Università degli Studi di Milano
Data Science and Economics (LM-91)

Statistical Learning

Antarctic Penguins

Species Exploration

Shihab Hamati
Nov 3, 2022

Abstract

Taxonomists and biologists till this day discover new species of animals and plants. To formally differentiate species, extensive, expensive, and time-consuming effort is required to acquire genetic samples and process at specialized laboratories. In the meanwhile, for faster results and a more convincing proposal to access advanced laboratories and research funds, scientists may resort to machine learning methods to explore patterns in their raw data. Such an example is in species discovery.

This paper exploits unsupervised machine learning techniques to explore Antarctic penguins' species from the Palmer Penguins dataset. Multiple clustering techniques were used on the raw scaled numerical dataset to identify distinct groups of observations. The data constitutes of four different physical measurements of the birds.

Problem Statement

When studying new species, taxonomists and biologists require the use of expensive and time-consuming resources to make definitive discoveries of new species. However, through the use of modern unsupervised machine learning algorithms it is possible to make preliminary discoveries of patterns across the collected data weeks, months, or even years prior to the conclusion of the formal genetic analysis. Not only that, but such techniques can handle a large amount of raw data and guide the researchers towards patterns that would otherwise be difficult or impossible to notice as a human.

Objective

The aim of this analysis is to explore the differences and similarities between the observations of the Palmer Archipelago Penguins dataset. Applying different clustering techniques, the aim is to distinguish the penguins based on some traits that are common to each group (or cluster) but significantly different across groups.

Dataset

The data was originally published in PLoS ONE by Gorman, Williams, and Fraser in 2014 (link), and the full datasets in 2020 in the Environmental Data Initiative. The dataset used in this study is directly obtained from the `palmerpenguins` R package. There are 344 records and 8 columns.

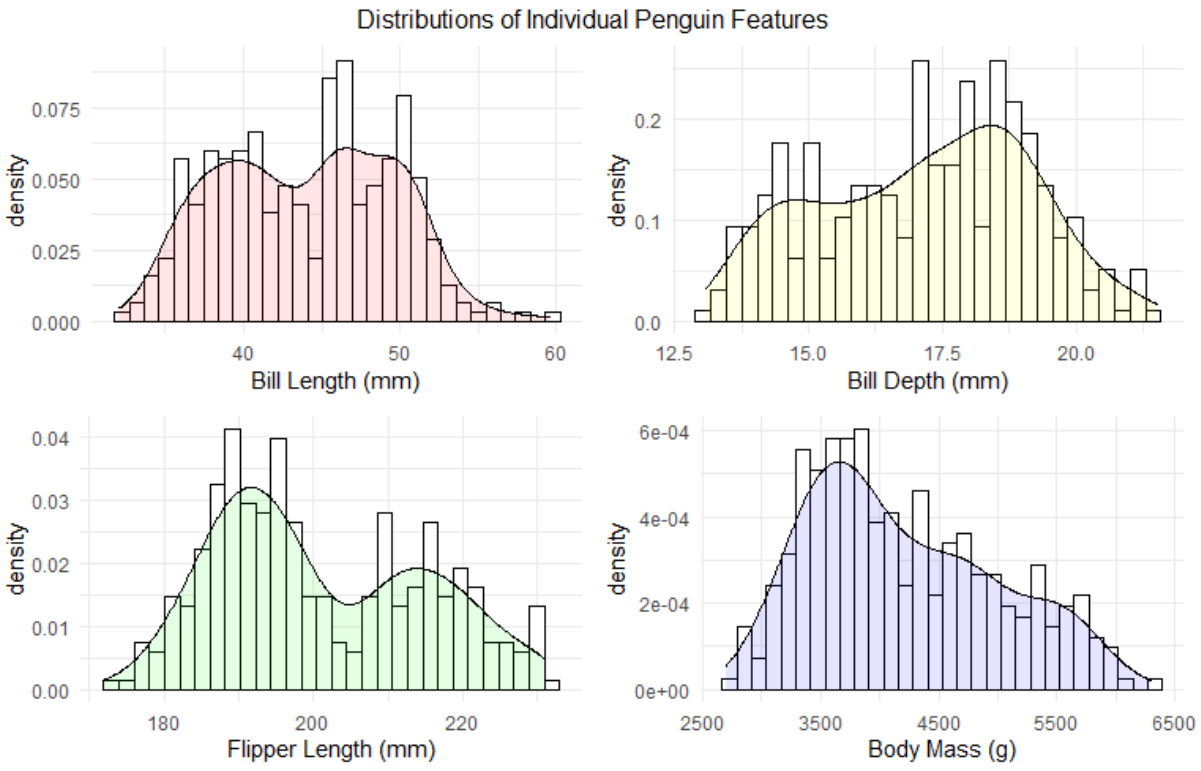
- Species: *multiclass categorical*, describes which of three species a bird belongs to – this column is dropped from all the unsupervised analyses, and used in lieu of subject matter experts only ex post facto to understand the clustering and decomposition results
- Island: *multiclass categorical*, the Antarctic region in which an observation was made
- Bill Length (mm): *numerical*, measures the length of the peak, from the head and towards to observer facing it
- Bill Depth (mm): *numerical*, measures the dimension of the beak from top to bottom
- Flipper Length (mm): *numerical*, the length of the "wing" or "arm"
- Body Mass (g): *numerical*
- Sex: *binary categorical*, male or female
- Year: *numerical,* year of recorded measurement (between 2007-2009)

Exploratory Data Analysis

The dataset is retrieved directly from within the `palmerpenguins` R package. It consists of 344 records and 8 columns. It has some missing values, either for the entire set of numeric features or for some. At any rate, the entire observation is dropped if it is not complete. This results in dropping only 11 observations. Below is a transposed sample of the first 5 records.

Species	Adelie	Adelie	Adelie	Adelie	Adelie
Island	Torgersen	Torgersen	Torgersen	Torgersen	Torgersen
Bill Length (mm)	39.1	39.5	40.3	36.7	39.3
Bill Depth (mm)	18.7	17.4	18	19.3	20.6
Flipper Length (mm)	181	186	195	193	190
Body Mass (g)	3750	3800	3250	3450	3650
Sex	male	female	female	female	male
Year	2007	2007	2007	2007	2007

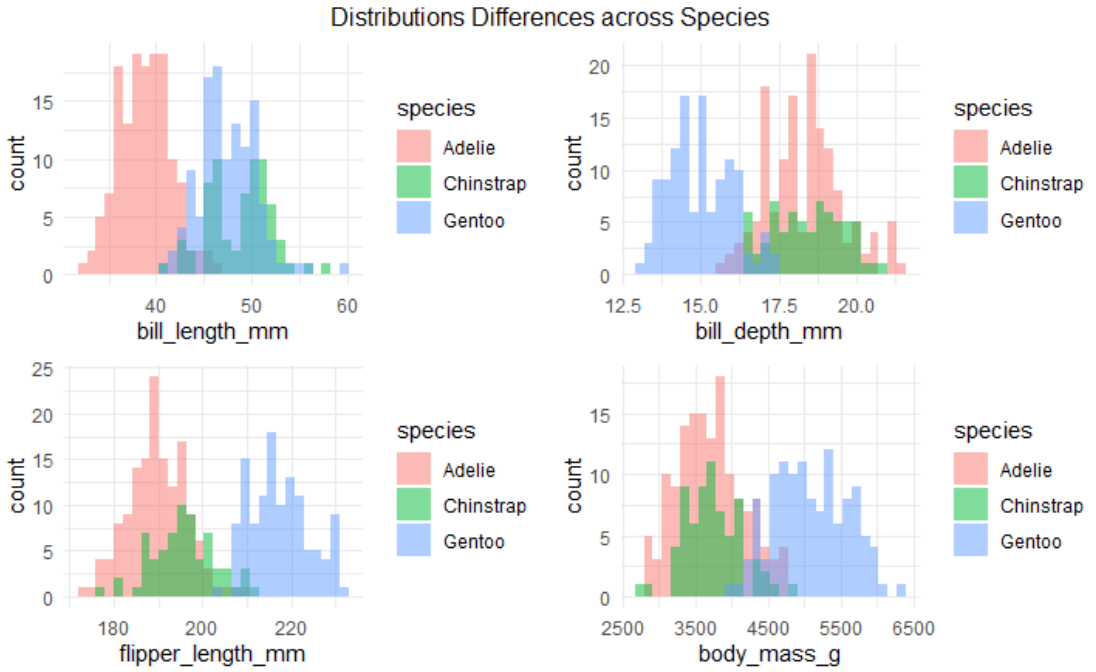
The histograms and distribution plots of each feature are plotted below.

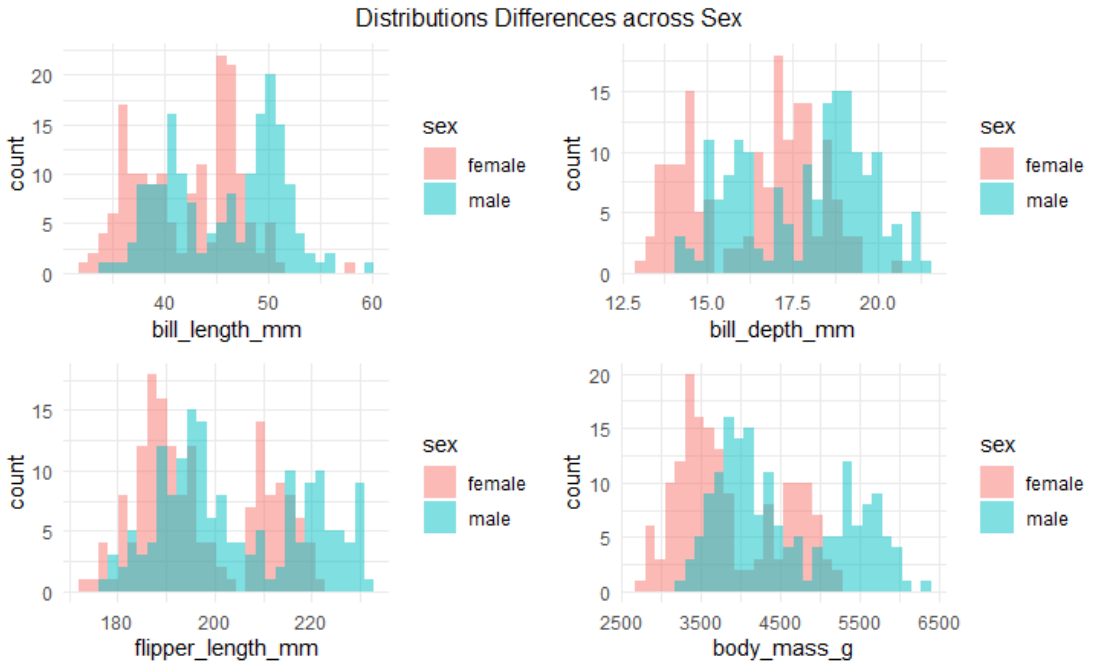


Contrast Across Species and Sex

The histograms are plotted again for each feature, but separated by species and sex. While the Adelie species appear to have shorter bills and the Gentoo shallower bills, longer flippers, and heavier weight, the Chinstrap is not easily distinguishable as such.

This data will not be used in the unsupervised learning section. It will act later on in place of field experts to guide the understanding of the generated clusters or principal components.

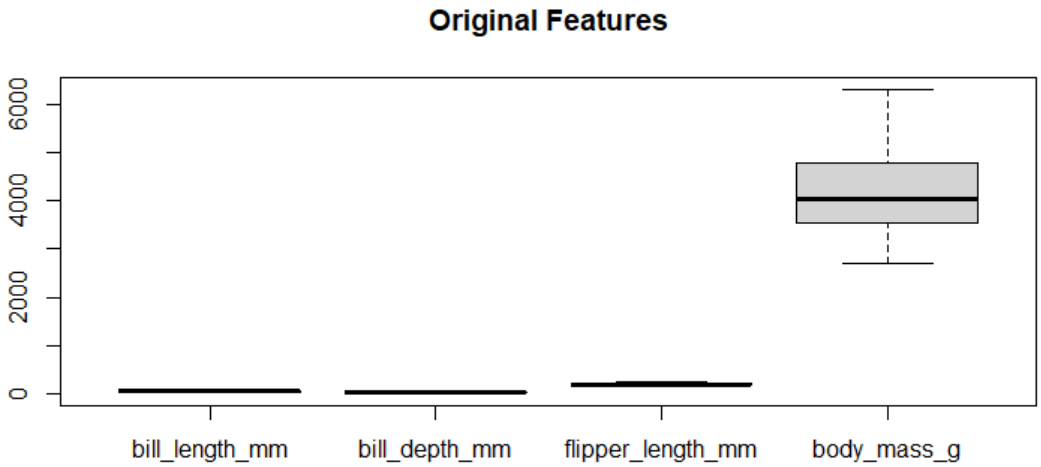




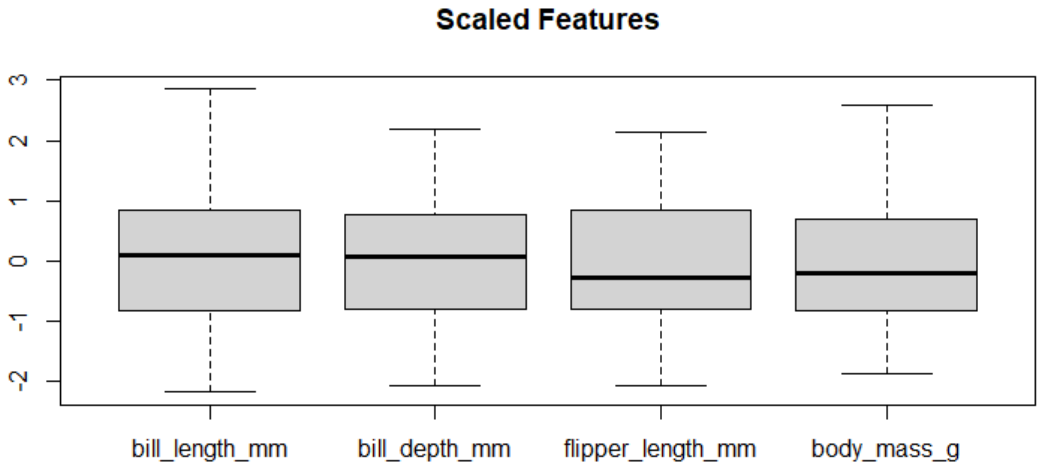
Data Scale

The four numeric features are used for the purpose of the unsupervised analyses. The summary statistics of the features are below.

bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g
Min. :32.10	Min. :13.10	Min. :172	Min. :2700
1st Qu.:39.50	1st Qu.:15.60	1st Qu.:190	1st Qu.:3550
Median :44.50	Median :17.30	Median :197	Median :4050
Mean :43.99	Mean :17.16	Mean :201	Mean :4207
3rd Qu.:48.60	3rd Qu.:18.70	3rd Qu.:213	3rd Qu.:4775
Max. :59.60	Max. :21.50	Max. :231	Max. :6300



The scales of the features differ wildly from each other (ranging from means of 17.16 up to 4207). The data must be scaled prior to its use in unsupervised learning techniques. This is because distances and variances across features are used, and differences in scale will masquerade as false dominant values. Each feature is centered by its mean and scaled by its standard deviation to harmonize them all together.



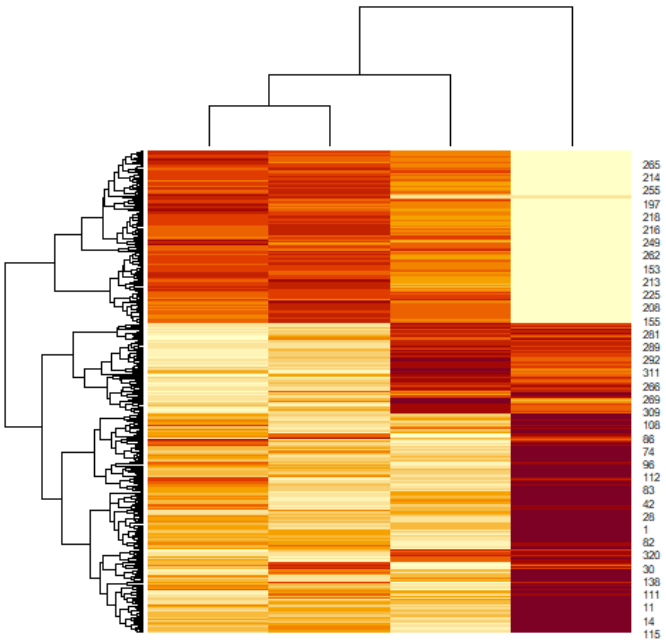
Correlation of Physical Features

The correlation heatmap below highlights important relations between features. The flipper length and the body mass are highly correlated. This is logical since the flipper length is a good portion of a penguin's height, and consequently the larger the penguin's size the heavier it is expected to weigh. This flipper length is also strongly correlated to the bill length. However, it appears that the larger the penguin, the narrower its beak. This reflects the function of the pinguin's beak, which is shaped like a narrow hook to grab and hold on to fish, which is a main source of food.

Clusterability

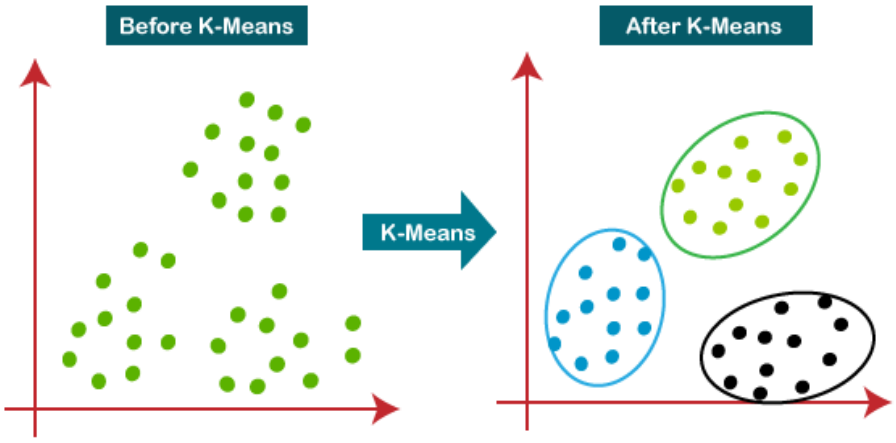
The Hopkins statistic measures the clustering tendency of the dataset. Values between 0.7 and 1 indicate a tendency to cluster. The penguins dataset has a Hopkins statistic greater than 0.9 thus exhibiting a clustering tendency.

A heatmap visualization of the dataset colors each value in each column in each record from a continuous color scale in proportion to its value. Creating a heatmap from the scaled dataset appears to indicate 3 clusters: 2 of which are clearly distinguishable (observe top left brown lump and bottom right brown strip) and a third which is harder to extract from the others. As there are only 2 genders of penguins, this likely points to a grouping along a different characteristic (perhaps species or island).



K-means Clustering

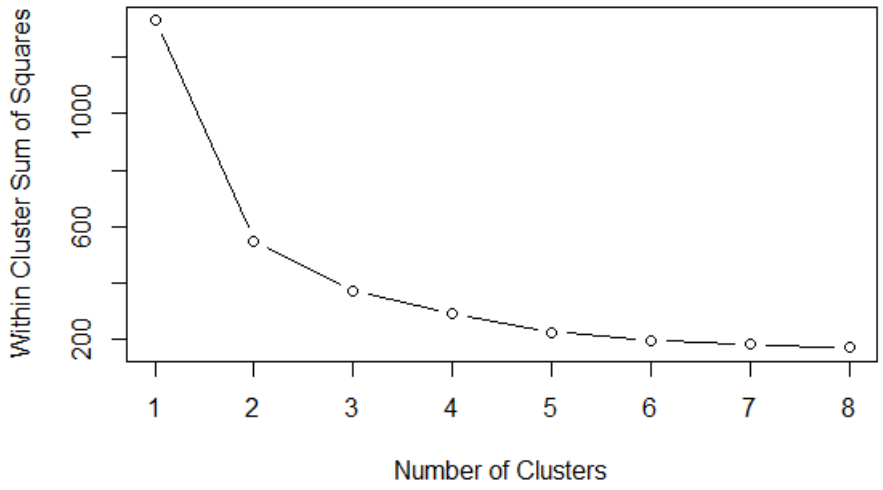
This method attempts to partition the observations into a pre-specified number of clusters. Below is a sketch illustrating the expected output of a k-means clustering with the pre-specified number of clusters of 3.



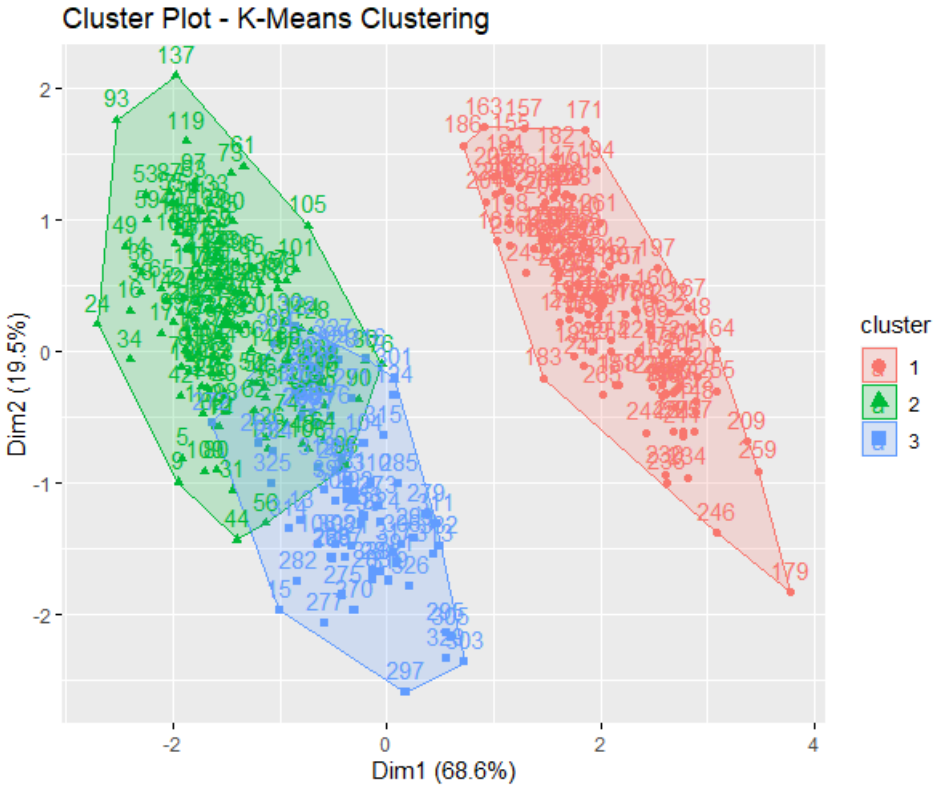
K-means achieves its objectives by iteratively honing on the best pre-specified number of cluster centers that minimize within-cluster variation. Euclidean distances between scaled observations were computed in 4D space.

Number of Clusters, Method 1: WCSS

To identify an appropriate number of clusters, the Elbow method was used on the Within Cluster Sum of Squares (WCSS) plot. This methods is rarely as clear cut as in theory. From the plot below, k=3 was chosen as the elbow point.



The result of running the K-means clustering algorithm is plotted below on a lower 2D space. The two dimensions represent the strongest decomposed components of the dataset.



Interpretation of Clusters

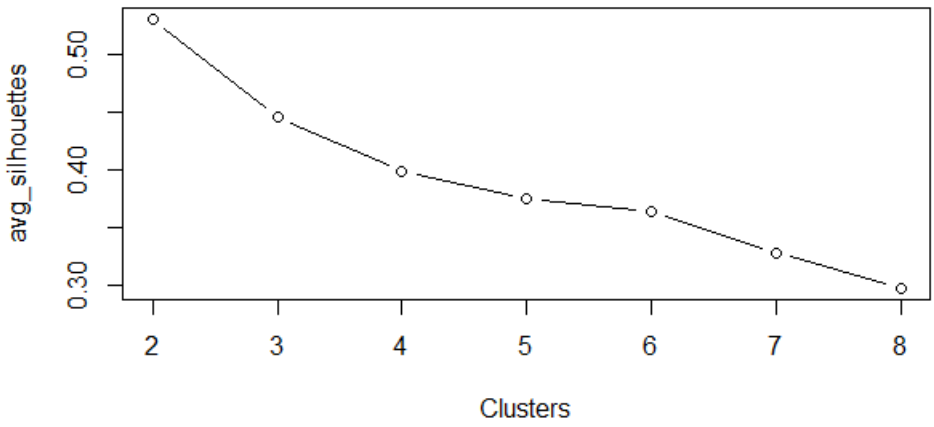
Unsupervised learning is much more subjective than supervised learning due to the absence of a target label. This is why close collaboration between data scientists and field experts is paramount to interpret or understand what the uncovered patterns are.

In this case, the clustering result is compared to the hidden species features. It appears that the K-means algorithm on the scaled numeric dataset uncovered that there are three distinct penguin species in the regions explored by the scientists.

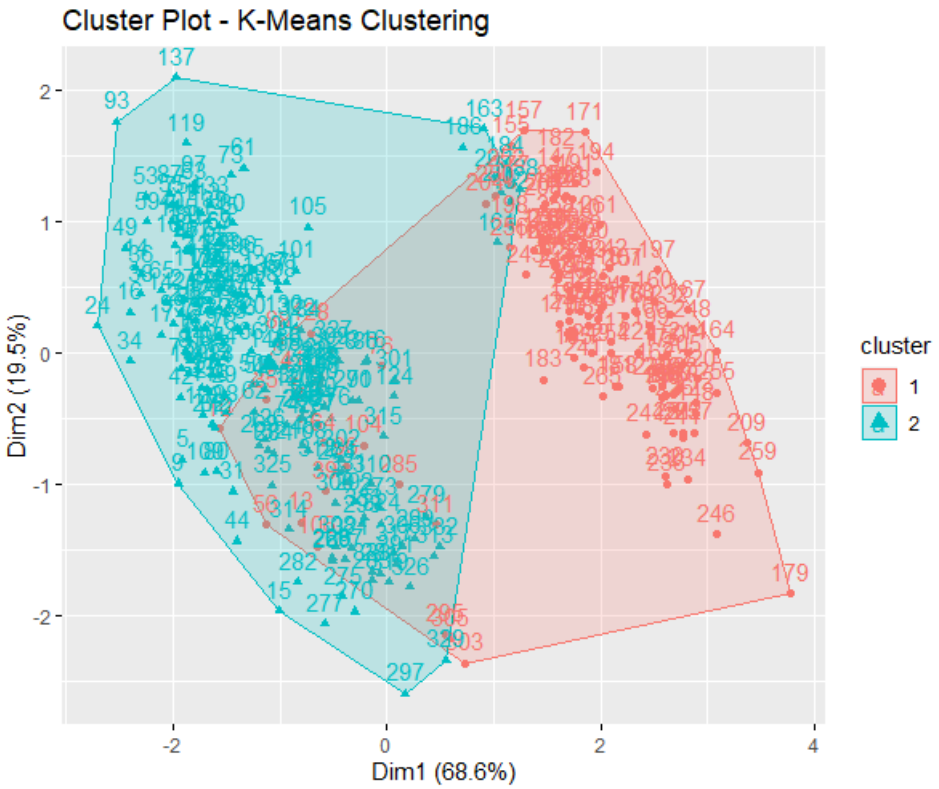
	Adelie	Chinstrap	Gentoo
1	0	0	119
2	139	5	0
3	7	63	0

Number of Clusters, Method 2: Silhouette

The Silhouette Method is a second approach to identifying an optimal number of clusters. This is done by computing a silhouette score for each observation, measuring its similarity to its own cluster versus other clusters.



The average silhouette plot suggests k=2 clusters (highest average silhouette score). The k-means clustering is repeating accordingly, and the following clusters are obtained.



In this case, the clusters are compared across the hidden categorical features (which act as a replacement of subject matter experts in this case). It appears with two clusters, the algorithm is distinguishing Gentoo vs non-Gentoo species. Most Gentoo live in Biscoe island (the two features are highly correlated, which makes sense since members of the same species form social groups that require geographical proximity).

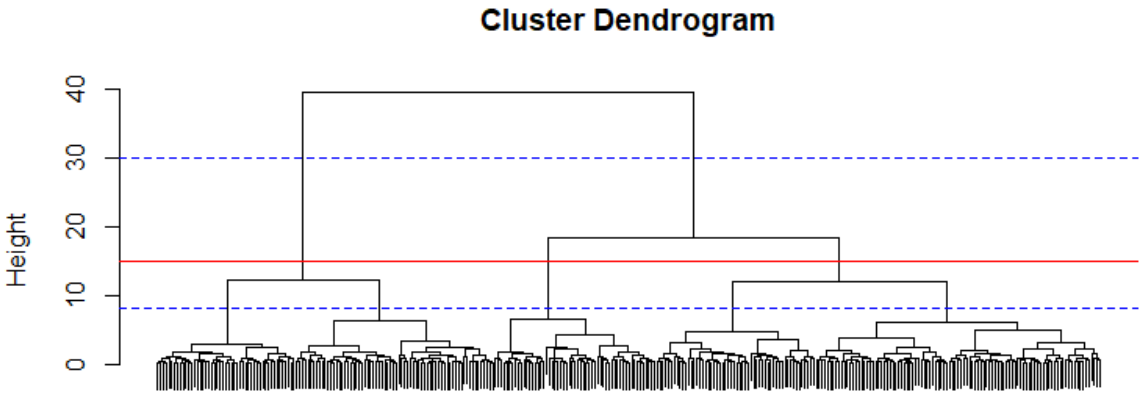
	female	male
1	50	80
2	115	88

	Adelie	Chinstrap	Gentoo
1	14	5	111
2	132	63	8

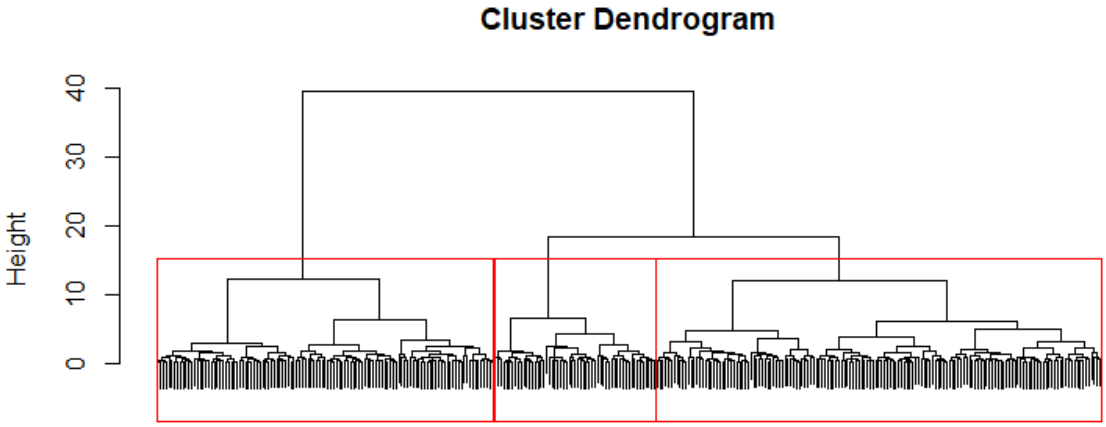
	Biscoe	Dream	Torgersen
1	115	10	5
2	48	113	42

Hierarchical Clustering

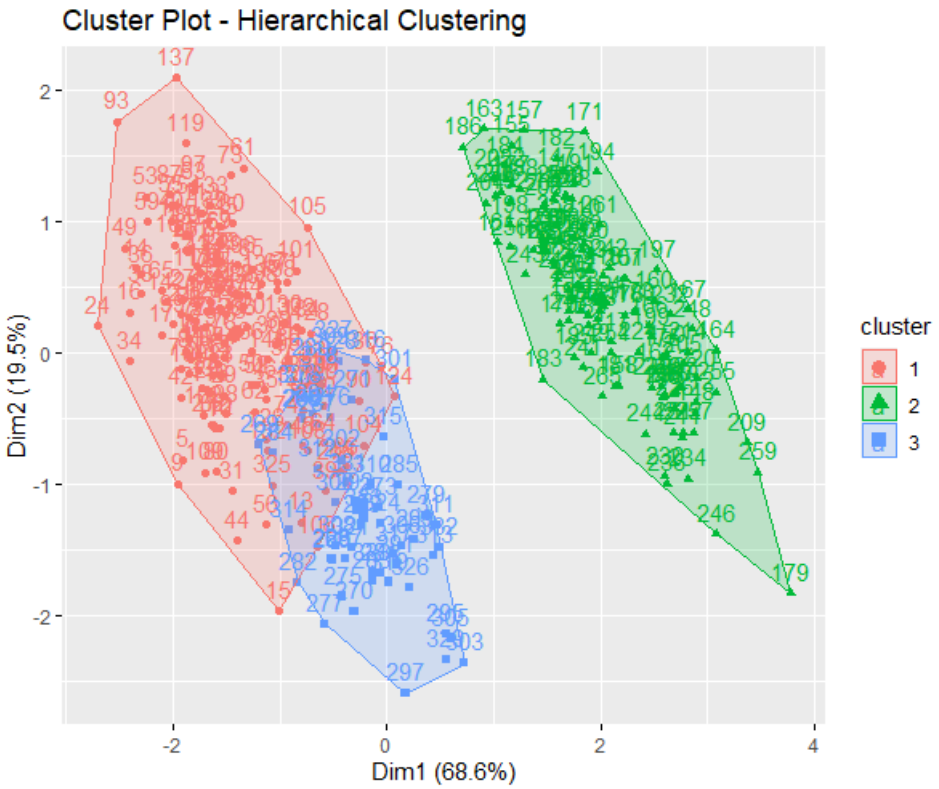
This is an alternative to k-means, with the advantage that it does not require a prior decision on the number of clusters. A dendrogram is built starting from the leaves (individual observations) and combining clusters all the way up to the root. In this way, the dendrogram displays all cluster sizes, and the analyst can visually choose an appropriate level to draw a horizontal line and count how many clusters there are at that level. The following dendrogram was generated using Ward's method.



Choices for 2, 3, or 5 clusters are all justifiable. The choice of 3 clusters is subjectively chosen as its mid-line is closest to the midpoint of the dendrogram's height. The second dendrogram visualizes the 3 cluster.



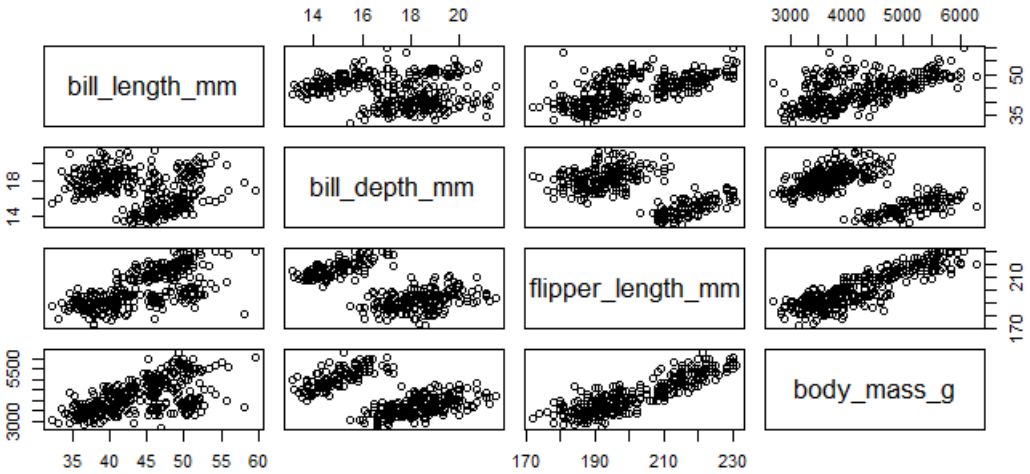
The result is plotted below on a lower 2D space. The two dimensions represent the strongest decomposed components of the dataset. It is similar to the k-means (k=3) clustering result.



Also as before, hierarchical clustering (k=3) appears to have identified the three difference species of penguins.

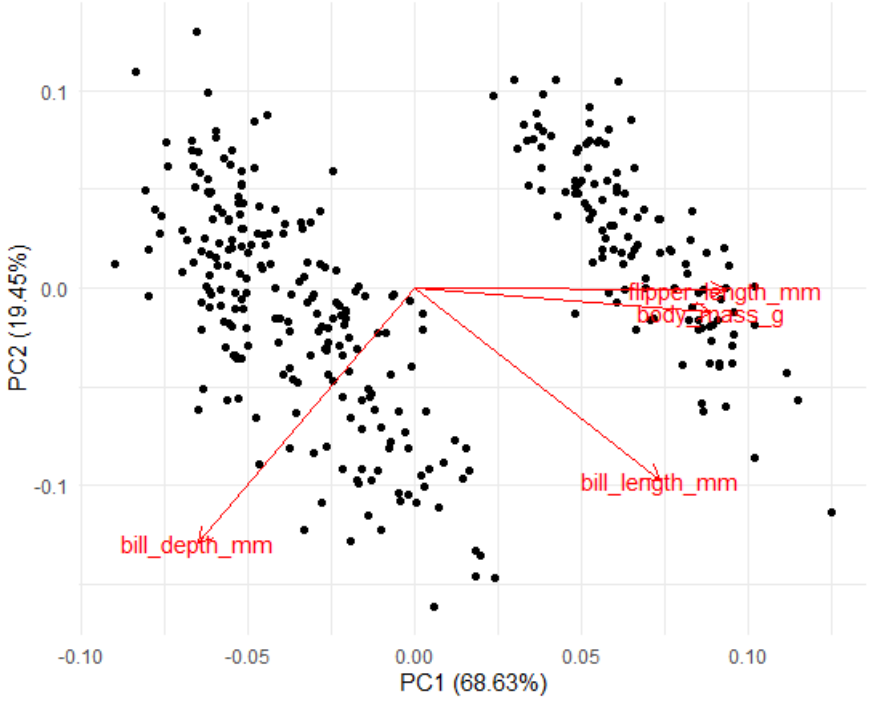
	Adelie	Chinstrap	Gentoo
1	146	11	0
2	0	0	119
3	0	57	0

Principal Component Analysis



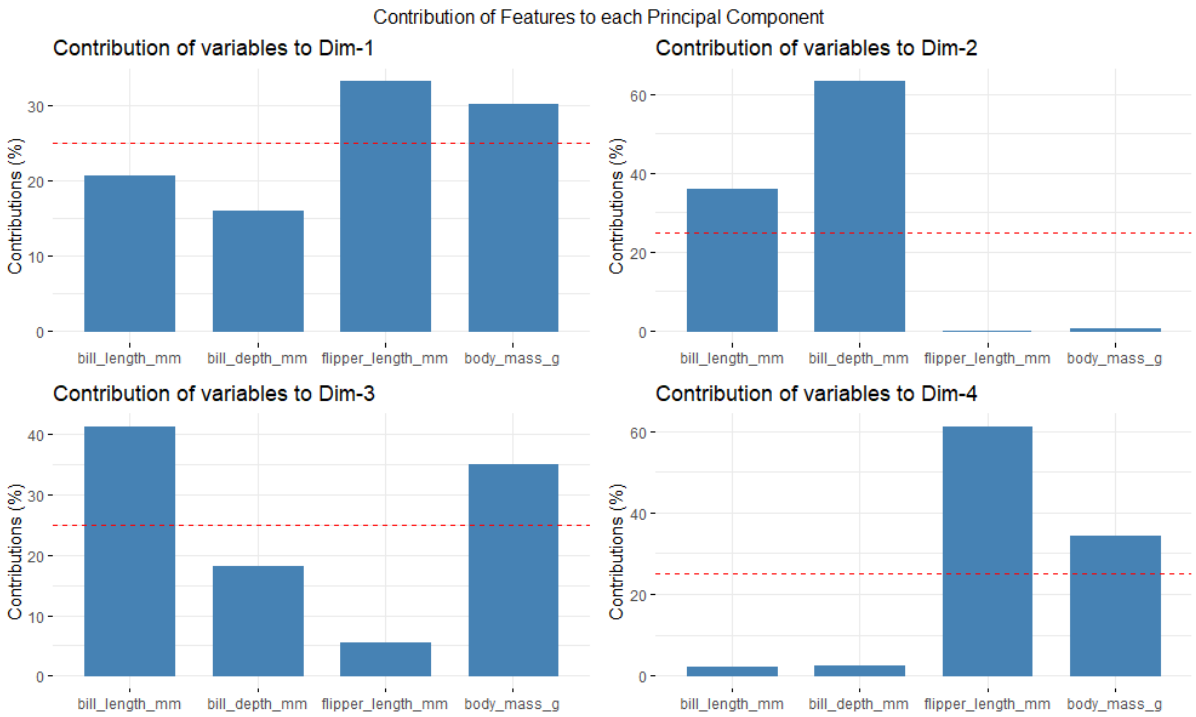
As seen from the page plots in the previous page, there is not one dominant physical features to describe variations across penguins sufficiently. Principal Component Analysis (PCA) is another unsupervised machine learning tool that produces a lower dimensional representation of the dataset. It achieves this by identifying the axis (or rotation) in the original feature space axes that has maximum variance. The resulting remaining principal components are orthogonal and uncorrelated. The first component explains more of the overall variance than any other original features.

This can be used to reduce the dimensionality of datasets with a huge number of features, or to densify the variance of the data into fewer components to potentially improve the performance of the clustering algorithms.



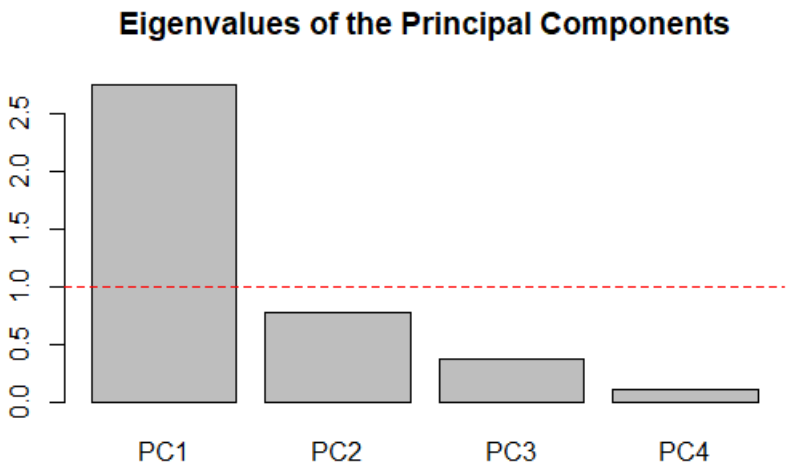
Above is a plot of the observations on a 2-D space made of the first two principal components. The red arrows represent the loadings. For example, the most important feature of the penguins is mostly reliant on its flipper length and body mass (i.e., the largest variation across penguins is their overall size rather than their beaks).

The following graphs visually represent how much each of the original features contributes to each principal components. The components have a descending share of variance explained.



Kaiser Criterion

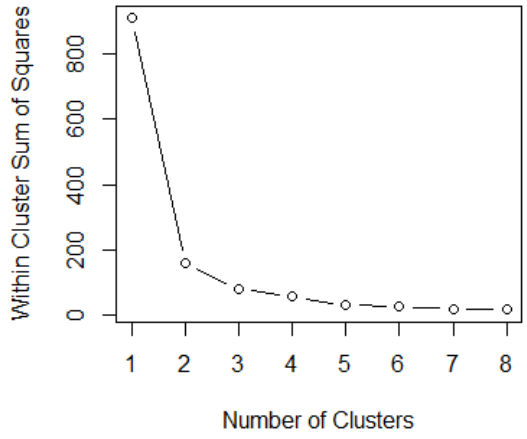
One way to determine how many principal components to keep is to apply the Kaiser criterion. It suggests keeping only PCs whose eigenvalues is greater than 1. PC1 is the only principal component that satisfies the Kaiser criterions.

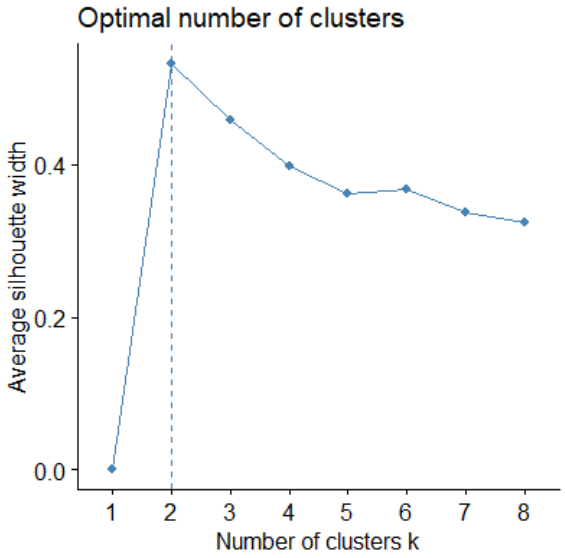


Clustering after PCA

Cluster after decomposing the dataset using PCA is another popular approach, especially for large datasets with a large number of features. The outcome may be better or worse since there are two opposite phenomena at play. The reduction in number of features by keeping only the top PCs that contain the most variability of the data reduces the curse of dimensionality. On the other hand, there is information loss by dropping any principal components.

The optimal number of clusters based on PC1 data only appears to be k=2, based on both the elbow WCSS and silhouette methods.





K-means clustering on just this one variable PC1 is able to achieve a comparable clustering for k=2. It also does an acceptable job at clustering into 3 groups (roughly matching the species) again based only on one feature compared to four.

k=2

K-means (full data)

	Adelie	Chinstrap	Gentoo
1	14	5	111
2	132	63	8

PCA (only PC1)

	Adelie	Chinstrap	Gentoo
1	0	7	119
2	146	61	0

k=3

	Adelie	Chinstrap	Gentoo
1	0	0	119
2	139	5	0
3	7	63	0

	Adelie	Chinstrap	Gentoo
1	37	59	1
2	109	9	0
3	0	0	118

Key Findings

- Different methods disagree on the optimal number of clusters. For k=3, both k-means and hierarchical clustering are able to split the observed penguins into meaningful groupings, according to species
- At k=2, the clustering methods split the observations into larger penguins vs smaller penguins. The size of the bird (directly related to its flipper length and body mass) are correlated and confirmed to be the features with the highest loadings in PC1
- PCA was able to replicate the clustering performance at both k=2 and k=3 to a satisfactory level with only one component, i.e., 75% less columns

Conclusion

Unsupervised learning techniques are subjective by nature, which makes them harder to work with than supervised methods. However, many real life situations involve exploring large datasets in hopes of discovering underlying patterns. The usefulness of clustering and PCA were successfully exhibited in the context of taxonomy. Had the researchers used these techniques prior to any expensive genetic testing, it would have guided them in the right direction for their species classification task.

Appendix: R Code

```
################################################################################
# PROJECT DETAILS


#-------------------------------------------------------------------------------
# ADMINSTRATIVE

# Name:         Shihab Hamati
# Matricola:    985941

# Module:       Statistical Learning
# Exam Date:    03 Nov 2022

# Part 2:       Unsupervised Learning

#-------------------------------------------------------------------------------
# REFERENCES

# Description:
# - A dataset consisting of 4 physical measurements of Antarctic penguins
# - It is known to consist of 3 species
# - This information is hidden from the clustering functions
# - It is used in lieu of experts to explore the meaning of clustering results



################################################################################
# LIBRARIES

library(palmerpenguins)
library(ggplot2)
library(gridExtra)
library(corrplot)
library(factoextra)
library(hopkins)
library(cluster)
library(ggfortify)



################################################################################
# DATA SETUP

head(penguins)
str(penguins)
summary(penguins)


#-------------------------------------------------------------------------------
# CLEAN DATASET

penguins[complete.cases(penguins), ]
data_full <- na.omit(penguins)
colnames(data_full)
data <- data_full[,3:6]
summary(data)
```

```
################################################################################
# EXPLORATORY DATA ANALYSIS (EDA)

#-------------------------------------------------------------------------------
# EACH NUMERIC FEATURE INDEPENDENTLY

p1a <- ggplot(data = data, aes(x = bill_length_mm)) +
  geom_histogram(aes(y = ..density..), colour="black", fill="white") +
  geom_density(alpha = 0.1, fill = "red") +
  xlab("Bill Length (mm)") +
  theme_minimal()

p1b <- ggplot(data = data, aes(x = bill_depth_mm)) +
  geom_histogram(aes(y = ..density..), colour="black", fill="white") +
  geom_density(alpha = 0.1, fill = "yellow") +
  xlab("Bill Depth (mm)") +
  theme_minimal()

p1c <- ggplot(data = data, aes(x = flipper_length_mm)) +
  geom_histogram(aes(y = ..density..), colour="black", fill="white") +
  geom_density(alpha = 0.1, fill = "green") +
  xlab("Flipper Length (mm)") +
  theme_minimal()

p1d <- ggplot(data = data, aes(x = body_mass_g)) +
  geom_histogram(aes(y = ..density..), colour="black", fill="white") +
  geom_density(alpha = 0.1, fill = "blue") +
  xlab("Body Mass (g)") +
  theme_minimal()

grid.arrange(p1a, p1b, p1c, p1d,
             ncol=2, top = "Distributions of Individual Penguin Features")


#-------------------------------------------------------------------------------
# HOW FEATURES DIFFER ACROSS SPECIES

p2a <- ggplot(data = data_full, aes(x = bill_length_mm,
                                    group = species, fill = species)) +
  geom_histogram(position = "identity", alpha = 0.5) +
  theme_minimal()

p2b <- ggplot(data = data_full, aes(x = bill_depth_mm,
                                    group = species, fill = species)) +
  geom_histogram(position = "identity", alpha = 0.5) +
  theme_minimal()

p2c <- ggplot(data = data_full, aes(x = flipper_length_mm,
                                    group = species, fill = species)) +
  geom_histogram(position = "identity", alpha = 0.5) +
  theme_minimal()

p2d <- ggplot(data = data_full, aes(x = body_mass_g,
                                    group = species, fill = species)) +
  geom_histogram(position = "identity", alpha = 0.5) +
  theme_minimal()

grid.arrange(p2a, p2b, p2c, p2d,
             ncol = 2, top = "Distributions Differences across Species")


#-------------------------------------------------------------------------------
# HOW FEATURES DIFFER ACROSS SEX

p3a <- ggplot(data = data_full, aes(x = bill_length_mm,
                                    group = sex, fill = sex)) +
  geom_histogram(position = "identity", alpha = 0.5) +
  theme_minimal()
```

```
p3b <- ggplot(data = data_full, aes(x = bill_depth_mm,
                                    group = sex, fill = sex)) +
  geom_histogram(position = "identity", alpha = 0.5) +
  theme_minimal()

p3c <- ggplot(data = data_full, aes(x = flipper_length_mm,
                                    group = sex, fill = sex)) +
  geom_histogram(position = "identity", alpha = 0.5) +
  theme_minimal()

p3d <- ggplot(data = data_full, aes(x = body_mass_g,
                                    group = sex, fill = sex)) +
  geom_histogram(position = "identity", alpha = 0.5) +
  theme_minimal()

grid.arrange(p3a, p3b, p3c, p3d,
             ncol = 2, top = "Distributions Differences across Sex")


#-------------------------------------------------------------------------------
# Importance of Scaling Features

p4a <- boxplot(data, main = "Original Features")

data_scaled <- data.frame(scale(data))
summary(data_scaled)

p4b <- boxplot(data_scaled, main = "Scaled Features")


#-------------------------------------------------------------------------------
# Corellelogram

corrplot(cor(data_scaled), method = 'number')



################################################################################
# CLUSTERING:
# K-MEANS & HIERARCHICAL

#-------------------------------------------------------------------------------
# Clusterability Test

set.seed(100)
hopkins(data_scaled)
# indicates our dataset is very clusterable
# since 0.96 is between 0.7-1, refer to documentation

#-------------------------------------------------------------------------------
# Visualizing Variations across all Features

heatmap(as.matrix(data_scaled), cexCol = 1)
# appears we might have 2-3 clusters

#-------------------------------------------------------------------------------
# Identifying Optimal Number of Clusters
# METHOD 1: Elbow Method of WITHIN CLUSTER SUM of SQUARES (WCSS)

wcssplot <- function(data, kmax){
  wcss <- (nrow(data) - 1) * sum(apply(data, 2, var))
  for (k in 2: kmax){
    set.seed(100)
    wcss[k] <- sum(kmeans(data, centers = k)$withinss)}
  plot(1:kmax, wcss, type="b", xlab="Number of Clusters",
       ylab="Within Cluster Sum of Squares")
  }
```

```
wcssplot(data_scaled, kmax = 8)
# k = 3 appears to be an elbow in the WCSS plot

#.....................................................................
# K-MEANS CLUSTERING (k = 3)

set.seed(100)
kmeans_fit <- kmeans(data_scaled, 3)

fviz_cluster(kmeans_fit, data = data_scaled,
             main = "Cluster Plot - K-Means Clustering")

#.....................................................................
# Exploring the meaning of the clusters
# - usually a target variable might not be present
# - the target variable is used after the clustering, in lieu of experts

table(kmeans_fit$cluster, data_full$species)
# clusters appear to line up with species


#---------------------------------------------------------------------------
# Identifying Optimal Number of Clusters
# METHOD 2: AVERAGE SILHOUETTE SCORE

sil_score <- function(k){
  km <- kmeans(data_scaled, centers = k, nstart = 25)
  ss <- silhouette(km$cluster, dist(data_scaled))
  mean(ss[, 3])
}

silplot <- function(kmax){
  k <- 2:kmax
  avg_silhouettes <- sapply(k, sil_score)
  plot(k,avg_silhouettes, type = 'b', xlab = "Clusters")
}

silplot(8)
# k = 2 clusters is optimal as it has highest avg sil score

#.....................................................................
# K-Means Clustering (k = 2)

set.seed(100)
kmeans_fit_k2 <- kmeans(data, 2)

fviz_cluster(kmeans_fit_k2, data = data,
             main = "Cluster Plot - K-Means Clustering")

#.....................................................................
# Exploring the meaning of the 2 clusters

table(kmeans_fit_k2$cluster, data_full$species)
# k = 2 seems to discriminate between Gentoo species and non-Gentoo

table(kmeans_fit_k2$cluster, data_full$sex)
# not clustered across gender

table(kmeans_fit_k2$cluster, data_full$island)
# appears to cluster across Biscoe island and the other 2 islands

table(data_full$species, data_full$island)
# however the species distribution is highly correlated to island
# so this is a Gentoo vs non-Gentoo species clustering
```

```
#-------------------------------------------------------------------------------
# Identifying Optimal Number of Clusters
# METHOD 3: HIERARCHICAL CLUSTERING

distances <- dist(data_scaled, method = "euclidean")
hier_fit <- hclust(distances, method="ward.D2")

plot(hier_fit, labels = FALSE)

# Multiple cut-off heights appear to yield an acceptable clustering result
abline(h = 30, col="blue", lty = 2)
abline(h = 15, col="red") # middle option is chosen, yields k = 3
abline(h = 8, col="blue", lty = 2)

# Re-plot dendogram with the 3 clusters
plot(hier_fit, labels = FALSE)
rect.hclust(hier_fit, k = 3, border = "red")

# Assign the 3 clusters
h_clusters <- cutree(hier_fit, k = 3)

fviz_cluster(list(data = data_scaled, cluster = h_clusters),
             main = "Cluster Plot - Hierarchical Clustering")

#.....................................................................
# Exploring the meaning of the 3 clusters

table(h_clusters, data_full$species) # also appears to align with species



#############################################################################
# PRINCIPAL COMPONENT ANALYSIS (PCA)

pca_decomp <- prcomp(data_scaled)

pairs(data_full[colnames(data)])

# Plotting all pairs of numeric features (original data)
colors <- c('red', 'green', 'blue')[unclass(data_full$species)]
pairs(data_full[colnames(data)],
      main = "Original Data Combinations (colored by species)",
      col = colors)

# Plotting across the top 2 principal components
autoplot(pca_decomp, data = data_full, colour = 'species') + theme_minimal()
autoplot(pca_decomp, data = data_full) + theme_minimal()

# ..with loadings
autoplot(pca_decomp, data = data_full, colour = 'species',
         loadings = TRUE, loadings.label = TRUE) + theme_minimal()
autoplot(pca_decomp, data = data_full,
         loadings = TRUE, loadings.label = TRUE) + theme_minimal()

#.....................................................................
# Extract variance explained by each PC
# and the composition of each PC

pca_var <- get_pca_var(pca_decomp)

p5a <- fviz_contrib(pca_decomp, "var", axes = 1, xtickslab.rt = 0) +
  scale_x_discrete(limits = colnames(data_scaled))

p5b <- fviz_contrib(pca_decomp, "var", axes = 2, xtickslab.rt = 0) +
  scale_x_discrete(limits = colnames(data_scaled))

p5c <- fviz_contrib(pca_decomp, "var", axes = 3, xtickslab.rt = 0) +
  scale_x_discrete(limits = colnames(data_scaled))
```

```
p5d <- fviz_contrib(pca_decomp, "var", axes = 4, xtickslab.rt = 0) +
  scale_x_discrete(limits = colnames(data_scaled))

grid.arrange(p5a, p5b, p5c, p5d, ncol=2,
             top = "Contribution of Features to each Principal Component")

#.....................................................................
# Kaiser Criterion: How many PCs to keep?

pca_eigen <- get_eigenvalue(pca_decomp)
barplot(pca_eigen$eigenvalue, names.arg = c("PC1", "PC2", "PC3", "PC4"),
        main = "Eigenvalues of the Principal Components")
abline(h = 1, col = "red", lty = 2)
# Kaiser criterion: keep PCs with Eigenvalues > 1
# Only PC1 is retained

#.....................................................................
# Number of Clusters

wcssplot(pca_decomp$x[,1, drop = FALSE], kmax = 8) # elbow wcss
fviz_nbclust(pca_decomp$x, FUNcluster = kmeans, k.max = 8) # silhouette

# Elbow WCSS and Silhouette both suggest k = 2 clusters for PC1 data

#.....................................................................
# Comparisons (PCA+Clustering vs Clustering on full dataset)

# Comparing performance of 2-means clustering: PC1-reduced dataset vs Original
set.seed(100)
pca_kmeans_fit <- kmeans(pca_decomp$x[,1], 2)
table(pca_kmeans_fit$cluster, data_full$species)
table(kmeans_fit_k2$cluster, data_full$species)

# Performing K-means clustering using the suggested 1 PC components
# yields slightly less species clustering than using the whole dataset
# But if the range of features was huge, this could provide an efficient
# approach to achieving comparable clustering with less features and dimensions

set.seed(100)
pca_kmeans_fit <- kmeans(pca_decomp$x[,1], 3)
table(pca_kmeans_fit$cluster, data_full$species)
table(kmeans_fit$cluster, data_full$species)


#######################################################################################
# COMMENTARY

# While the data plots and Hopkins test suggest the clusterability of the data,
# different approaches point to varying optimal number of clusters, albeit
# close: 2 or 3 cluster groups

# In a real scenario, and since no "target" or "response" variable exists to
# give feedback to tuning the hyperparameter k (num of clusters), the groups
# have to be examined by experts in an attempt to understand what traits
# differentiate the clusters and how many clusters is practical and relevant



#######################################################################################
# END
```