Unsupervised Machine Learning: Exploration of mRNA Gene Clustering Patterns within Salmoninae through Alignment-Based Sequence Clustering

BINF6210 Assignment #5

Yuqi Li

2023-12-08

Contents

Introduction **2**

Description of Dataset **2**

Github Link **2**

Code Section 1: Exploratory Data Analysis (EDA) **3**

1) Loading Library . . . 3
2) Data Acquisition . . . 3
3) Data Filtering & Exploration . . . 3
4) Data Exploration & Figures . . . 5
5) Quality Control . . . 7

Main Software Tool Description **7**

Code Section 2: Main Analysis **8**

1) Sequence Alignment . . . 8
2) Distance Matrix . . . 8
3) K-means Clustering . . . 9
4) Validation with Silhoutte Index . . . 12

Results and Discussion **13**

Reflection **13**

References **14**

Introduction

Clustering is a main task performed in unsupervised machine learning research, wielding substantial influence in the field of bioinformatics. It not only unravels intricate functional and structural relationships among genes but also provides valuable insights into deciphering their roles in biological processes. First of all, clustering patterns aid in annotating gene functions by identifying groups with similar sequences. These similarities often signify shared functions or biological involvements(Overbeek, Fonstein, D'Souza, Pusch, & Maltsev, 1999). Furthermore, the clustering of genes unveils a deeper layer of knowledge of their evolutionary relationships. Genes that cluster together may trace back to a common evolutionary ancestor, contributing to our understanding of the evolutionary history of genes and species(Gori, Suchan, Alvarez, Goldman, & Dessimoz, 2016).

Historically, the Cytochrome Oxidase I (COI) and Cytochrome b (CYTB) mitochondrial RNA (mRNA) genes have been recognized as prominent molecular markers in the realm of bioinformatics research(Rodrigues, Morelli, & Jansen, 2017, Yacoub, Fathi, & Mahmoud, 2013). Having distinct properties in their nucleotide sequences, these genes serve as indispensable tools for the classification of genetic sequences. This exploration aims to determine not only the presence of clustering within each gene but also the potential differences in the clustering patterns of these two molecular markers. The COI and CYTB genes are anticipated to have distinct optimal clustering numbers, reflective of their inherent biological differences.

The focal point of this study remains on the taxonomic family Salmoninae chosen for its ecological significance in both terrestrial and aquatic ecosystems(Walsh et al., 2020). The primary objective of this study is to investigate if mRNA genes COI and CYTB exhibit different clustering patterns within the same taxonomic group Salmoninae. The research methodologies commenced with exploratory data analysis, followed by sequence alignment, and continued with K-means clustering. The robustness of the clustering strength is validated using the Silhouette Index. A thorough discussion of results ensues to address limitations and pave the way for future avenues of research.

Description of Dataset

In this bioinformatics project, the focal point is the analysis of mRNA gene sequences, specifically COI and CYTB, from the Salmoninae family. The dataset was extracted from the public NCBI Nuccore database on December 10, 2023. Leveraging functions in the "rentrez package", data was downloaded directly into the R environment with specified keywords. For an in-depth understanding of the data retrieval process, detailed information is available in the "Entrez_Functions.R" file. The unfiltered dataset contains 4748 observations and 2 variables: "Title" and "Sequence". "Title" encapsulates specimen specifics such as identifier, genus, and mRNA gene type (COI vs. CYTB) while "Sequence" contains mRNA sequences in FASTA format. For this analysis, the key variables of interest are the mRNA gene type and sequence. This analysis aims to unveil clustering patterns inherent in these genetic sequences within the Salmoninae family.

Github Link

https://github.com/raxhelli/BINF6210_Assignment5.git

Code Section 1: Exploratory Data Analysis (EDA)

1) Loading Library

```r
# Loading library
library(Biostrings)
library(muscle)
library(ape)
library(ggplot2)
library(rentrez)
library(tidyverse)
library(gridExtra)
library(factoextra)
library(cluster)
```

2) Data Acquisition

```r
# Download data from NCBI using helper functions in "Entrez_Functions.R"
source("Entrez_Functions.R")
FetchFastaFiles(searchTerm = "Salmoninae[ORGN] AND (COI[Gene] OR CytB[Gene])",
                seqsPerFile = 100, fastaFileName = "Salmon_fetch.fasta")

# Merge all fasta files and store it as a data frame
dfNCBI <- MergeFastaFiles(filePattern = "Salmon_fetch.fasta*")
```

3) Data Filtering & Exploration

```r
dfSalmon <- dfNCBI %>%
  # filter out samples with missing sequence
  filter(!is.na(Sequence)) %>%
  # filter out sequences with more than 5% N to avoid interference with alignment
  filter(str_count(Sequence, "N") <= (0.05 * str_count(Sequence))) %>%
  # count the number of characters in each sequence, store in Sequence_Length variable
  mutate(Sequence_Length = nchar(Sequence)) %>%

  # extract the words "complete" and "partial" to
  mutate(Sequence_Type = word(str_extract(Title, "(?i)complete|partial"))) %>%
  # extract gene name into a new column
  mutate(Gene_Name = word(str_extract(Title, "(?i)cytb|COI"))) %>%

  # convert all letter to upper case
  mutate(Gene_Name = str_to_upper(Gene_Name)) %>%

  # filter out complete genomes
  filter(Sequence_Type == "partial") %>%
  # filter out samples of COI or CYTB genes
  filter(Gene_Name == "COI" | Gene_Name == "CYTB") %>%
```

```
  # extract column 2 within in the Title column, store in Genus_Name variable
  mutate(Genus_Name = word(Title, 2L)) %>%
  # filter selected genera
  filter(Genus_Name == "Brachymystax" |Genus_Name == "Hucho" |
         Genus_Name == "Oncorhynchus" | Genus_Name == "Parahucho" |
         Genus_Name == "Salmo" | Genus_Name == "Salvelinus" |
         Genus_Name == "Salvethymus") %>%

  # create a new variable called id
  mutate(id = row_number()) %>%
  # rearrange columns
  select("id", "Title", "Genus_Name", "Gene_Name", "Sequence", "Sequence_Length")

# Check if all the complete sequences have been deleted and the columns in Gene_Name
# are properly filtered
sum(dfSalmon$Sequence_Type == "(?i)complete")
```

```
## [1] 0
```

```
table(dfSalmon$Gene_Name)
```

```
##
##  COI CYTB
## 1947 1944
```

```
# Summary statistics for COI gene sequence length before quality control
dfSalmon_COI <- dfSalmon %>%
  filter(Gene_Name == "COI")
summary(dfSalmon_COI$Sequence_Length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    51.0   648.0   652.0   643.7   652.0  1548.0
```

```
# Summary statistics for COI gene sequence length before quality control
dfSalmon_CYTB <- dfSalmon %>%
  filter(Gene_Name == "CYTB")
summary(dfSalmon_CYTB$Sequence_Length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    53.0   557.0  1015.0   834.9  1131.2  1599.0
```

4) Data Exploration & Figures

```
dfSalmon_COI %>% ggplot(aes(x = Sequence_Length)) +
  geom_histogram(bins = 10, color = "#000000", fill = "mediumpurple") +
  ggtitle("Distribution of COI Gene Sequence Length in Salmoninae Samples") +
  xlab('Sequence Length (Nucleotides)') +
  ylab('Frequency') +
  # solid vertical lines represent the median
  geom_vline(aes(xintercept = median(Sequence_Length)),
             color = "#000000", linewidth = 1) +
    # dashed lines represent the 1st & 3rd quartiles
  geom_vline(aes(xintercept = quantile(Sequence_Length, probs = c(0.25))),
             color = "orange", linewidth = 1, linetype = "dashed") +
  geom_vline(aes(xintercept = quantile(Sequence_Length, probs = c(0.75))),
             color = "orange", linewidth = 1, linetype = "dashed") +
  theme(plot.title=element_text(size = 14))
```

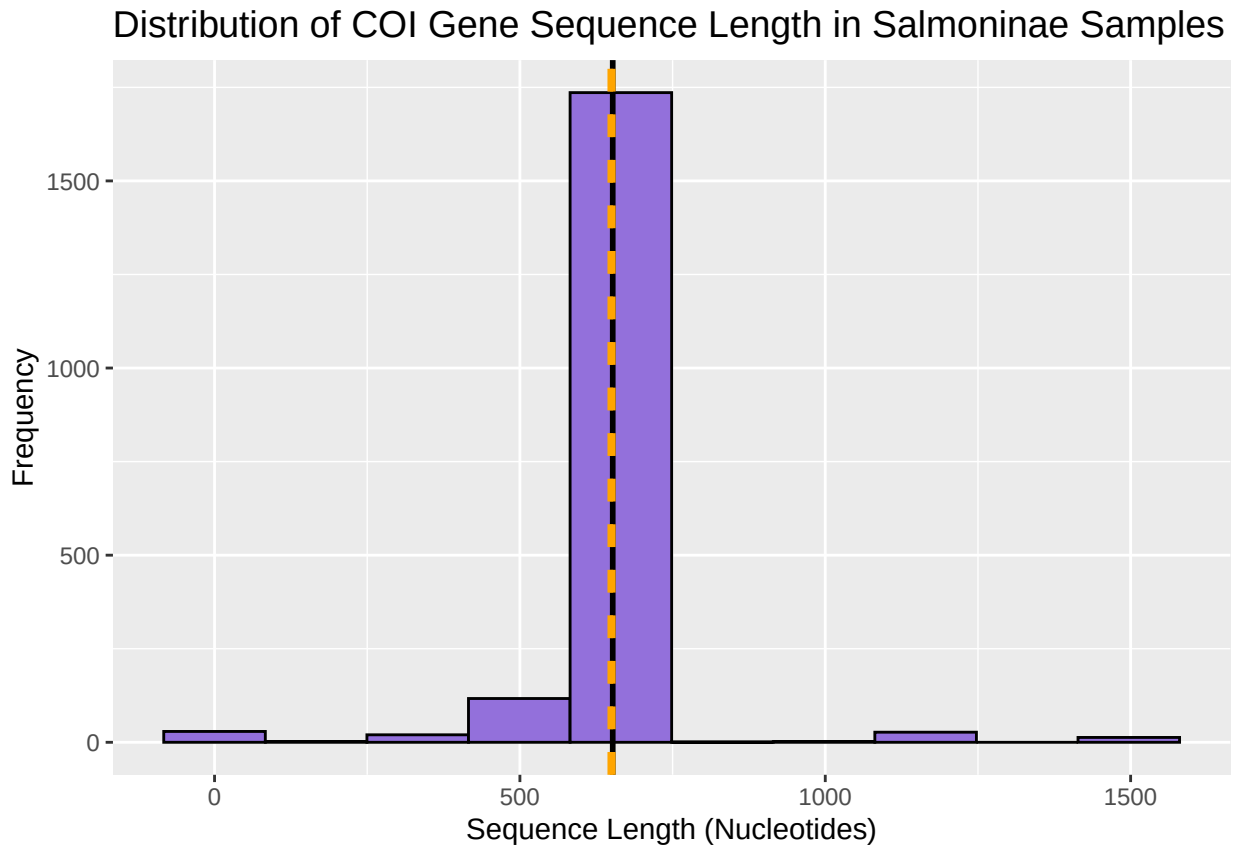

Figure 1: **Histogram Illustrating the Distribution of COI Gene Sequence Length in Salmoninae Samples**. The x-axis divides mRNA sequence lengths into 10 equal bins, ranging approximately from 0 to 1500 nucleotides. The y-axis illustrates frequency. Each bar represents the number of samples falling within a specific bin. The solid black vertical line denotes the median value of 652 nucleotides, and the two orange dashed lines signify the 1st (q1) and 3rd (q3) quartiles. The clustering of the three lines results from the proximity of their values. Notably, the majority of COI gene sequences span 648 to 652 nucleotides in length. (n = 1947)

```r
dfSalmon_CYTB %>% ggplot(aes(x = Sequence_Length)) +
  geom_histogram(bins = 10, color = "#000000", fill = "mediumpurple") +
  ggtitle("Distribution of CYTB Gene Sequence Length in Salmoninae Samples") +
  xlab('Sequence Length (Nucleotides)') +
  ylab('Frequency') +
  # solid lines represent the median
  geom_vline(aes(xintercept = median(Sequence_Length)),
             color = "#000000", linewidth = 1) +
  # dashed lines represent the 1st & 3rd quartiles
  geom_vline(aes(xintercept = quantile(Sequence_Length, probs = c(0.25))),
             color = "orange", linewidth = 1, linetype = "dashed") +
  geom_vline(aes(xintercept = quantile(Sequence_Length, probs = c(0.75))),
             color = "orange", linewidth = 1, linetype = "dashed") +
  theme(plot.title=element_text(size = 14))
```

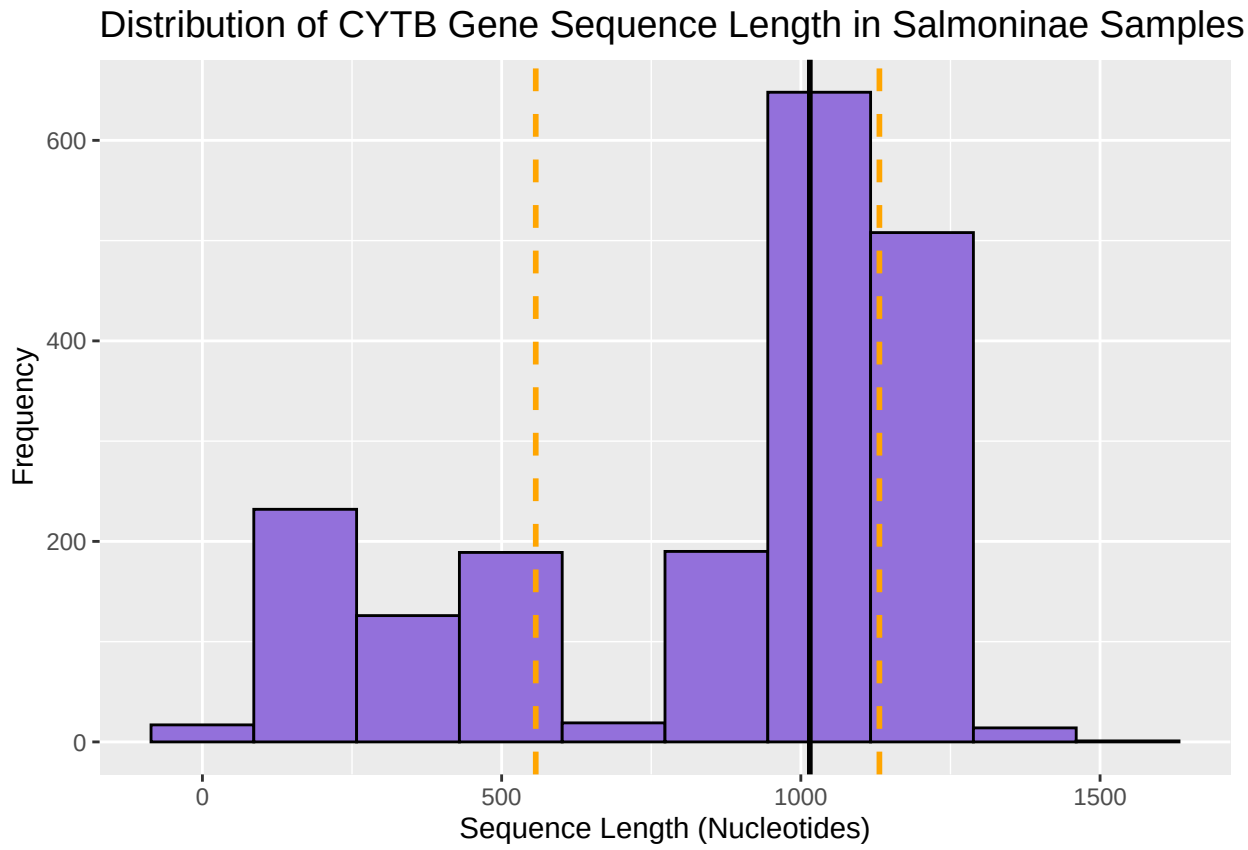

Figure 2: **Histogram Illustrating the Distribution of CYTB Gene Sequence Length in Salmoninae Samples**. The x-axis divides mRNA sequence lengths into 10 equal bins, ranging approximately from 0 to 1500 nucleotides. The y-axis illustrates frequency. Each bar represents the number of samples falling within a specific bin. The solid black vertical line denotes the median value of 1018 nucleotides, and the two orange dashed lines signify the 1st (q1) and 3rd (q3) quartiles. Notably, the majority of CYTB gene sequences span 866 to 1141 nucleotides in length. However, a substantial portion of shorter sequences spanning approximately 50 to 550 nucleotides also contribute to the overall distribution. (n = 1944)

5) Quality Control

```
# Filter out CYTB sequences shorter than 500 nucleotide due to poor sequence quality
dfSalmon_CYTB <- dfSalmon_CYTB %>%
  filter(str_count(Sequence) >= 500)

# Calculate the 1st and 3rd quartiles of COI sequences
q1_COI <- quantile(nchar(dfSalmon_COI$Sequence), probs = 0.25, na.rm = TRUE)
q3_COI <- quantile(nchar(dfSalmon_COI$Sequence), probs = 0.75, na.rm = TRUE)

# Calculate the 1st and 3rd quartiles of CYTB sequences
q1_CYTB <- quantile(nchar(dfSalmon_CYTB$Sequence), probs = 0.25, na.rm = TRUE)
q3_CYTB <- quantile(nchar(dfSalmon_CYTB$Sequence), probs = 0.75, na.rm = TRUE)

# Filter out sequences outside the interquartile range
dfSalmon_COI <- dfSalmon_COI %>%
  filter(str_count(Sequence) >= q1_COI & str_count(Sequence) <= q3_COI)
dfSalmon_CYTB <- dfSalmon_CYTB %>%
  filter(str_count(Sequence) >= q1_CYTB & str_count(Sequence) <= q3_CYTB)
```

```
# Randomly sample 100 observations for each gene type for computational efficiency
dfSalmon_COI <- dfSalmon_COI[sample(nrow(dfSalmon_COI), 100), ]
summary(dfSalmon_COI$Sequence_Length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   648.0   652.0   652.0   651.8   652.0   652.0
```

```
dfSalmon_CYTB <- dfSalmon_CYTB[sample(nrow(dfSalmon_CYTB), 100), ]
summary(dfSalmon_CYTB$Sequence_Length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     973    1015    1016    1049    1122    1140
```

Main Software Tool Description

In this investigation, the Bioconductor R packages “muscle” and “cluster” played essential roles in our analytical framework. Muscle, renowned for its power in multiple sequence alignment, serves as a robust tool for delineating genetic relationships among mRNA gene sequences within the Salmoninae family(Edgar, 2004). Its strength lies in its adaptability to divergent sequences, ensuring reliable alignments, although computational intensity poses challenges with larger datasets. In addition, the cluster package is instrumental in executing K-means clustering and validating clustering strength through the Silhouette Index(Maechler et al., 2023). Its user-friendly implementation and efficiency are notable strengths, though sensitivity to initial center choices and outliers. The synergistic use of other R packages such as ggplot2, Biostrings, and rentrez has further streamlined and enriched our analytical process. These packages were chosen for their established utility, and while alternatives were explored, this combination emerged as a well-suited suite for our objectives.

Code Section 2: Main Analysis

1) Sequence Alignment

```
# Change Sequence in dfSalmon_COI and dfSalmon_CYTB to DNAStringSet class
# Perform sequence alignment to calculate distance matrix in step 2)

## COI Gene
dfSalmon_COI$Sequence <- DNAStringSet(dfSalmon_COI$Sequence)
dfCOI_alignment <- DNAStringSet(muscle::muscle(dfSalmon_COI$Sequence))

## CYTB Gene
dfSalmon_CYTB$Sequence <- DNAStringSet(dfSalmon_CYTB$Sequence)
dfCYTB_alignment <- DNAStringSet(muscle::muscle(dfSalmon_CYTB$Sequence))
```

2) Distance Matrix

```
# Calculate distance matrix for COI using model "TN93" for its ability to
# distinguish between transitions and transversions.
# Check for missing elements in the matrix; if missing elements are found,
# perform imputation based on the mean SDD value.

## COI Gene
dnaBin_COI <- as.DNAbin(dfCOI_alignment)
distanceMatrix_COI <- dist.dna(dnaBin_COI, model = "TN93", as.matrix = TRUE,
                               pairwise.deletion = TRUE)
if (any(is.na(distanceMatrix_COI))) {
  # Calculate the mean distance excluding missing values
  mean_distance <- mean(distanceMatrix_CYTB, na.rm = TRUE)
  # Impute missing values with the mean
  distanceMatrix_CYTB <- ifelse(is.na(distanceMatrix_CYTB), mean_distance, distanceMatrix_CYTB)
  # Check again for missing values after imputation
  table(is.na(distanceMatrix_CYTB))
}

## CYTB Gene
dnaBin_CYTB <- as.DNAbin(dfCYTB_alignment)
distanceMatrix_CYTB <- dist.dna(dnaBin_CYTB, model = "TN93", as.matrix = TRUE,
                                pairwise.deletion = TRUE)
if (any(is.na(distanceMatrix_CYTB))) {
  mean_distance <- mean(distanceMatrix_CYTB, na.rm = TRUE)
  distanceMatrix_CYTB <- ifelse(is.na(distanceMatrix_CYTB), mean_distance, distanceMatrix_CYTB)
  table(is.na(distanceMatrix_CYTB))
}
```

3) K-means Clustering

```r
# Calculate the Sum of Squared Distance (SSD) value between each point and the
# centroid within a cluster for each value of k(1:10)
# Create a ggplot to visualize the results

## COI Gene
ssd_COI <- sapply(1:10, function(k) {
  COI_kmeans <- kmeans(distanceMatrix_COI, centers = k, nstart = 10)
  return(COI_kmeans$tot.withinss)
})
plot_data_1 <- data.frame(k = 1:10, SSD = ssd_COI)
plot1 <- ggplot(plot_data_1, aes(x = k, y = SSD)) +
  geom_line() +
  geom_point() +
  ggtitle("Number of Clusters for COI Genes") +
  xlab("Number of Clusters (k)") +
  ylab("Sum of Squared Distances (SSD)") +
  theme_minimal()+
  theme(plot.title=element_text(size = 10))

## CYTB Gene
ssd_CYTB <- sapply(1:10, function(k) {
  CYTB_kmeans <- kmeans(distanceMatrix_CYTB, centers = k, nstart = 10)
  return(CYTB_kmeans$tot.withinss)
})
plot_data_2 <- data.frame(k = 1:10, SSD = ssd_CYTB)
plot2 <- ggplot(plot_data_2, aes(x = k, y = SSD)) +
  geom_line() +
  geom_point() +
  ggtitle("Number of Clusters for CYTB Genes") +
  xlab("Number of Clusters (k)") +
  ylab("Sum of Squared Distances (SSD)") +
  theme_minimal() +
  theme(plot.title=element_text(size = 10))
```

```
grid.arrange(plot1, plot2, nrow = 2)
```

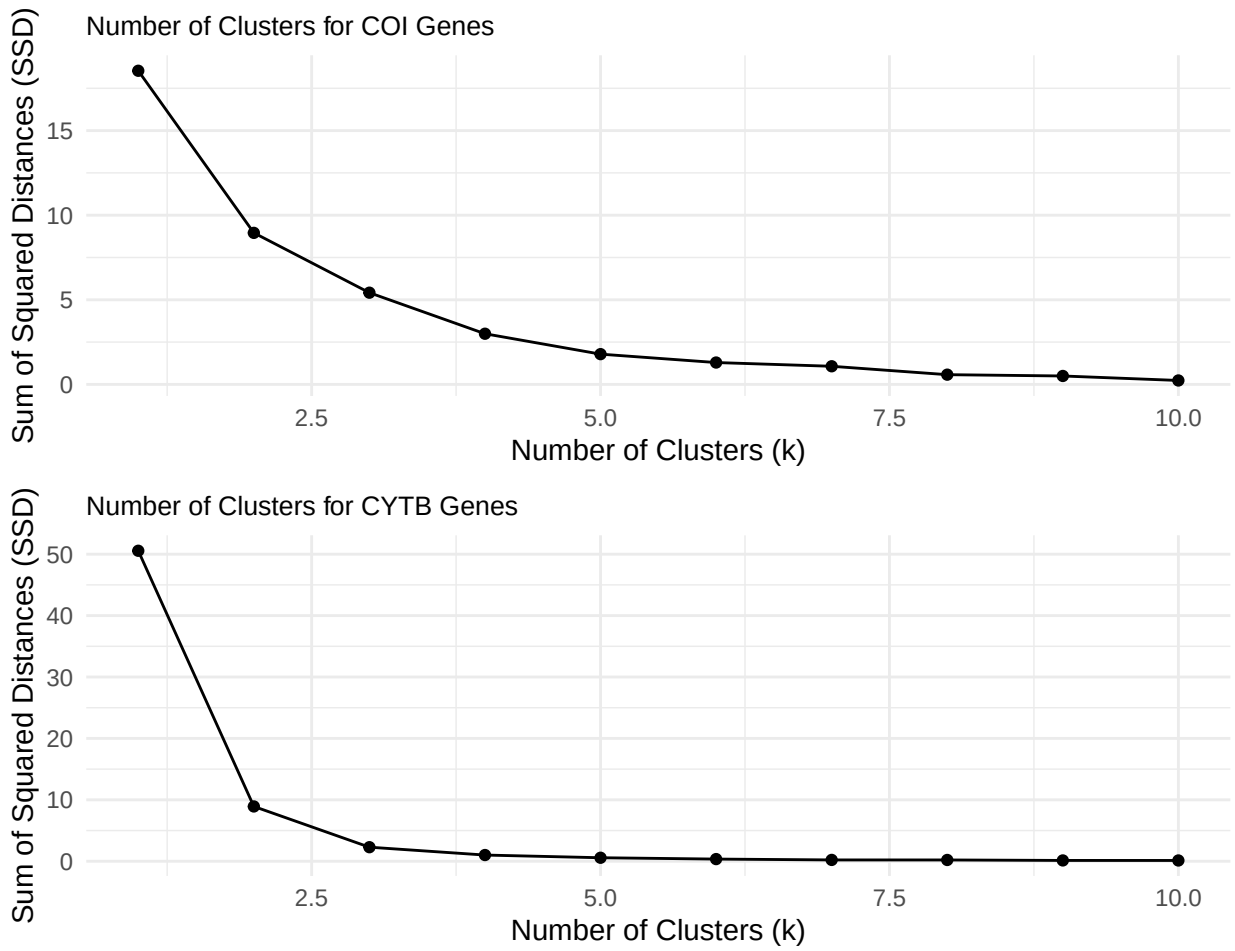

Figure 3: **Elbow Method Graph depicting the Optimal Cluster Value (K) Determination for COI and CYTB Genes in Salmoninae Samples**. The top panel illustrates the results for COI genes, while the bottom panel corresponds to CYTB genes. The x-axis denotes the number of clusters (k), and the y-axis represents the Sum of Squared Distance (SSD) values between each point and the centroid within a cluster. Notably, for both mRNA genes, the graph exhibits a pronounced change in value at k = 2, forming a distinct elbow shape. Consequently, 2 emerges as the optimal number of clusters for both mRNA genes, and k = 2 is consequently utilized for subsequent analyses (n = 100).

```r
# Perform K-means clustering for genes based on optimal number of clusters (2)
## COI Gene
cluster_COI <- kmeans(distanceMatrix_COI, centers = 2)

## CYTB Gene
cluster_CYTB <- kmeans(distanceMatrix_CYTB, centers = 2)
```

```r
# Visualize the clustering results
my_palette <- c("orange", "mediumpurple")

## COI Gene
plot3 <- fviz_cluster(cluster_COI, data = distanceMatrix_COI, geom = "point",
             ellipse.type = "convex", palette = my_palette,
             ggtheme = theme_minimal(), main = "K-means Clustering for COI (k = 2)")
plot3 <- plot3 + theme(plot.title = element_text(size = 12))

## CYTB Gene
plot4 <- fviz_cluster(cluster_CYTB, data = distanceMatrix_CYTB, geom = "point",
             ellipse.type = "convex", palette = my_palette,
             ggtheme = theme_minimal(), main = "K-means Clustering for CYTB (k = 2)")
plot4 <- plot4 + theme(plot.title = element_text(size = 12))

grid.arrange(plot3, plot4, ncol = 2)
```

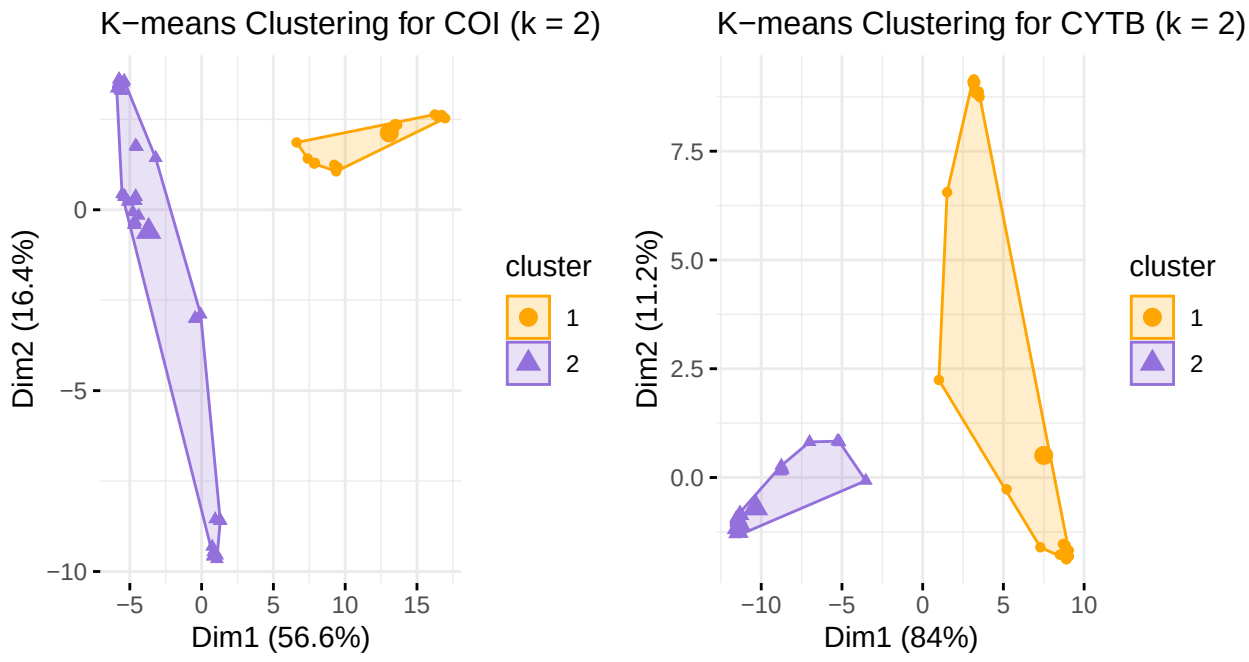

Figure 4: **Visualization of Clustering Patterns in COI and CYTB Genes within Salmoninae Samples**. The left panel corresponds to COI genes, while the right panel illustrates CYTB genes. At the chosen cluster value (k = 2), a distinct separation between clusters is evident. Notably, cluster 2 in the COI gene and cluster 1 in the CYTB genes display widely dispersed points, suggesting the potential for further subdivision into smaller clusters to capture underlying structure and enhance the resolution of the clustering analysis (n = 100).

4) Validation with Silhoutte Index

```
# Calculate Silhouette Index to evaluate the clustering strength in step 3

## COI Gene
SilhouetteIdx_COI <- silhouette(cluster_COI$cluster, dist(distanceMatrix_COI))

## CYTB Gene
SilhouetteIdx_CYTB <- silhouette(cluster_CYTB$cluster, dist(distanceMatrix_CYTB))

par(mfrow=c(2,1))
plot(SilhouetteIdx_COI, main = "Silhouette Plot for COI Genes",
     col = c("orange", "mediumpurple"))
plot(SilhouetteIdx_CYTB, main = "Silhouette Plot for CYTB Genes",
     col = c("orange", "mediumpurple"))
```

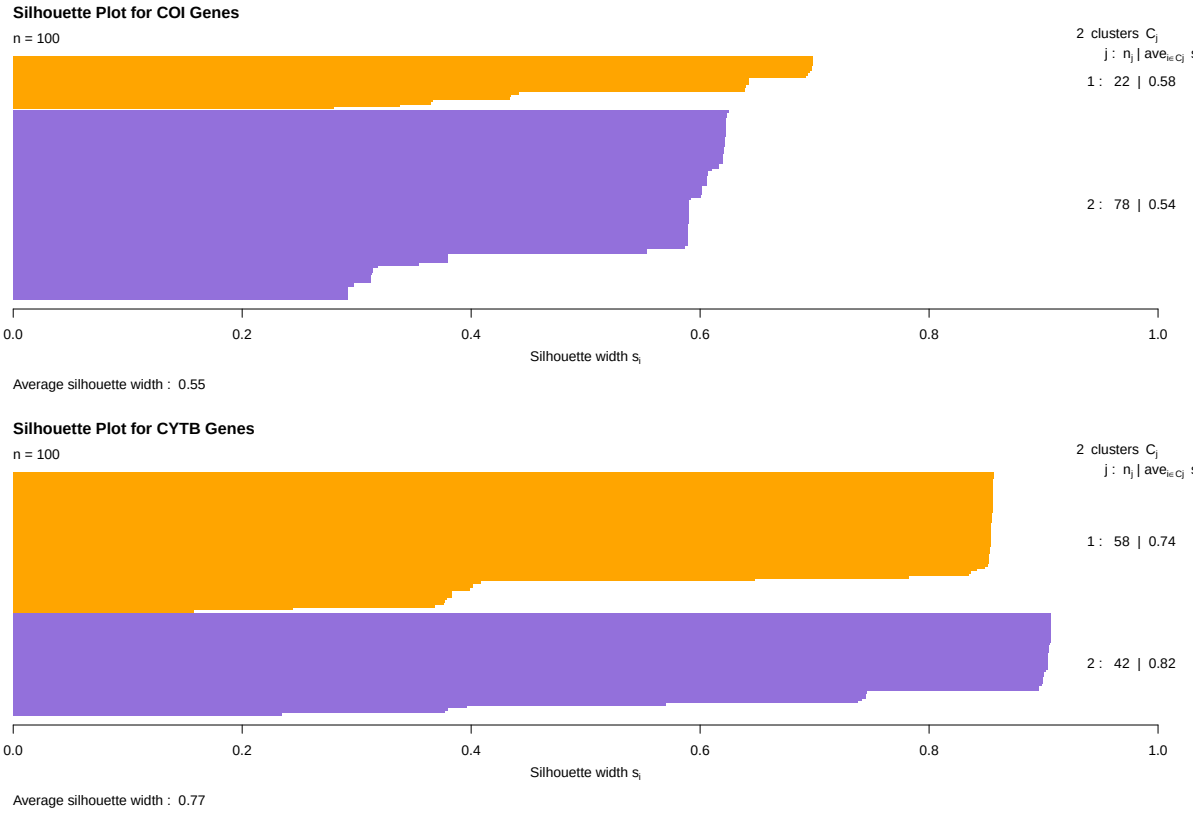

Figure 5: **Silhouette Plots assessing the Clustering Strength for COI and CYTB Genes in Salmoninae Samples**. The top panel corresponds to COI genes, while the bottom panel illustrates CYTB genes. The silhouette plots provide an evaluation metric: when the average silhouette coefficient approaches 1, it indicates robust matching of data points to their respective clusters and poor alignment with neighboring clusters. Conversely, an average silhouette coefficient near 0 or negative values suggests potential inadequacies in the clustering. For COI genes, the average silhouette value is 0.55, indicating a moderate match, and implying the potential need for further clustering refinement. In contrast, CYTB genes exhibit a higher average silhouette value of 0.77, indicating well-defined clusters and suggesting a robust clustering structure for the data points (n = 100).

Results and Discussion

In response to our primary objective, our exploration into mRNA gene clustering within the Salmoninae family revealed the optimal number of clusters to be 2 through K-means clustering for both COI and CYTB genes. The Elbow Method Graphs capture this optimal clustering point, demonstrated by a distinct change in the trend line at k = 2 (Figure 4). Clustering plots (Figure 5) further accentuate this finding, portraying a clear separation of clusters in both genes at the chosen optimal cluster value. Surprisingly, cluster 2 in COI cluster 1 in CYTB genes exhibited widely dispersed data points, challenging our initial expectations. The Silhouette Index evaluation metrics state that 1 indicating robust match of data to their respective clusters and 0 or negative values suggest inadequate clustering(Shutaywi & Kachouie, 2021). The evaluations indicated a moderate match for COI genes (average Silhouette value of 0.55) and a well-defined clustering structure for CYTB genes (average Silhouette value of 0.77), adding an unexpected layer to our results.

A notable caveat in our study pertains to the limited sample size, driven by computational constraints. Despite obtaining 3891 sequence samples, the computational capacity allowed only 100 sequences from each gene during sequence alignment. This limitation could potentially impact the robustness of our conclusions. Additionally, the absence of a published clustering pattern for COI and CYTB genes in the Salmoninae family poses a challenge, introducing uncertainty regarding the validity of our findings.

In the realm of future research, expanding the scope of clustering analysis by testing various k values would provide a more comprehensive understanding of the clustering dynamics within COI and CYTB genes. Additionally, conducting the same analysis on a well-sequenced reference genome would serve to validate the credibility of our workflow. These steps would not only refine our clustering patterns but also contribute to the broader understanding of mRNA gene clustering dynamics in the context of evolutionary biology.

Reflection

Reflecting on the journey through the five assignments in BINF6210, this course has been an enlightening experience, offering insights that extend beyond the confines of classroom materials. One significant revelation was the recognition that a thorough bioinformatics project necessitates extensive independent research. This includes delving into new packages, exploring vignettes, engaging with online tutorials, and referencing previous publications to augment the depth of understanding. A critical aspect of my learning process was mastering the art of managing multiple projects concurrently. This involved honing the skill of prioritization and allocating time to each project accordingly. The understanding of the importance of peer collaboration also became evident. Constructive feedback from peers proved instrumental in improving the quality of our individual work. I am committed to carrying these insights into my future coursework and career. Project management skills will remain a focal point of my development, alongside a dedicated effort to enhance my software skills in the realm of bioinformatics. These experiences serve as stepping stones toward my aspiration of becoming a proficient bioinformatician, contributing meaningfully to the field.

References

Scientific Literature:

1. Gori, K., Suchan, T., Alvarez, N., Goldman, N., & Dessimoz, C. (2016). Clustering genes of common evolutionary history. Molecular Biology and Evolution, 33(6), 1590–1605. doi:10.1093/molbev/msw038

2. Overbeek, R., Fonstein, M., D'Souza, M., Pusch, G. D., & Maltsev, N. (1999). The use of gene clusters to infer functional coupling. Proceedings of the National Academy of Sciences, 96(6), 2896–2901. doi:10.1073/pnas.96.6.2896

3. Rodrigues, M. S., Morelli, K. A., & Jansen, A. M. (2017). Cytochrome c oxidase subunit 1 gene as a DNA barcode for discriminating trypanosoma cruzi dtus and closely related species. Parasites & Vectors, 10(1). doi:10.1186/s13071-017-2457-1

4. Shutaywi, M., & Kachouie, N. N. (2021). Silhouette analysis for performance evaluation in machine learning with applications to clustering. Entropy, 23(6), 759. doi:10.3390/e23060759

5. Walsh, J. C., Pendray, J. E., Godwin, S. C., Artelle, K. A., Kindsvater, H. K., Field, R. D., ... Reynolds, J. D. (2020). Relationships between Pacific salmon and aquatic and terrestrial ecosystems: Implications for ecosystem-based management. Ecology, 101(9). doi:10.1002/ecy.3060

6. Yacoub, H. A., Fathi, M. M., & Mahmoud, W. M. (2013). DNA barcode through cytochromebgene information of mtdna in native chicken strains. Mitochondrial DNA, 24(5), 528–537. doi:10.3109/19401736.2013.770489

R Packages:

1. Auguie B (2017). *gridExtra: Miscellaneous Functions for "Grid" Graphics*. R package version 2.3, https://CRAN.R-project.org/package=gridExtra.

2. Edgar, R.C. (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Res 32:1792-1797. doi:10.1093/nar/gkh340

3. H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.

4. Kassambara A, Mundt F (2020). *factoextra: Extract and Visualize the Results of Multivariate Data Analyses*. R package version 1.0.7, https://CRAN.R-project.org/package=factoextra.

5. Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M., Hornik, K.(2023). cluster: Cluster Analysis Basics and Extensions. R package version 2.1.6.

6. Pagès H, Aboyoun P, Gentleman R, DebRoy S (2023). *Biostrings: Efficient manipulation of biological strings*. doi:10.18129/B9.bioc.Biostrings https://doi.org/10.18129/B9.bioc.Biostrings, R package version 2.68.1, https://bioconductor.org/packages/Biostrings.

7. Paradis E, Schliep K (2019). "ape 5.0: an environment for modern phylogenetics and evolutionary analyses in R." *Bioinformatics*, *35*, 526-528. doi:10.1093/bioinformatics/bty633 https://doi.org/10.1093/bioinformatics/bty633.

8. R Core Team (2023). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.

9. Wickham H, Averick M, Bryan J, Chang W, McGowan LD, François R, Grolemund G, Hayes A, Henry L, Hester J, Kuhn M, Pedersen TL, Miller E, Bache SM, Müller K, Ooms J, Robinson D, Seidel DP, Spinu V, Takahashi K, Vaughan D, Wilke C, Woo K, Yutani H (2019). "Welcome to the tidyverse." *Journal of Open Source Software*, *4*(43), 1686. doi:10.21105/joss.01686 https://doi.org/10.21105/joss.01686.

10. Winter, D. J. (2017) rentrez: an R package for the NCBI eUtils API The R Journal 9(2):520-526

11. Wright ES (2016). "Using DECIPHER v2.0 to Analyze Big Biological Sequence Data in R." *The R Journal*, *8*(1), 352-359.