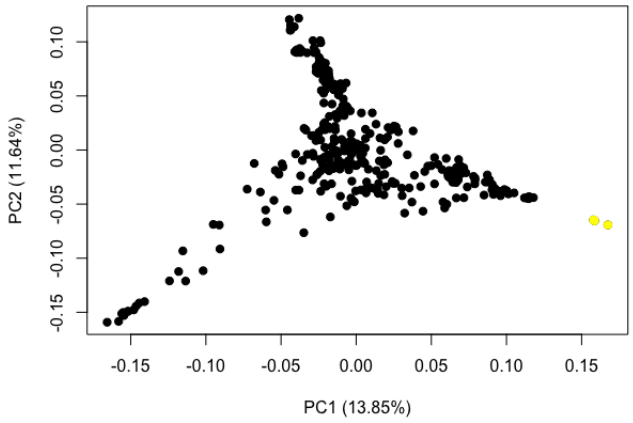


Figure 9: PCA Plot of Azerbaijan Wolves. Yellow labels indicate chosen samples.

These wolves were identified to be from Azerbaijan as described in the supplementary material for [21].

PCA was performed on each of these subsetsets using PLINK and then imported into R. The PCA plots are done using the raw PCA results obtained from PLINK, and any distance measure calculated between points uses the Mahalanobis Distance, which would require the PCA points to be scaled. Note that scaling the data does not change the actual shape of the data, it only changes the axes because all points are divided by a scalar relative to the original vector - in this case, all points in a vector $\mathbf{x}$ are divided by the standard deviation of that vector, σ_x, so the actual shape doesn't change. This data was then imported into R and plotted below. In all proceeding plots, yellow circles indicate where the Azerbaijan Wolf samples are.

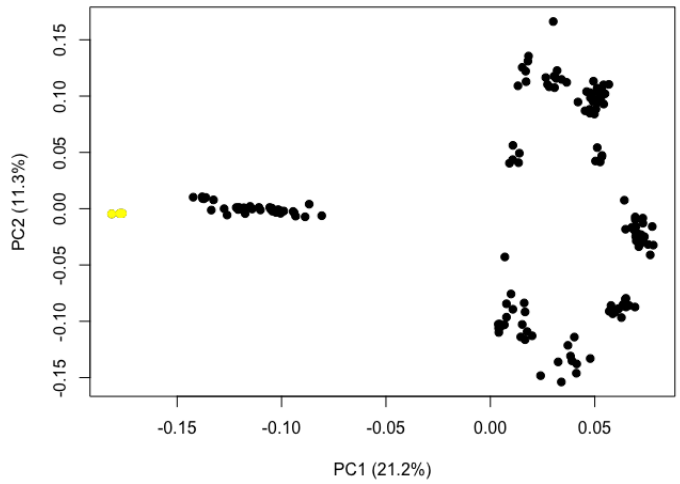


Figure 10: PC1 and PC2 Plot of Only Mutts and Wolves

The above plot demonstrates the PCA results from plotting on a space containing only mutts and the 4 selected wolf samples. From the plot, it is clear that variant data divides the mutts and wolves into 2 groups: a group that contains wolves, and mutts with very similar variants to the wolves, which corresponds to the horizontal line; and a ring in the positive PC1 half, containing only mutts. This variation in grouping indicates that there's a clear separation of groups based upon the variants across mutts.

The next graph shows the space containing all 549 village dogs and the 4 selected Asian Wolf samples.

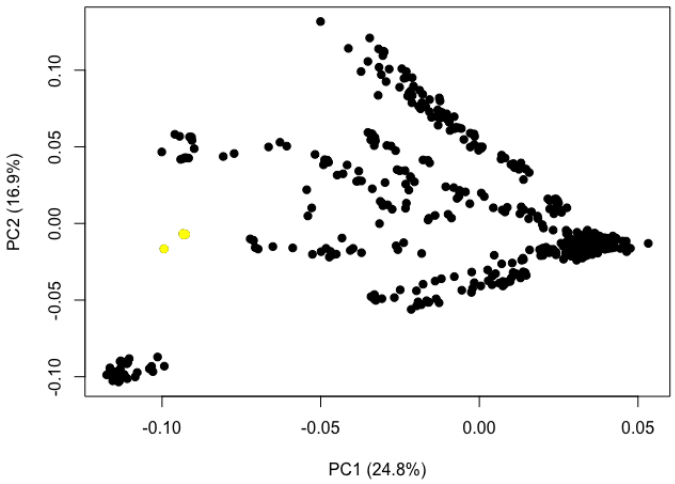
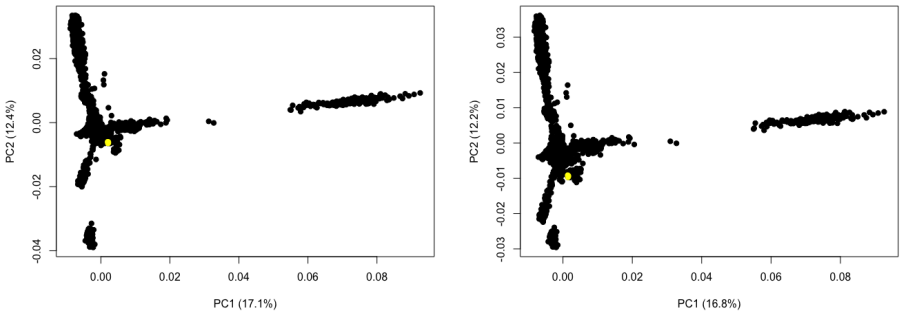


Figure 11: PC1 and PC2 Plot of Village Dogs and the Azerbaijan Wolves

The next graph is an example of PCA results obtained from plotting on a space that contains no village dogs, and the entire data set, and the 4 selected Asian wolf samples. Note that the entire data set retains its overall shape, but shifts the graph down slightly and contains more points in the centroid object on the right.



(a) No Village Dogs

(b) All Dogs

From here, the k-means results are shown, with the following table indicating the most optimal number of clusters as chosen by the Silhouette Coefficient. Note that the random starting point chosen by R during the k-means process is chosen manually to be the same number used in the DAPC results.

Data Set	k
All Dogs	4
Village Dogs	3
No Village Dogs	4
Mutts	3

First is the k-means clustering results on the data set containing village dogs and the Azerbaijan Wolves as shown in the following figure. Here, the cluster results appear to be very coarse, selecting large clusters that appear to follow an ellipsoid shape overall. In addition, they fail to identify the general 3- or 4-pronged shape clear from the original data set.

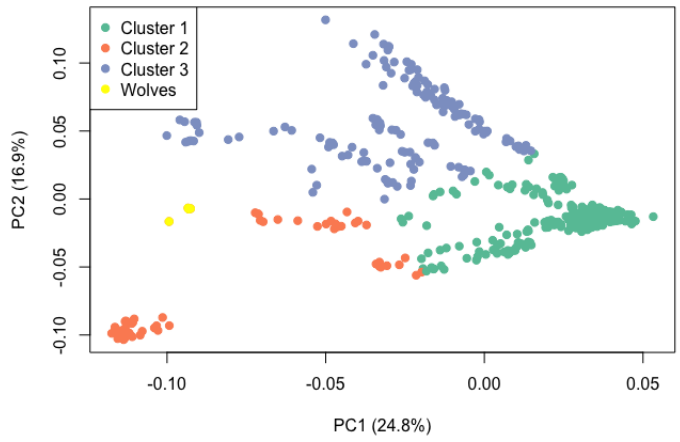


Figure 13: k-means clustering results for Village Dogs and the Azerbaijan Wolves

The next figure demonstrates the k-means clustering results for the set of all Mutts and the Azerbaijan Wolves. Note that in this plot, it appears that the Silhouette Coefficient returned a recommended cluster number of 3, which splits the doughnut shape into an upper u-shaped cluster and a lower u-shaped cluster.

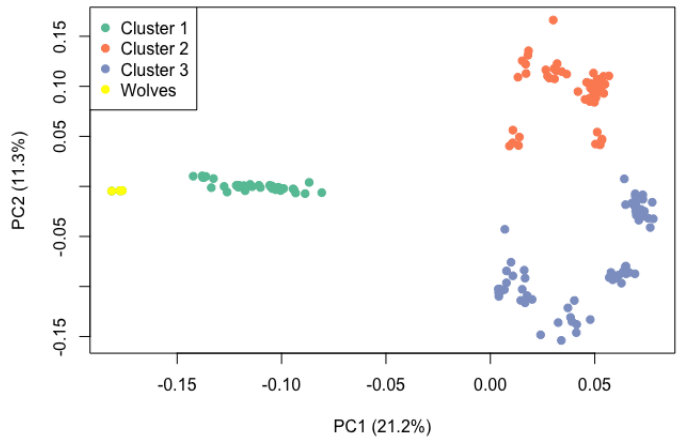
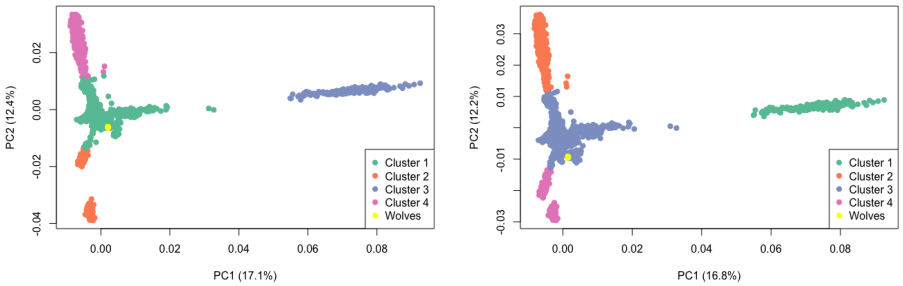


Figure 14: k-means clustering results for Mutts and the Azerbaijan Wolves

From this, because the datasets of all dogs and the set of all dogs without village dogs are so similar, their figures are shown side by side below. Note that the dataset without village dogs (a) has a much larger top cluster, and the dataset containing all dogs (b) has more data points concentrated in the center orange cluster.



(a) No Village Dogs

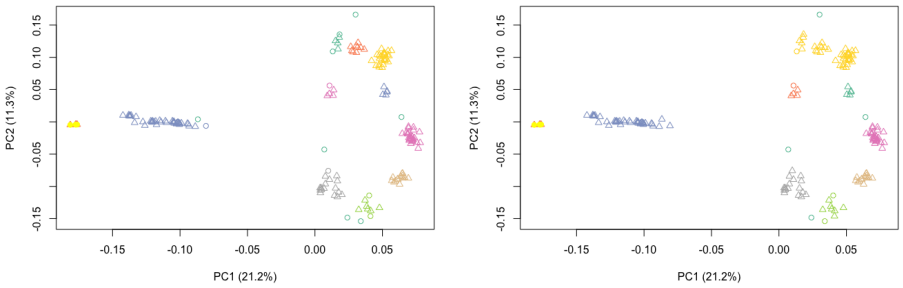
(b) All Dogs

After this, the results from the DBSCAN method are shown and discussed. The following is a summary table discussing the parameters chosen based on the discussion in the DBSCAN section. Note that, for the datasets without the village dogs, as the *MinPts* parameter increased, the ϵ_0 value also increased. However, the values of ϵ_0 chosen still have some level of subjectivity due to the need to visually evaluate the graph of their *MinPts*-nearest neighbors. In

addition, the points corresponding to triangles correspond to points belonging to a cluster, and the circles correspond to outliers.

Data Set	MinPts	ϵ_0	MinPts	ϵ_0
All Dogs	3	0.0008	4	0.0008
Village Dogs	3	0.007	4	0.007
No Village Dogs	3	0.0008	4	0.0009
Mutts	3	0.01	4	0.014

The first plot discussed is the DBSCAN results for the Mutts and Azerbaijan Wolves.

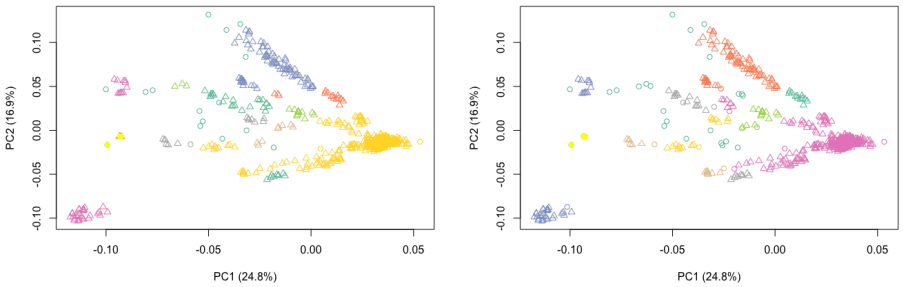


(a) Mutts (MinPts=3)

(b) Mutts (MinPts=4)

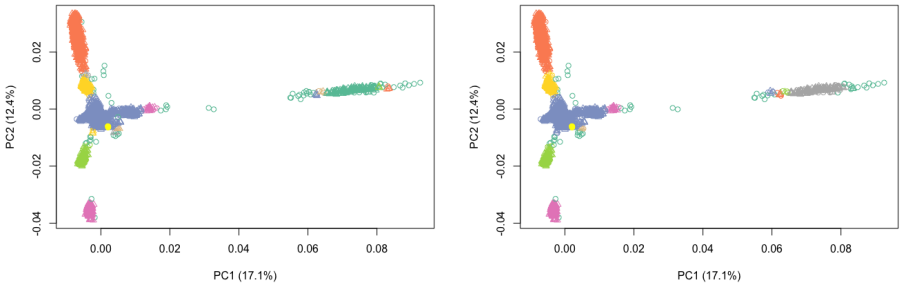
From the results, it appears that when MinPts = 3, there are 3 clusters at the apex of the circle that become grouped together when MinPts = 4. This is because there appears to be a visually distinct gap between the groups which is not detected when the minimum number of points in a cluster is 4 — the gaps are ignored in order to create a cluster that satisfies the condition.

Next is the plot the DBSCAN clustering results of only village dogs. From the results, it appears that, while some of the branches appear to fall into many small clusters, the large branch at the top as well as the "centroid" consistently fall into their own clusters. Comparing this to the geographical clustering demonstrates that, while DBSCAN appears to group together the African and Egyptian village dogs well, it separates many of the points in Central Asia into many small clusters.

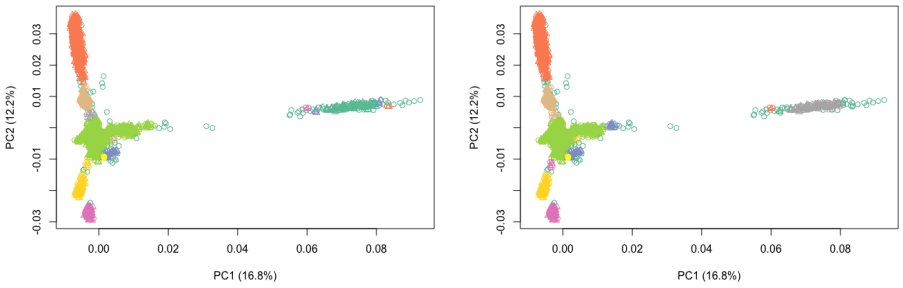


(a) Village Dogs (MinPts=3) (b) Village Dogs (MinPts=4)

Finally, we can compare the clusters obtained between the set of no village dogs and the set of all available dogs.



(a) No Village Dogs (MinPts=3) (b) No Village Dogs (MinPts=4)



(a) All Dogs (MinPts=3) (b) All Dogs (MinPts=4)

The clusters identified across both datasets are very similar, though it appears

Data Set	Number of Clusters (Silhouette)	BIC Range	Number of PCs
All Dogs	13	(10, 20)	11
Village Dogs	15	(6, 16)	10
No Village Dogs	12	(12, 20)	11
Mutts	17	(8, 20)	9

that there's a very small cluster nestled close to the wolves (tan in the No Village Dogs set) that grows in size in the set containing all dogs.

The next results presented will be the DAPC clustering results. All of the presented clustering results were run using the same seed for reproducibility as mentioned above when discussing k-means which is discussed in the Appendix section. The native plotting function for DAPC does not include axes labels because it's intended as a diagnostic tool to identify trends in the data based on where the clusters lay [13]. However, all plots show the projection of the first two linear discriminants, where the x axis represents the first linear discriminant and the y axis represents the second linear discriminant.

A table of the summary statistics used during the DAPC diagnostic process is presented below. Note that, generally, the number of clusters tends to be on the higher end for smaller datasets and on the lower end for the larger data sets. The number of PCs was chosen so that they explained at least 70% of the variance.

The first plot represents the data set containing only mutts. The group identified in this plot containing the wolves is group 14, which only contains the values associated with the wolves. Close to it is group 5, which contains a very small group of about eight mutts.

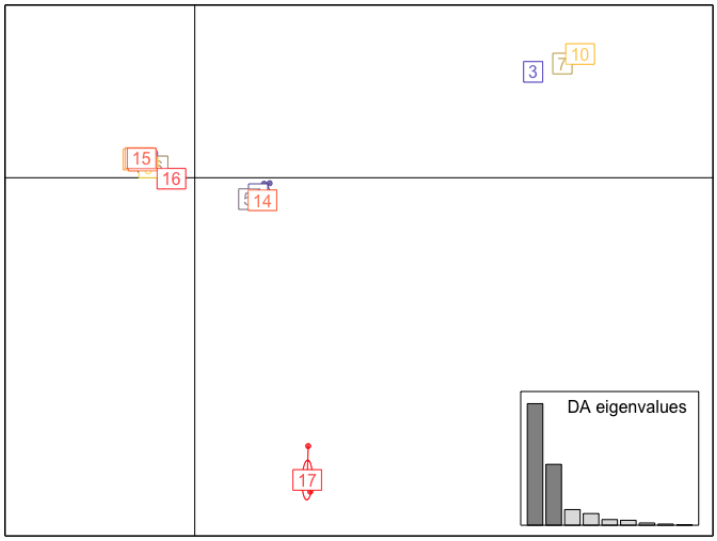
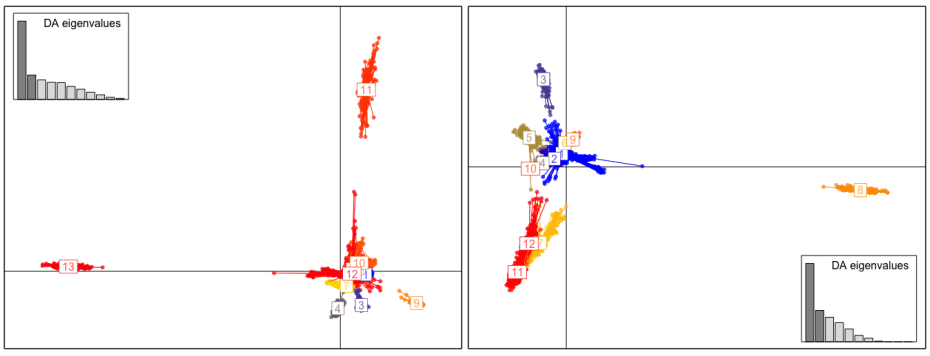


Figure 20: Only Mutts

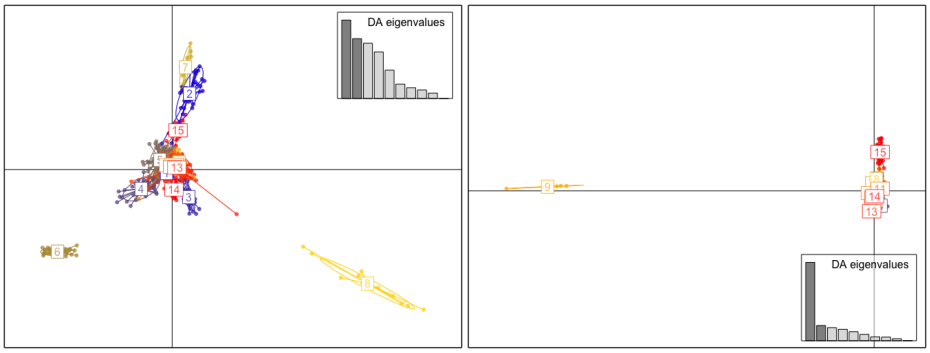
The next plot shows the DAPC clustering results on the data set containing all dogs and no village dogs. Both of these clusters are large, with the cluster containing the wolves in the all dogs data set, cluster 2, containing 496 members, some of which are village dogs from Afganistan and Basenji breed dogs. The cluster containing the wolves in the no village dogs data set, cluster 1, contains over 2,000 members, with many different breeds including Gordon Setters and Doberman Pinschers.



(a) All Dogs (b) No Village Dogs

Finally, these two plots correspond to the DAPC clustering results obtained from the data set containing only village dogs. This demonstrates the importance of ensuring that the seed is documented for reproducability, as the plot on the left was created with the same seed as the rest of the DAPC plots, and the plot on

the right was generated with a different seed. On the left, the cluster containing the wolves, cluster 8, also contains eight village dogs from Alaska, whereas the cluster containing the wolves on the right, cluster 9, exclusively consists of the wolves.



(a) Village Dogs (same seed) (b) Village Dogs (different seed)

From here, the results using dendrograms are discussed. Because the dataset containing mutts and wolves has less than 200 samples, each datapoint is viewed as its own in the dendrogram. Averages are taken for the sets of all dogs and the sets of no village dogs so that each branch represents the average of all its members (i.e. the Border Collie branch would represent the mean value of all Border Collie samples.)

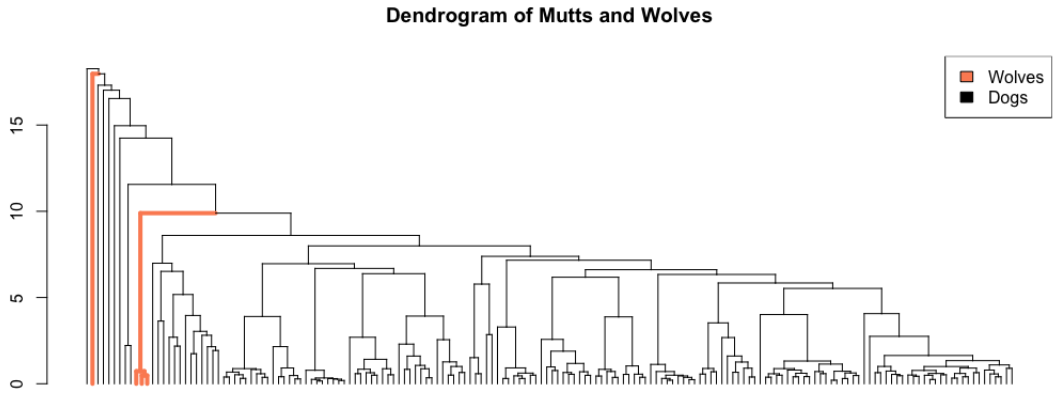


Figure 23: Dendrogram of Mutts and Wolves

The above dendrogram uses the Mahalanobis distance (standardized Euclidean Distance), and demonstrates that, while the 4 wolf samples colored in pink are quite far away from the other samples, they still are close enough in distance to some mutts that those mutts may be interesting to interpret. However, that cluster of mutts is assuredly an outlier due to its relative distance from the other, more densely packed clusters.

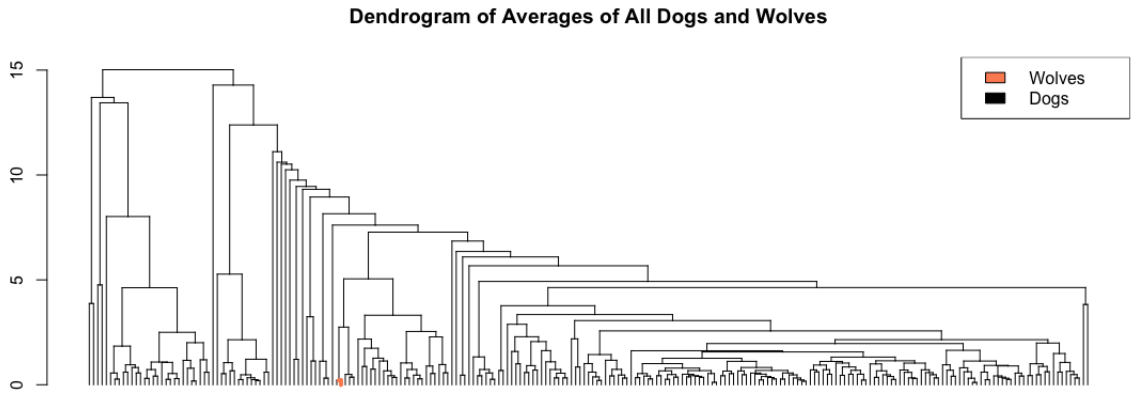


Figure 24: Dendrogram of the Averages of All Dogs and Wolves

Note in the above dendrogram that the clusters appear to have a cascading pattern of some relatively small and some relatively large clusters. This actually is not reflected when the village dogs are removed, where there is a very strange branching pattern.

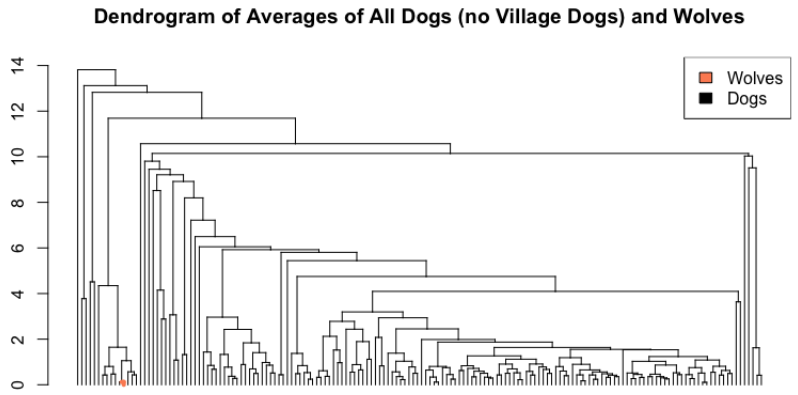


Figure 25: Dendrogram of the Averages of All Dogs without Village Dogs and Wolves

Here there seems to be a small group of outliers to the far right, and the cascading pattern of clusters shown in the previous plots. The wolves fall into a very small cluster toward the center, which is a bit odd.

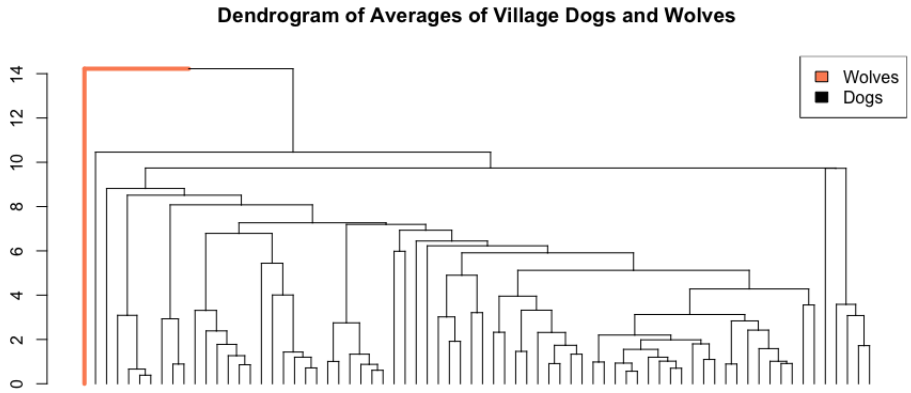


Figure 26: Dendrogram of the Averages of All Village Dogs and Wolves

Finally, the group of village dogs only is much clearer to interpret. The wolves

belong to a very small cluster in the center, and the cascading pattern of clusters is still visible.

6 Conclusion

This paper was a summary of different clustering and distance measure algorithms in an attempt to better describe the genetic distances between two similar species. Rather than taking a visual approach using the first two PCs as is the norm in current literature [2] [10], this paper is a mixture of different clustering methods such as DAPC, hierarchical clustering, partitioning clustering, and DBScan to determine if there are any additional insights that cannot be gleaned from merely eyeballing the data.

From the k-means data gathered, it's clear that the clusters detected are indeed visually distinct groups, but these groups often contain many different dogs. While it does define groups that are distinct enough from each other to be partitioned in individual clusters, the recommended number of clusters result in coarse clusters that also fail to take into account the shape of the PCA plots (see the previous k-means plots on the mutt dataset) in determining clusters. In addition, this method still relies on only using the first two PCs, which means that, often, less than 40 percent of all variance is accounted for in these plots.

Using the DBSCAN method of clustering, more unique clusters are identified in most cases, the sole exception being the case in clustering all dogs with an ϵ_0 distance of 0.001 and a *MinPts* of 3. Generally, this method seems to be better at identifying points that are very close to each other and would theoretically be similar to each other in terms of their genetic variants, which could be helpful for identifying different specimens similar to each other that geographical clustering would fail to identify, especially in the large sample cases like the set of all specimens. However, like k-means, this is again limited to a 2-dimensional plot for interpretation which means that in the majority of cases less than 40% of the variance is accounted for.

Finally, using the DAPC method of clustering, more unique groups can be identified and easily visualized in distinct groups. However, one major drawback of the DAPC plots in their current implementation is that they are somewhat difficult to interpret, given their lack of labels on the axes — thus, evaluating how close or far a cluster is from its neighbors can only be done very coarsely. In addition to this drawback, running DAPC results in different results each time, due to the random nature of both k-means and how R finds the linear discriminant functions. This means that situations like the one discussed, where a cluster could be found either near its neighbors or very far away, could occur and lead to irreproducible results if a seed is not specified beforehand. A future method of resolving this issue could be to run DAPC multiple times and identify how far or how close a certain cluster is to its neighbors. This metric of how

close or how far a cluster is, however, would have to be calculated with respect to the dataset itself, and is hard to implement without a proper scale set on the DAPC plots. Furthermore, by using DAPC, more PCs can be used in the plots, but now that the axes are the projections of the linear discriminant functions, the problem of 2-dimensional projections is simply shifted onto the linear discriminants rather than the PCs.

Both the issue of the 2-dimensional projection and the lack of PCs is addressed by dendrogram plots, which are able to consider the entire PCA space when identifying points either close to or far away from any point of interest. However, visualization is a key part of the dendrograms - without a guide, they can be confusing to interpret, especially for large data sets. Even in small datasets, such as the notional iris dataset or the set of wolves and mutts, the density of branches makes interpretation difficult. Taking the averages can help account for this, but the averages can potentially be skewed by outliers within the sampled groups.

Note that, in all of the clustering methods, the identified clusters do not match very well with the clusters identified from looking at the geographic regions, as shown in the original text [2]. In addition, this analysis of the data also reveals something interesting about the sampled wolves - in particular, that even though automated clustering algorithms tend to view them as outliers, that's not the case for all of the clustering methods, even when the entire PC space is considered. This does make sense due to the fact that wolves are close relatives to the modern domesticated dog.

From all of the above, I would argue that some clustering methods, such as k-means, produce algorithmically optimal clusters that are simply too coarse. While an algorithm may not surpass the clusters identified via geographical region, additional analysis of PCA data with various other algorithms can reveal similarities that were not previously considered from simply analyzing the geographic clusters. This is especially evident with hierarchical clustering using the Mahalanobis distance, where the dendrograms demonstrate similar points using the entire PCA space to explain similar points given an uncorrelated space which explains 100% of the variance in the original data set.

Furthermore, I would argue for, alongside the current literature standard of using geographic clusters, using dendrograms on the entire PCA space to further identify connections between samples of interest and the rest of the data. As stated previously, this would allow for a comparison between clusters identified based on sampled region of origin and clusters identified on the entire data space. While this study alone does not help answer exactly where dogs were domesticated, this study does demonstrate that algorithmic clustering methods may not be optimal for identifying where exactly dogs were first domesticated.

7 Acknowledgements

There are so many people I would like to acknowledge for their support and guidance in the past year of research; in particular, my advisors, Prof. James Degnan, Prof. Jens Lorenz, and Prof. Chris Holden, my parents, my partner, Zachary Mead, and my friends.

8 Appendix

8.1 Data Structure

The data set for the 5,406 dog sampled used in this study is stored across 3 different file types:

- bim - extended variant call file. This can be read as a text file and contains the following information in tabular format:
    1. Chromosome code
    2. Variant ID
    3. Position in centimorgans (unit of measurement for genetic linkage, defined as the distance between chromosome positions where the expected average number of intervening chromosomal crossovers in a single generation is 0.01.
    4. Base-pair coordinate
    5. ALT (alternative) allele code
    6. REF (reference) allele code
- bed - binary biallelic genotype table. This contains the genotype data stored in binary format to reduce the file size.
- fam - Information about the samples. This is stored as a text file containing the following categorical data.
    1. Family ID
    2. Individual ID
    3. Individual ID of father
    4. Individual ID of mother
    5. Sex code
    6. Phenotype value

This data for the 334 wolves is stored in ped/map format. This format is similar to the bed/bim/fam trio of files described above, and stores the following information:

- .ped - Pedigree/Genotype table. Contains informaiton on the genotypes.
  1. Family ID
  2. Individual ID
  3. Individual ID of father
  4. Individual ID of mother
  5. Sex code
  6. Affected (Numerical code that describes if a variant exists within the genotype)
  7. Genotypes
- .map
  1. Chromosome code - which chromosome the variant was found in.
  2. Variant ID
  3. Genetic distance in centimorgans
  4. Physical position in the chromosome

8.2 Code Notes

The seed set at the beginning of the DAPC code, seed number 19274927 in R, is only used for the DAPC plots for reproducability's sake. The same seed was used for the k-means code for the same reason. The DAPC code is a stripped down version of the source code from the DAPC package [13]. The Other Code was used for the other clustering methods, and is not meant to be run all at once. It is meant to be run in smaller pieces, and those outputs and any changes are meant as inputs for the next part of the code.

8.2.1 DAPC

```
library(MASS)
library(adegenet)
library(cluster)
library(factoextra)
set.seed(19274927)

.compute.wss <- function(x, f) {
  x.group.mean <- apply(x, 2, tapply, f, mean)
```

```
  sum((x - x.group.mean[as.character(f),])^2)
}

#in order to pass in the requisite PCs
sb = read.table('/filepath/to/file.eigenvec')
sb2 = read.table('/filepath/to/file.eigenval')
sbNames = read.csv('ShannonBoyko_All_NoPheno.csv',header=F)

PCLevels <- 100*cumsum(sb2[1])/sum(sb2[1])
x <- 1:20
totalPCs = sb[, -c(1:2)]
plot(x,t(PCLevels),xlab="Number of Retained PCs",ylab="Percent of Variance Accou
i <- 10
PCs <- totalPCs[,1:i]
N<-nrow(PCs)
minNumClust = 2
maxNumClust = 40
nbClust = minNumClust:maxNumClust
WSS <- numeric(0)
avg.silhouette <- numeric(0)
for (j in 1:length(nbClust)){
  temp <- kmeans(PCs,centers=nbClust[j],iter.max=1e5,nstart=10)
  WSS[j] <- .compute.wss(PCs,temp$cluster)
  avg.silhouette[j] <- mean(silhouette(temp$cluster,dist(PCs))[,3])
}
#find AIC
WSS.ori <- sum(apply(PCs,2,function(v) sum((v-mean(v))^2)))
k <- nbClust
myStat <- N*log(c(WSS.ori,WSS)/N) + 2*c(1,nbClust) #need to define WSS and N
myLab <- "AIC"
myTitle <-"Value of AIC \nversus Number of Clusters"

#find BIC
WSS.ori <- sum(apply(PCs,2,function(v) sum((v-mean(v))^2)))
k <- nbClust
myStat <- N*log(c(WSS.ori,WSS)/N) + log(N)*c(1,nbClust) #need to define WSS and
myLab <- "BIC"
myTitle <-"Value of BIC \nversus Number of Clusters"

#plot AIC values
plot(c(2,nbClust),myStat,xlab="Number of Clusters",ylab=myLab,main=myTitle,type=
abline(h=0,lty=2,col="red")

#calculate the silhouette and plot it

#mine
```

```
plot(nbClust-1,avg.silhouette,type="o",col="blue",xlab="(Number of Clusters)",yl
#package to double check
fviz_nbclust(PCs, kmeans, method = "silhouette",k.max=maxNumClust)

#use the plots to inform num of clusters
k <- 15

#grp <- find.clusters(totalPCsubset, max.n.clust=40) #
grp <- kmeans(PCs,k)
ldaX <- lda(PCs, grp$cluster, tol=1e-30)
barplot(ldaX$svd^2, xlab="Linear Discriminants", ylab="F-statistic", main="Discr

#choose the number of linear discriminants to retain
f <- 10
predX <- predict(ldaX, dimen=f)

#create a DAPC object
grp <- as.factor(grp$cluster)
res <- list()
res$n.pca <- i
res$n.da <- f
res$tab <- PCs
res$grp <- grp
res$var <- PCLevels
res$eig <- ldaX$svd^2
res$loadings <- ldaX$scaling[, 1:f, drop=FALSE]
res$means <- ldaX$means
res$ind.coord <-predX$x
res$grp.coord <- apply(res$ind.coord, 2, tapply, grp, mean)
res$prior <- ldaX$prior
res$posterior <- predX$posterior
res$assign <- predX$class
res$call <- match.call()

class(res) <- "dapc"
scatter(res)
```

8.2.2 Other Code

```
library(adegenet)
library(MASS)
library(ggplot2)
library(factoextra)
library(FactoMineR)
library(fpc)
library(dbscan)
```

```
library(dendextend)
library(RColorBrewer)
library(stringr)
palette(brewer.pal(n=8,name="Set2"))
set.seed(19274927)
#import datasets
sb = read.table('/filepath/to/file.eigenvec')
sb2 = read.table('/filepath/to/file.eigenval')
sbNames = read.csv('ShannonBoyko_All_NoPheno.csv',header=F)
sbNames2 = read.csv('ShannonBoyko_VillageDogNames.csv',header=F)

PC1 = sb$V3
PC2 = sb$V4

#check to see how much of the variance the PCs correspond to
PCLevels <- 100*cumsum(sb2[1])/sum(sb2[1])
(sb2[1])/sum(sb2[1])
#plot code for generic r plot. PC% come from line 11.
plot(PC1,PC2,pch=19,xlab="PC1 (21.2%)",ylab="PC2 (11.3%)")
points(PC1[1:4],PC2[1:4],pch=19,col="yellow")
PCi <- data.frame(PC1,PC2)
ggplot(PCi,aes(x=PC1,y=PC2)) + geom_point(size=3,alpha=0.5)

#kmeans code for grouping the points.
fviz_nbclust(PCi, kmeans, method = "silhouette",k.max=20)

k = 3 #number of clusters
clusterInfo <- kmeans(PCi,k)
clusterCol <- clusterInfo$cluster
plot(PC1,PC2,col=clusterCol,pch=19,xlab="PC1 (24.8%)",ylab="PC2 (16.9%)")
points(PC1[1:4],PC2[1:4],pch=19,col="yellow")
legend("topleft",legend=c("Cluster 1","Cluster 2","Cluster 3","Wolves"),col=c(1,

k = 4 #number of clusters
clusterInfo <- kmeans(PCi,k)
clusterCol <- clusterInfo$cluster
plot(PC1,PC2,col=clusterCol,pch=19,xlab="PC1 (16.8%)",ylab="PC2 (12.2%)")
points(PC1[c(3,4,6,8)],PC2[c(3,4,6,8)],pch=19,col="yellow")
legend("bottomright",legend=c("Cluster 1","Cluster 2","Cluster 3","Cluster 4","W

namesCol = factor(sbNames2$V2)
indexes = match(sb$V1,sbNames$V1)
indexes[1:4]=NA
plot(PC1,PC2,pch=19,col=namesCol[indexes])
points(PC1,PC2,pch=19,col=namesCol[indexes])
legend(1,0.1,namesCol[indexes])
```

```
#mahalanobis distances on the first 2 PCs
#following along with Chemometrics as an exercise
PC1 = scale(PC1)
PC2 = scale(PC2)
#declare a centroid
centroidX = (min(PC1) + max(PC1))/2
centroidY = (min(PC2) + max(PC2))/2
MLDist = sqrt((PC1-centroidX)^2 + (PC2-centroidY)^2)
which(MLDist==max(MLDist))
#don't declare centroid
centroidX = 0
centroidY = 0
MLDist = sqrt((PC1-centroidX)^2 + (PC2-centroidY)^2)
which(MLDist==max(MLDist))

totalPCs = sb[, -c(1:2)]
totalPCs = scale(totalPCs)
#Hirearchal clustering of principal components
totalPCsubset = as.data.frame(totalPCs) #want a smaller subset of data
x <- hclust(dist(totalPCsubset,'euclidean'))
dend <- as.dendrogram(x)
dend %>% set("by_labels_branches_col",value=c(235),TF_values = c(2,Inf)) %>% set
legend("topright", legend = c("Wolves", "Dogs"), fill = c(2,"black"))

plot(x,label=FALSE)
y <- hclust(dist(totalPCSubset,'maximum'))
plot(y,label=FALSE)
x$order #returns the index order for the dendogram branches from left to right

#dbscan clustering
#determine the e-distance
minPts = 4
dbscan::kNNdistplot(PCi, k =  minPts)
abline(h = 0.007, lty = 2)
eps = 0.007
db <- fpc::dbscan(PCi, eps = eps, MinPts = minPts)
plot(db,PCi,xlab="PC1 (24.8%)",ylab="PC2 (16.9%)")
points(PC1[c(3,4,6,8)],PC2[c(3,4,6,8)],pch=19,col="yellow")

index <- match(sb$V2,sbNames$V1)
nameLst <- sbNames$V2[index]
datFrame <- data.frame(sb,nameLst)
sampleList <- as.character(unique(sbNames$V2))
a <- length(sampleList)
```

```
#find the terms
avgVals <- matrix(data=NA,nrow=a,ncol=21)
for (i in 1:a){
  temp <- which(str_detect(as.character(nameLst),sampleList[i]))
  temp2 <- totalPCsubset[temp,]
  temp3 <- colMeans(temp2)
  avgVals[i,2:21] <- temp3
  avgVals[i,1] <- sampleList[i]
}
```

References

[1] Single nucleotide polymorphism. `https://www.nature.com/scitable/definition/single-nucleotide-polymorphism-snp-295`. Accessed: March 21st, 2019.

[2] Laura R. Botigue et al. Ancient european dog genomes reveal continuity since the early neolithic. *Nature Communications*, 8(May), 2017.

[3] Richard G. Brereton. The mahalanobis distance and its relationship to principal component scores. *Journal of Chemometrics*, 29(3):143–145, 2015.

[4] Richard Cangelosi and Alain Goriely. Component retention in principal component analysis with application to cdna microarray data. *Biology Direct*, 2, Jan 2007.

[5] Christopher Chang et al. Second-generation plink: rising to the challenge of larger and richer datasets. *GigaScience*, 4(1):1–16, 2015.

[6] Michel Deza and Elena Deza. *Encyclopedia of Distances*. 2009.

[7] Martin Ester, Hans-Peter Kriegel, Jorg Sander, and Xiaowei Xu. A density based algorithm for discovering clusters in large spatial databases with noise. *KDD-96 Proceedings*, 1996.

[8] Martin Ester, Hans-Peter Kriegel, Jorg Sander, and Xiaowei Xu. Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data Mining and Knowledge Discovery*, 1998.

[9] Zhenxin Fan et al. Worldwide patterns of genomic variation and admixture in gray wolves. *Genome Research*, pages 163–173, 2016.

[10] Laurent A. F. Frantz et al. Genomic and archaeological evidence suggests a dual origin of domestic dogs. *Science*, 352(6920):1228–1231, 2016.

[11] Luc Janssens, Liane Giemsch, Ralf Schmitz, Martin Street, Stefan Van Dongen, and Philippe Crombe. A new look at an old dog: Bonn-oberkassel reconsidered. *Journal of Archaeological Science*, 92:126–138, 2018.

[12] Thibaut Jombart. adegenet: a r package for the multivariate analysis of genetic markers. *Bioinformatics*, 24:1403–1405, 2008.

[13] Thibaut Jombart, Sebastien Devillard, and Fancois Balloux. Discriminant analysis of principal components: A new method for the analysis of genetically structured populations. *BMC Genetics*, 10, 2010.

[14] Leonard Kaufman and Peter Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. 1990.

[15] P.A. Lachenbruch and M. Goldstein. Discriminant analysis. *Biometrics*, 35:69–85, 1979.

[16] Greger Larson and Daniel G. Bradley. How much is that in dog years? the advent of canine population genomics. *PLOS Genetics*, 10(1):1–3, 01 2014.

[17] Greger Larson et al. Rethinking dog domestication by integrating genetics, archeology, and biogeography. *Proceedings of the National Academy of Sciences of the United States of America*, 109(23):8878–8883, 2012.

[18] David MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[19] Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[20] Amanda L. Pendleton, Feichen Shen, Angela M. Taravella, Sarah Emery, Krishna R. Veeramah, Adam R. Boyko, and Jeffrey M. Kidd. Comparison of village dog and wolf genomes highlights the pivotal role of the neural crest in dog domestication. *BMC Biology*, 16, 2018.

[21] Malgorzata Pilot et al. On the origin of mongrels: Evolutionary history of free-breeding dogs in eurasia. *Proceedings of the Royal Society B: Biological Sciences*, 282(1820), 2015.

[22] Erich Schubert, Martin Ester, Hans-Peter Kriegel, Jorg Sander, and Xiaowei Xu. Dbscan revisited, revisited: Why and how you should (still) use dbscan. *ACM Transactions on Database Systems*, 42, 2017.

[23] Laura M. Shannon et al. Genetic structure in village dogs reveals a central asian domestication origin. *Proceedings of the National Academy of Sciences*, 112(44):13639–13644, 2015.

[24] Jian Yang, S. Hong Lee, Michael E. Goddard, and Peter M. Visscher. Gcta: A tool for genome-wide complex trait analysis. *American Journal of Human Genetics*, 88:76–82, 2011.