STAT 489: Final Project Report

Owen Holliday, Corey Voyles, Clara Cherotich,
Meghan Ryan, Riwaz Poudel, and Lydia Morningstar

December 12, 2024

TABLE OF CONTENTS

LIST OF FIGURES

Figure Page

LIST OF TABLES

Table Page

CHAPTER 1

Part 1: Cluster Analysis

1.1 Introduction

Cluster Analysis is the method of grouping a set of observations with similar characteristics for analytic purposes. This can be used in many fields including sales. We will use cluster analysis to analyze a dataset of 200 mall customers. The dataset includes the following variables: Age, Gender, Annual Income, and Spending Score. *(Appendix A: Dataset Detail)*

1.2 Data Exploration

Data exploration was done on each of the variables in the dataset to understand the range of the dataset and possibly find some patterns. *Table 1 in Appendix A* is used to show the range of the quantitative predictors in the dataset. It was also found that there were 112 Females in the dataset and 88 Males in the dataset meaning there is an almost even split between males and females. Future predictions or cluster assigning may be inaccurate for those much older than 70 or those with an income much higher than $137,000 as those values are not observed in the dataset. From this histogram of Age shown in *Figure 2 in Appendix A*, we can see that ages 50+ seem to be observed less frequently than ages 20-50. From the histogram of Annual Income in *Figure 3 in Appendix A*, we observed most observations with an income less than $80,000 and very few customers with an income above $100,000. From the histogram of Spending Score *Figure 4 in Appendix A, i*t appears that there are three distinct groups, people with low spending scores, moderate spending scores, and high spending scores. Boxplots between quantitative variables and gender in *Figure 9-11 in Appendix A* show an equal distribution of males and females in age, spending score, and annual income. No significant correlation between the two variables was observed in the correlation matrix in *Table 5 in Appendix A.*

1.2.1 Exploring Relationships

We explored relationships between variables before clustering to find any distinct patterns in the data. First, a scatterplot was made of every variable to see if any plots stood out which is given in *Figure 6 in Appendix A*. From this graph, two scatterplots stood out, Annual Income vs. Spending Score and Age vs. Spending Score. These scatterplots were reproduced and are given in *Figure 7 and 8 in Appendix A*. From these scatterplots, observations with an annual income between $40,000 and $70,000 have a spending score between 40 and 60. It can also be shown that customers observed who were above the age of 40 did not have a spending score above 60. Each graph seems to have 4-5 clusters of data clamped up together showcasing a diverse spending pattern among age and income.

1.3 Clustering Analysis

1.3.1 K-Means Clustering

Based on the elbow method shown in *Figure 9 in Appendix B*, we selected k = 4 for our analysis as it represents the most optimal point on the graph. 4-means clustering was performed on the entire dataset 25 times, each time with a different random initialization of the cluster center. The resulting cluster, which has the lowest within-cluster sum of squares among 25 runs, visualization is displayed in *Figure 1 of Appendix B*. However, since the dataset contains more than two variables, the cluster diagram does not provide clear insights, as the clusters overlap significantly in two dimensions. To gain a deeper understanding, scatterplots were utilized to explore relationships between quantitative variables within each cluster, while bar graphs were employed to examine the gender distribution across clusters, as shown in *Figures 2-5 in Appendix B*. Additionally, From *Figure 6-8 in Appendix B,* box plots between age and other quantitative variables for each cluster are created to showcase relationships between gender and other quantitative variables.

1.3.2 Hierarchical Clustering

Since our dataset includes a categorical variable, we utilized Gower distance to perform hierarchical clustering. *Figures 1-3 in Appendix-C* illustrate the dendrograms for hierarchical clustering using Single Linkage, Average Linkage, and Complete Linkage methods. Upon reviewing the results, both Average Linkage and Complete Linkage produced distinct and similar clustering patterns. For further analysis, we selected four clusters with the Average Linkage method.

The decision to use four clusters was taken by two factors. First, the dendrogram separates four distinct clusters with minimal overlap, making this choice visually supported. Second, the elbow method applied in k-means clustering also identified four as the optimal number of clusters, supporting our selection.

Figure 4-6 in Appendix-C represents the scatterplots showcasing the relationship between all the quantitative variables among clusters. *Figure 7 in Appendix-C* shows the gender distribution in each cluster. Additionally, the box plot between age and categorical variables is represented from *Figure 8-10 in Appendix-C.*

In our analysis, the clustering results from K-means and hierarchical clustering revealed distinct patterns in how data was grouped. K-means produced balanced clusters with a proportional representation of variables, while hierarchical clustering, using Gower distance, distinctly separated clusters by gender, reflecting the categorical structure of the dataset. These differences also influenced the visual patterns in scatter plots and box plots, where K-means excelled in quantitative variable relationships, and hierarchical clustering highlighted categorical distinctions. For a detailed data-driven discussion, refer to **Appendix C: Comparison between Hierarchical and K-means clustering details**.

1.4 Post-Clustering Analysis

1.4.1 Intra Cluster Analysis

Cluster	**Age**	**Annual Income**	**Spending Score**	**Gender Distribution**
1	varying age	Average to high annual income	very low to average spending score	Slightly more males than females
2	above average to high age	average to low annual income	average to below average spending score	Moderately more females than males
3	low to average age	low to average annual income	average to high spending score	Moderately more females than males
4	below average age	above average to high annual income	high spending score	Slightly more females than males

Table 1.1: Characteristics of each cluster using k-means clustering with 4 clusters

1.4.2 Inter Cluster Analysis

Figure 1-4 in Appendix D presents box plots for age, annual income, and spending score across four clusters, along with a bar graph showing gender distribution. Among these variables, spending score is the most distinctly clustered, with Cluster 1 having the lowest scores and Cluster 4 the highest, indicating a strong separation based on spending behavior. Age shows some clustering, with three clusters being somewhat distinct, although one overlaps with two others. Annual income displays less separation, as two high-income clusters overlap significantly, while two low-income clusters are distinct from the high-income groups but overlap with each other. Gender distribution appears relatively even across the clusters, though two clusters have a higher proportion of females, likely reflecting the dataset's overall female-dominant composition. Overall, spending score emerges as the primary driver of clustering, with age contributing moderately, while income and gender show less distinct patterns.

1.5 Summary and Recommendations

To summarize, data exploration revealed that higher annual incomes often correlated with moderate spending scores, while lower incomes showed more varied spending. Age was also linked to spending, with customers above 40 rarely having high spending scores. Using the elbow method, k = 4 was chosen for k-means clustering, while hierarchical clustering leveraged Gower distance to capture categorical variables. Spending score emerged as the primary driver of clustering, with age contributing moderately and income showing overlapping patterns.

For **Cluster 1**, which includes high-income, low-spending individuals, strategies should focus on **exclusive promotions, loyalty programs, high-end stores, and significant discounts on premium products** to encourage higher spending. For **Cluster 2**, consisting of middle-aged to older customers with low income, **discounts on everyday items like groceries or annual membership deals** could help build loyalty and increase their spending habits. In this cluster, no individual above the age of 40 had a spending score above 60 so it might be beneficial for the mall to bring in **family-oriented stores that would interest the middle-aged community**. In **Cluster 3**, made up of younger, low-income customers with high spending tendencies, **introducing a mall-branded credit card** could allow these customers to spend more even with limited income. For **Cluster 4**, which features high-income, high-spending individuals, **premium products alongside personalized shopping experiences** should be provided to retain them and encourage repeated visits to maintain spending levels. Overall, the highest-spending individuals in the mall are younger individuals so more options for old individuals, especially ones with high income should also be targeted.

CHAPTER 2

Part 2: Medical Insurance Dataset Analysis

2.1 Introduction

This section aims to identify the best predictive model for estimating individual medical insurance costs using demographic and lifestyle variables. We will start with numerical and graphical exploration to understand variable relationships and correlations. The dataset includes quantitative variables like age, sex, and BMI, qualitative variables such as children, smoking status, region, and the response variable, expenses. Following exploration, various regression models will be applied, evaluated, and compared to select the most suitable model for predicting insurance costs on future unseen data.

2.2 Data Exploration

With this data, we are specifically considering 4 numerical variables, including the response: Age, BMI, Children, and Expenses. We observed each of these variables' distributions by creating four respective histograms for each variable and looking at their numerical summaries (Appendix G). From this, we can see that our BMI variable is approximately normally distributed. Whereas, the age variable seems rather random and uniform, while our children and expenses variables seem to be right skewed. With further analysis, we could apply transformations on the predictors or other methods to meet our normality assumption.

Looking at the correlations between all the numeric variables alone, they all seemed to be rather low. However, there seems to be a rather significant correlation between Smoking Status and Expenses. We observed this correlation by creating a pair plot (Appendix G). From this, we can see that smoking seems to be a significant predictor of our response. This also means that even though our other predictor variables may seem to have a poor relationship with our response, they could still be useful in combination with our Smoking

Status variable.

To observe further, we can see clear differences in groups within our Age vs. Expenses scatterplot because of the smoker variable. We see that most nonsmokers have a low to medium medical insurance cost while smokers have larger insurance costs. We can also tell that Age and Expenses seem to have a positive linear relationship. Similarly, with BMI vs. Expenses, we see that there is a clear separation with nonsmokers having lower BMIs than smokers overall. Also, we can see that individuals with larger BMIs have larger insurance costs than those with lower BMIs. Therefore, we can again see in this graph that nonsmokers seem to have lower costs than smokers, which is consistent with the previous graph.

These relationships are most prominent in our analysis and the information gathered here is important in being able to make further conclusions about our data with the model we fit. We note again that the other correlations between the other numeric and categorical variables were low. This signifies that there is little to no multicollinearity between our predictors, which is a good thing when we are using these variables to fit our models. We will now fit our five different statistical learning methods to see which would perform the best and give us the best results.

2.3 Data Preprocessing, Model Building and Evaluation Technique

Before fitting the models, we standardize quantitative predictors to ensure a consistent scale and avoid bias. Qualitative variables are also converted into factors as needed. The dataset is split into training (75%) and testing (25%) sets, with 10 fold cross-validation applied to the training set for each model. This approach ensures the model captures underlying patterns while the independent testing set evaluates its robustness. To evaluate model performance, we use both cross-validation and test set errors, focusing on metrics such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). These metrics measure the average magnitude of errors between predicted and actual values, with lower

values indicating better predictive accuracy. Initially, we assess models using cross-validation errors, comparing RMSE and MAE across different models. Later, these results are validated against errors from the test dataset. Finally, the model with the lowest RMSE and MAE on the test set will be selected as the final model for its best predictive ability.

2.4 Applying Multiple Linear Regression

We use 10-fold cross-validation on the training set to fit a baseline multiple linear regression model predicting expenses from all predictors. From the output in our code, our estimated function is rather complex, but we can still make interpretations of the variables in relation to the response. Specifically, it seems that Age, BMI, Children, and the smokers variable all seem to be highly significant predictors in this model. For example, Age has a coefficient estimate of approximately 0.283. This means that for a 1 unit increase in age, we approximately have a 0.283 increase on average in our standardized insurance cost. This would be interpreted similarly for the other variables as well. We can also see that all of these variables are positively related to the response. We observed a cross-validation RMSE of 0.5094 and MAE of 0.355

2.5 Applying Ridge Regression

The Ridge Regression method is used to reduce any complexity in our model like multi-collinearity, which is caused by highly co-related predictors and overfitting. Ridge Regression does this by imposing a penalty on the coefficients and seeks the coefficients that fit the data well by making RSS small. The final values used for the model were α=0 and λ=0.0777 and as a result, we got cross-validation RMSE and MAE of 0.5135750 and 0.3638328 respectively.

2.6 Applying Lasso Regression

Lasso regression, a newer alternative to ridge regression, improves on its limitations by shrinking certain coefficients to exactly zero, allowing for regularization and feature selection.

We used cross-validation to get our optimal tuning parameter λ. As seen in the code in Appendix K, for this model we had $\alpha = 1$ and $\lambda = .0111$. After fitting the model, we obtained a cross-validation RMSE and MAE of 0.5094412 and 0.3552108 respectively.

After tuning the model, we examined the coefficients used in the final output. Consistent with our earlier findings in linear regression, the smokeryes variable has the largest impact on insurance expenses, while sex has the smallest effect.

2.7 Applying Elastic Net

While Ridge and Lasso are similar models designed to address each other's limitations, Elastic Net serves as a hybrid approach that combines the strengths of both. After running cross-validation for choosing the optimal α and λ values, we got $\alpha = 0.1111$ and $\lambda = 0$. Finally, after running the full model (code in Appendix K) we got a cross-validation RMSE and MAE of 0.5094323 and 0.3557492 respectively.

2.8 Applying Subset Selection

With subset selection, we applied stepwise selection, a way to include or exclude variables at each step in building each sized model. This method was applied to our original multiple linear regression model to see if we could improve results by decreasing the number of predictors.

Our analysis slightly improved prediction accuracy and error. The best model includes four predictors: age, BMI, children, and smoking status. However, the original model with all six predictors remains optimal, yielding a cross-validation RMSE and MAE of 0.50896 and 0.35422 respectively.

2.9 Principle Components Regression (PCR)

Here, we split the data into six principle components for each of the six predictors. We cannot specifically make conclusions about what variables are most significant in each

principle component loading since we did *not* perform Principle Components Analysis in this case. However, we can see the percentage of the explained variance each component explains. From principle component 1 to 6 we have 28%, 53.46%, 75.89%, 82.22%, 88.47%,, and 100%, respectively.

We can see that none of the RMSE's or MAE's are nearly as good as what they were for the previous models. We can see that the lowest RMSE is 0.9406573 with an MAE of 0.7292832 resulting from using 4 principle components. When using principle component regression in general, we want to reduce the dimensionality of the data to reduce the multicollinearity. Therefore, using 4 principle components compared to 6 variables like we were originally using is not that significant of a dimensionality reduction, and thus not necessarily desired with this method.

2.10 Summary, Results, and Recommendations

From the table containing test RMSE and MAE in Appendix I, we conclude that the Lasso regression model performed the best with the lowest cross-validation error and testing error for both metrics. We used Lasso regression to build a final model using the entire dataset and obtained a cross-validation RMSE and MAE of 0.5001185 and 0.3465158 respectively. From the model, smoking is the most significant predictor of high medical insurance costs. Since most models performed similarly, it may not be ideal to discard those with close performance. Instead, further research and a deeper analysis of the data should be conducted to identify patterns and relationships. Additionally, techniques like combining or interacting variables could help improve the model's predictive accuracy and overall robustness. To conclude, this was an insightful application of the wide number of topics covered throughout this course.

APPENDIX A

Data Exploration Pre-Cluster

Dataset Detail

The dataset contains following variables:

- Age: The age of the customer.
- Gender: The gender of the customer.
- Annual Income: The annual income of the customer in thousands of dollars.
- Spending Score: A score assigned by the mall based on customers spending behavior with range (1 – 100).

Of these variables, Age, Annual Income, and Spending Score are quantitative while Gender is qualitative.

Statistic	**Age**	**Annual Income**	**Spending Score**
Min.	18	15	1
1st Qu.	38.75	41.5	34.75
Median	36	61.5	50
Mean	38.85	60.56	50.2
3rd Qu.	49	78	73
Max.	70	137	99

Table 1: Summary of the Quantitative Variables in the Mall Customers Dataset

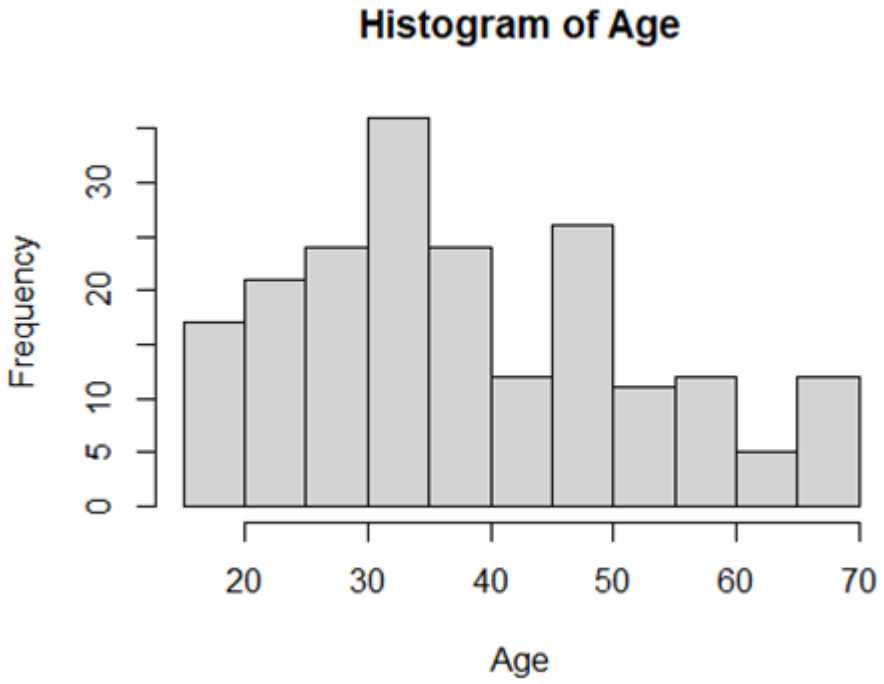


Fig. 2: Histogram of Age from the Mall Customers Dataset

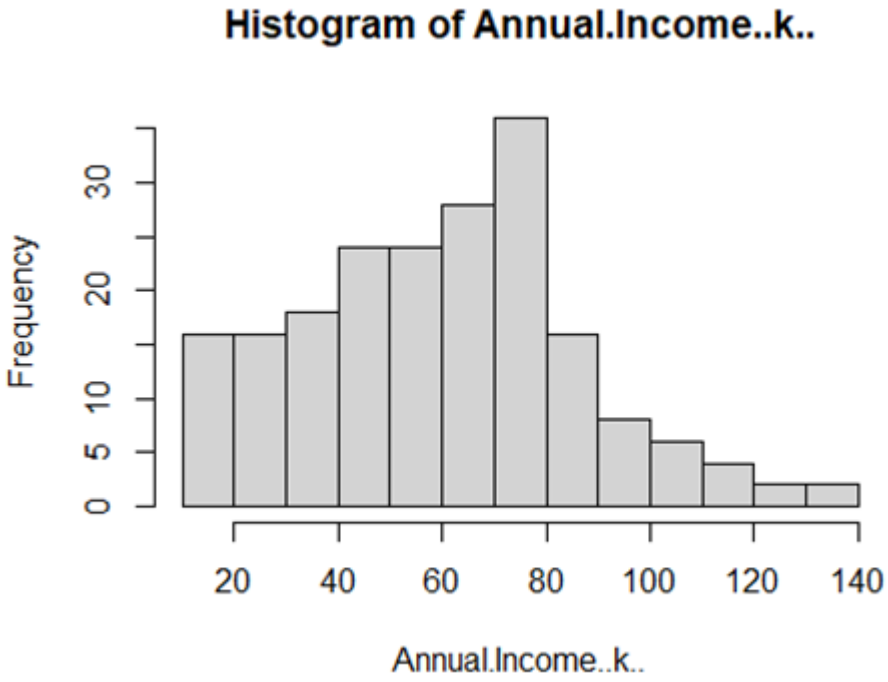


Figure 3: Histogram of Annual Income from the Mall Customers Dataset

Figure 4: Histogram of Spending Score from the Mall Customers Dataset

	Age	Annual Income	Spending Score
Age	1	-0.012398043	-0.327226846
Annual Income	-0.012398043	1	0.009902848
Spending Score	-0.327226846	0.009902848	1

Table 5: Correlation Matrix of the Mall Customers Dataset

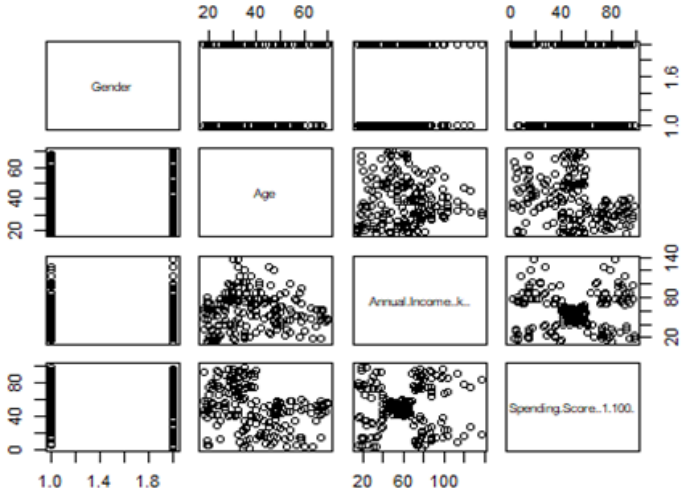


Figure 6: Scatterplot of All Variables in the Mall Customers Dataset

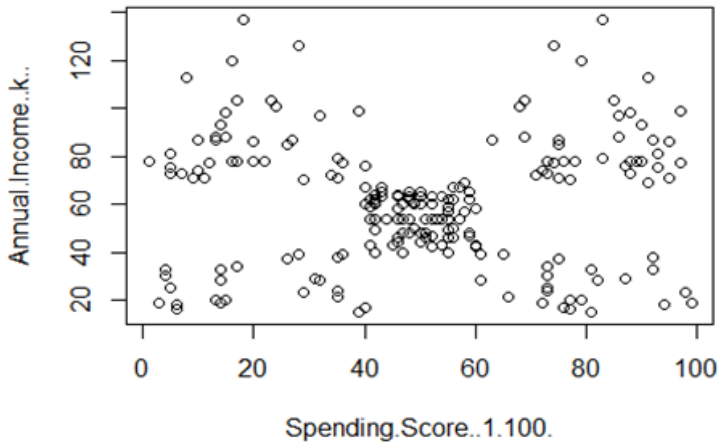


Figure 7: Scatterplot of Annual Income vs Spending Score

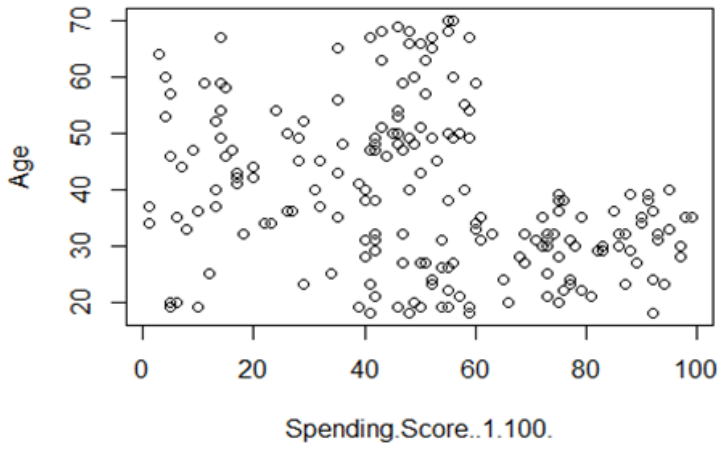


Figure 8: Scatterplot of Age vs. Spending Score

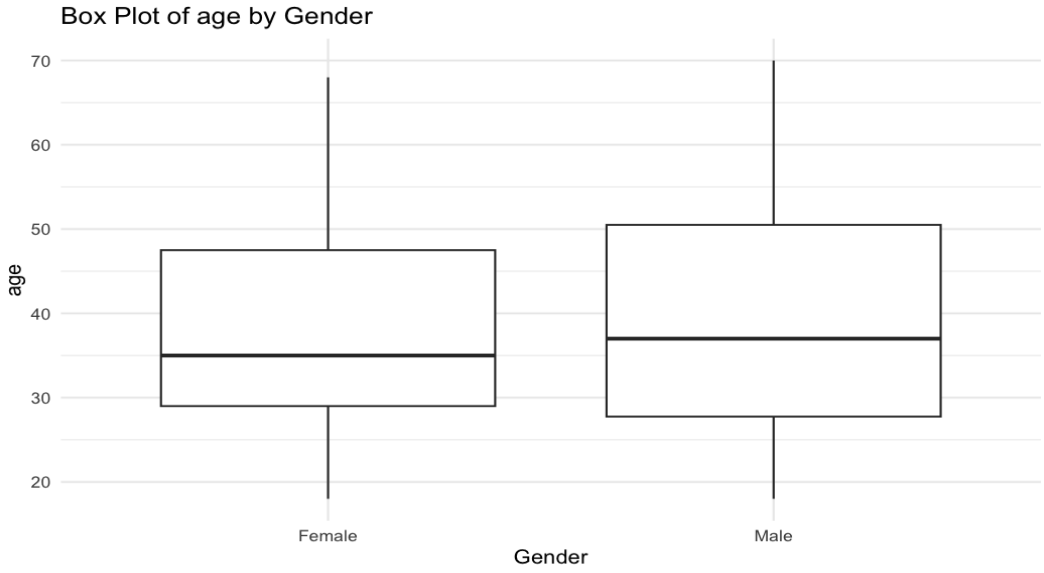


Figure 9: Boxplot of Age vs. Gender

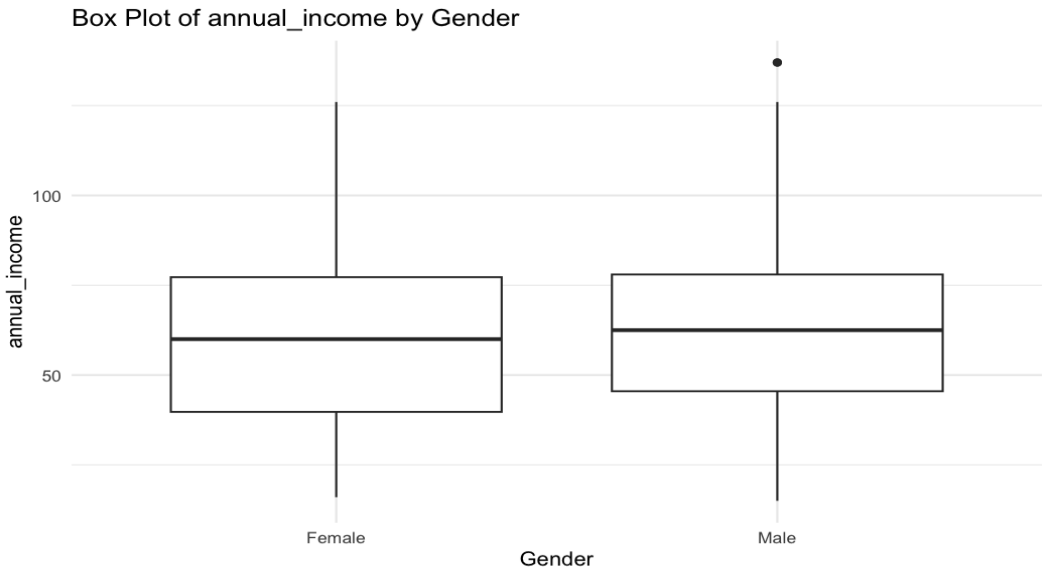


Figure 10: Boxplot of Annual Income vs Gender

Figure 11: Boxplot of Spending Score vs Gender

APPENDIX B

K-Means Clustering

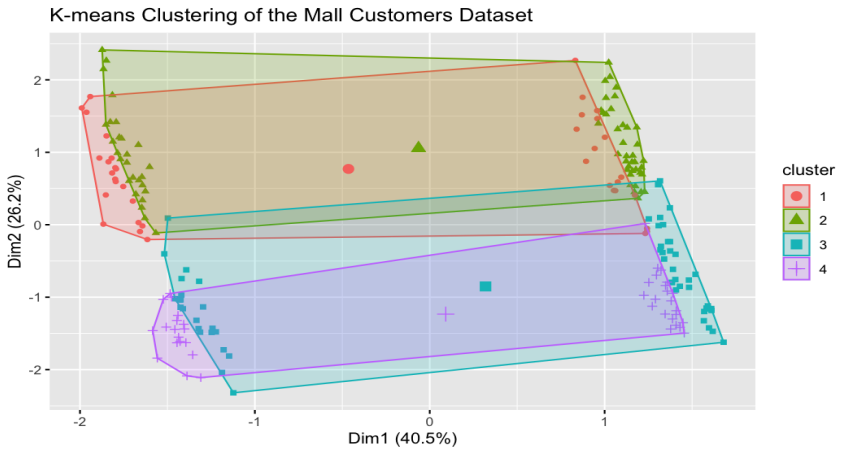


Fig 1: K-means Clustering of the Mall Customer Dataset with 4 clusters

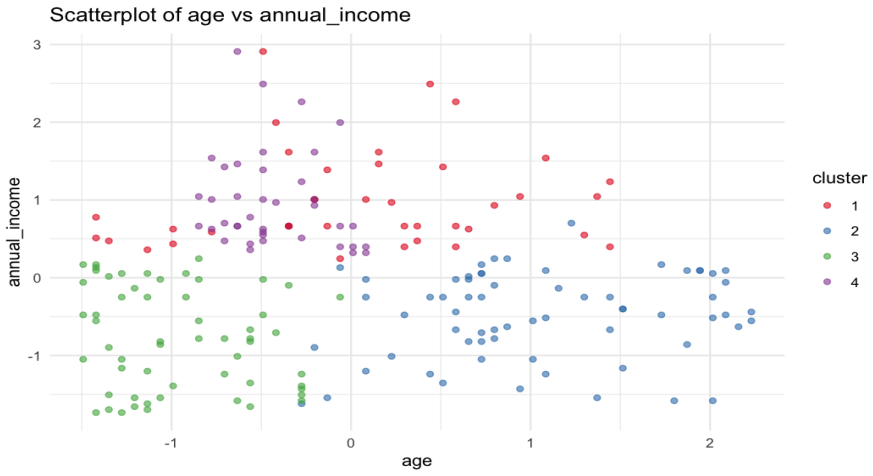


Fig 2: Age v/s annual income scatter plot with 4 labelled clusters from K-means

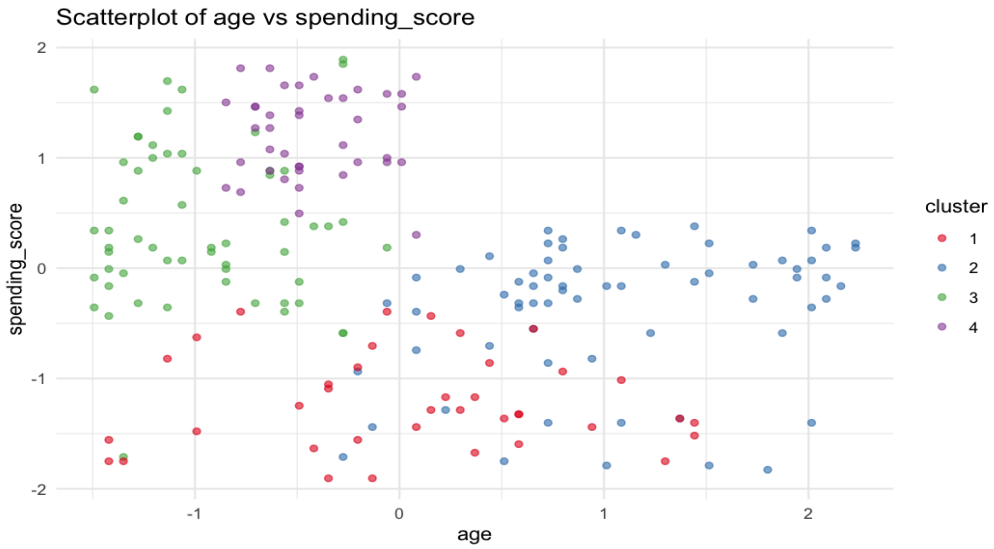


Fig 3: Age v/s spending score scatter plot with 4 labelled clusters from K-means

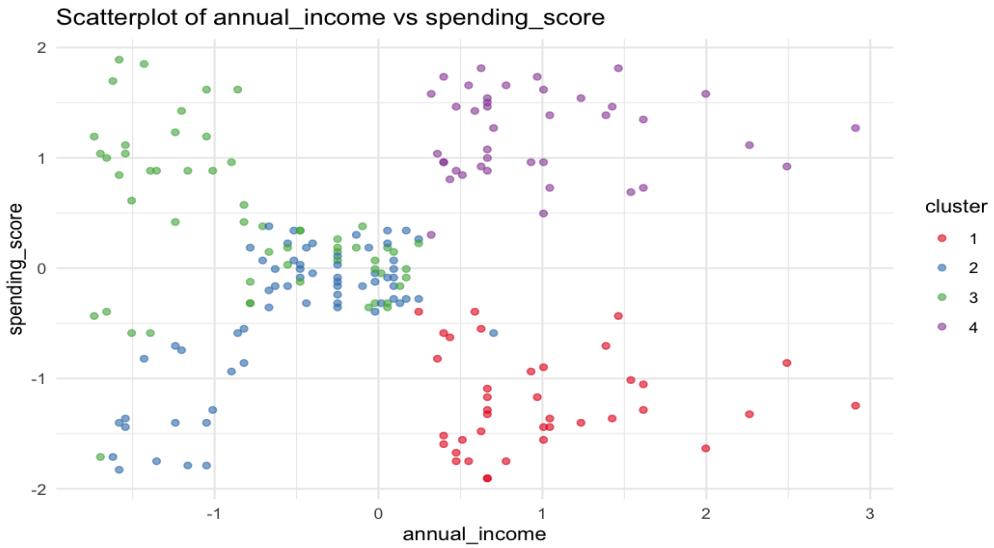


Fig 4: Annual Income v/s spending score scatter plot with 4 labelled clusters from K-means

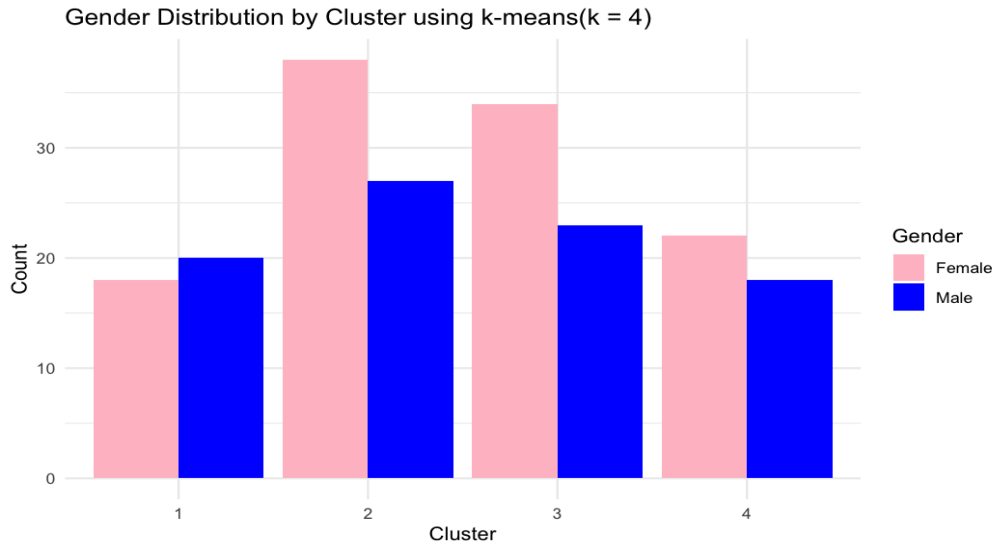


Fig 5: Gender Distribution in 4 different clusters using K-means

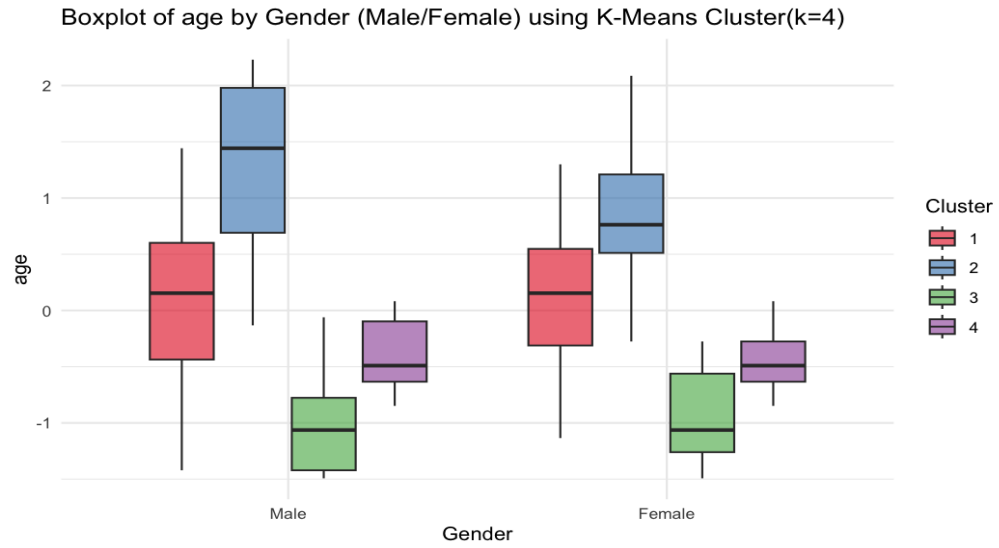


Fig 6: Box plot of Gender v/s age using K-means (k=4)

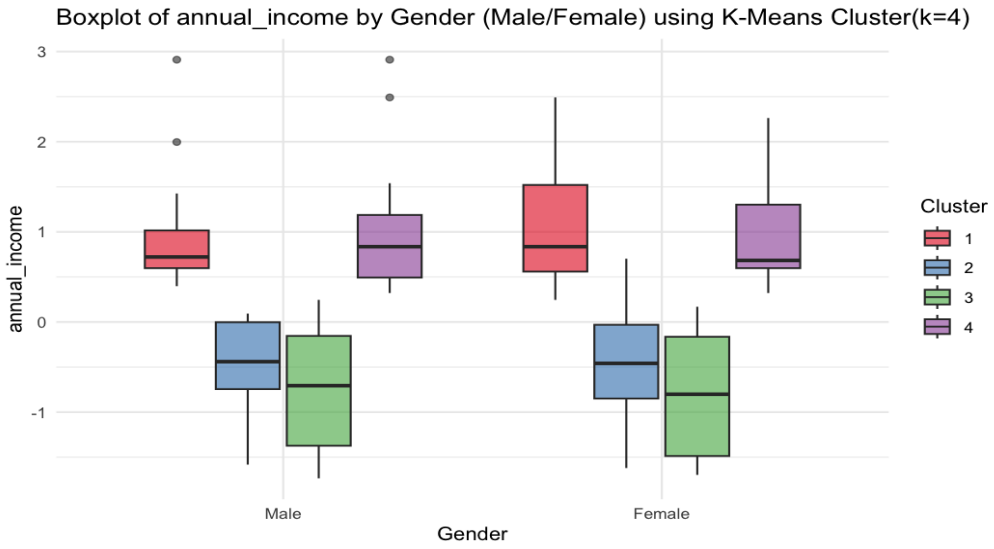


Fig 7: Box plot of Gender v/s Annual Income using K-means (k=4)

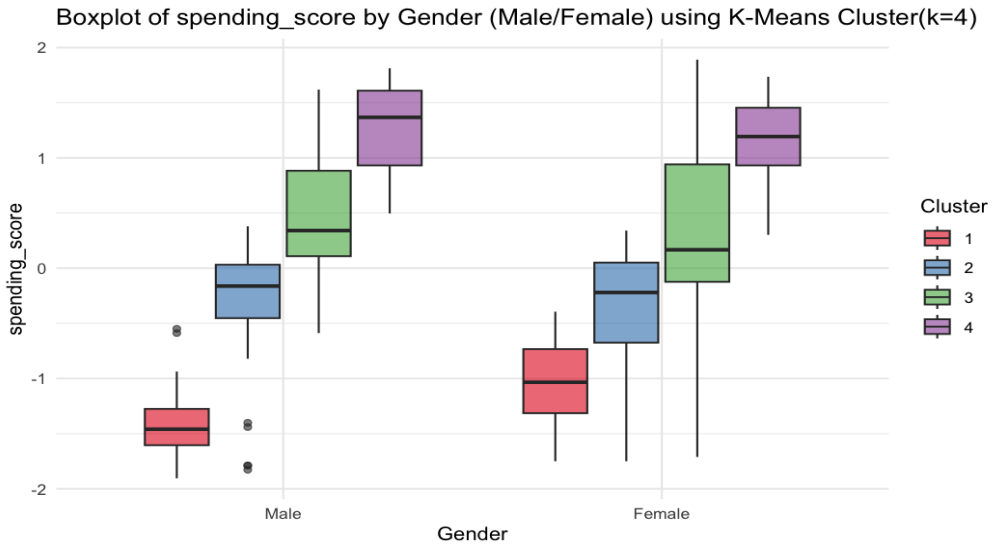


Fig 8: Box plot of Gender v/s Spending Score using K-means (k=4)

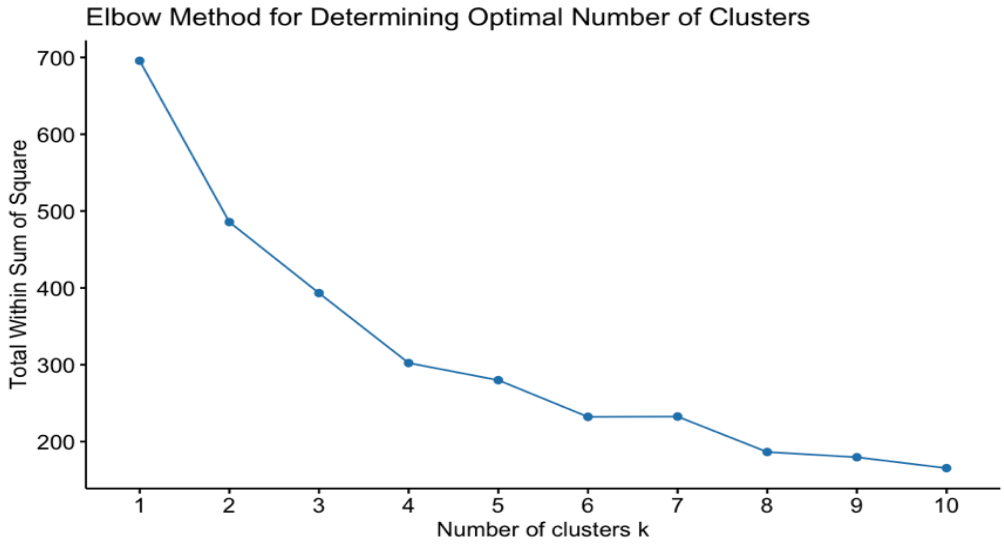


Fig 9: Elbow method for Determining Optimal Number of Clusters

APPENDIX C

Hierarchial Clustering

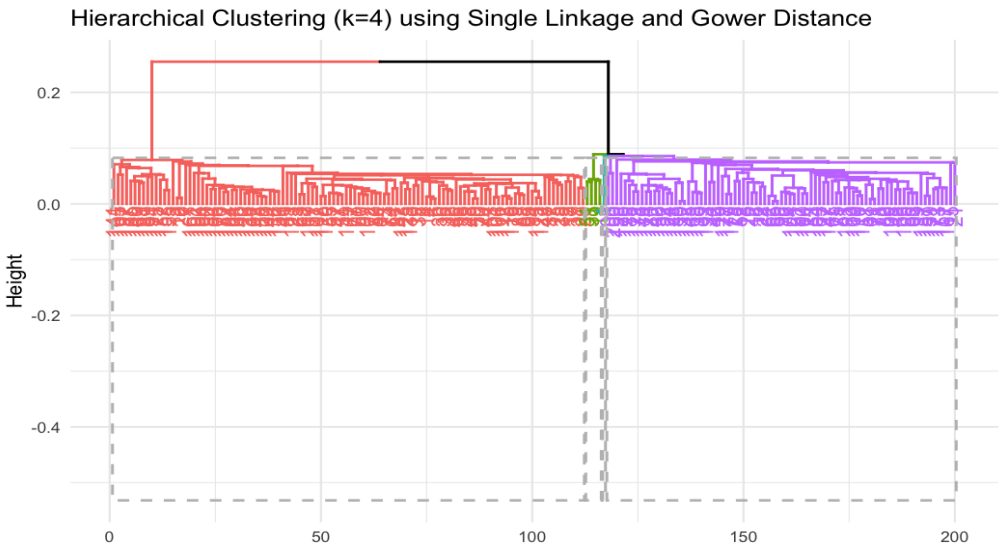


Fig 1: Hierarchical Clustering (k=4) using Single Linkage and Gower Distance

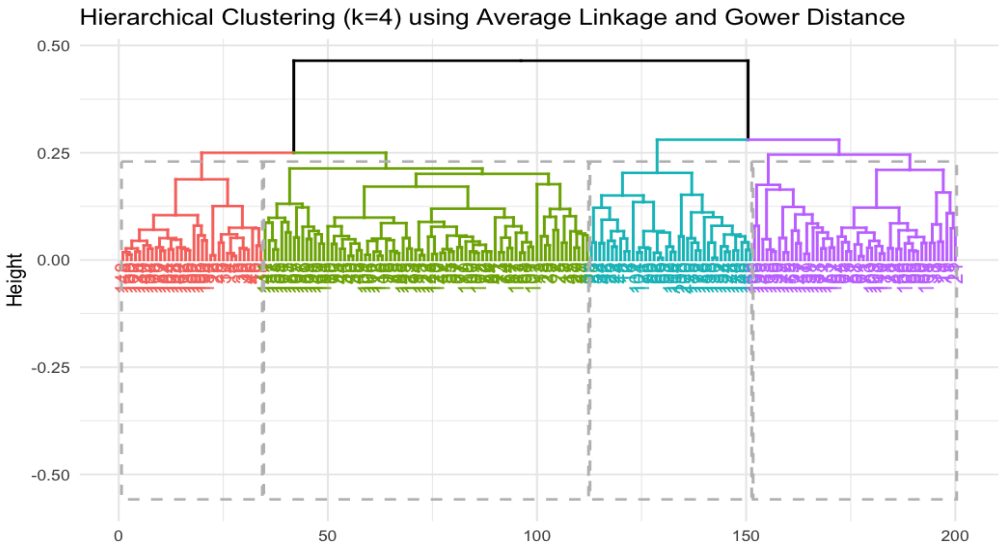


Fig 2: Hierarchical Clustering (k=4) using Average Linkage and Gower Distance

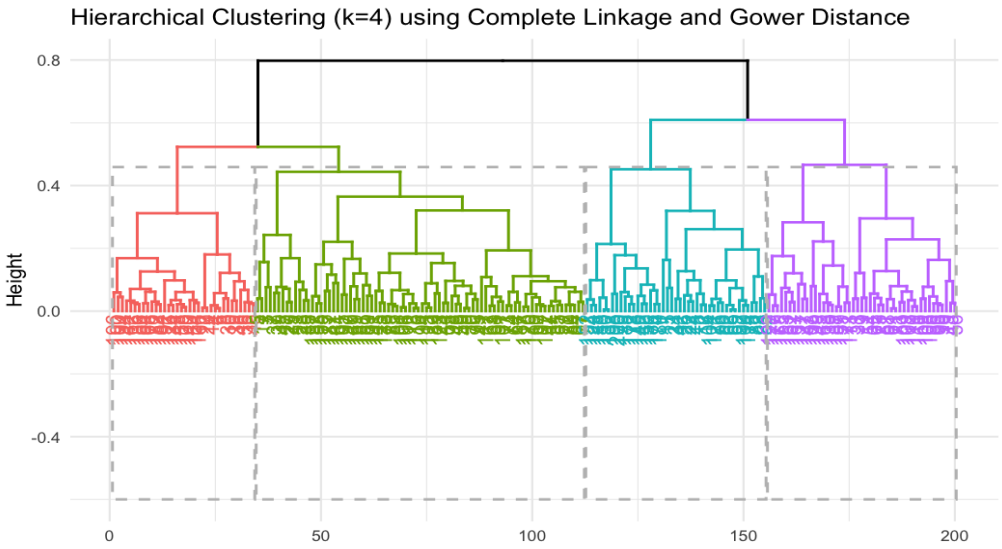


Fig 3: Hierarchical Clustering (k=4) using Complete Linkage and Gower Distance

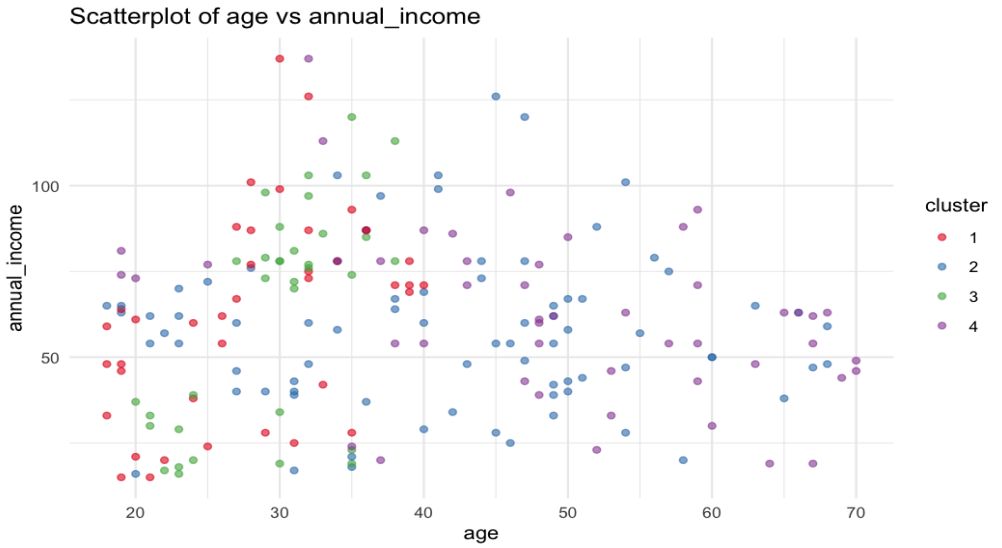


Fig 4: Scatterplot of age v/s annual income with 4 clusters using Hierarchical Clustering using Gower Distance

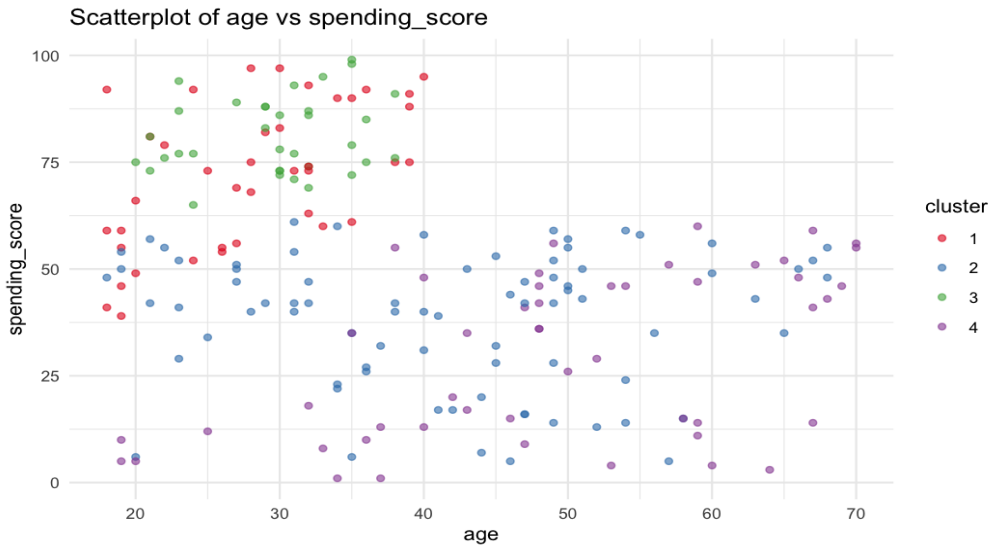


Fig 5: Scatterplot of age v/s spending score with 4 clusters using Hierarchical Clustering using Gower Distance

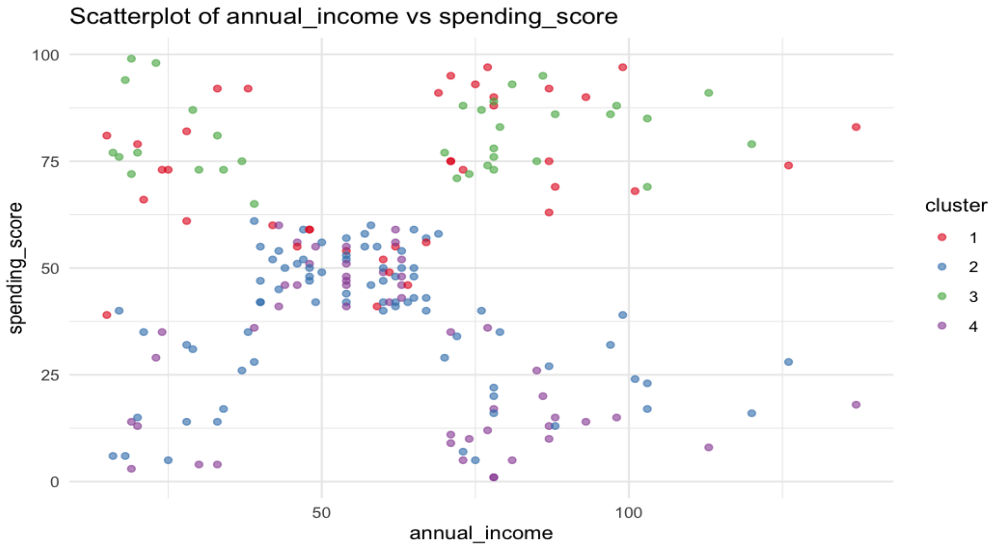


Fig 6: Scatter Plot of annual income v/s spending score with 4 clusters using Hierarchical Clustering using Gower Distance

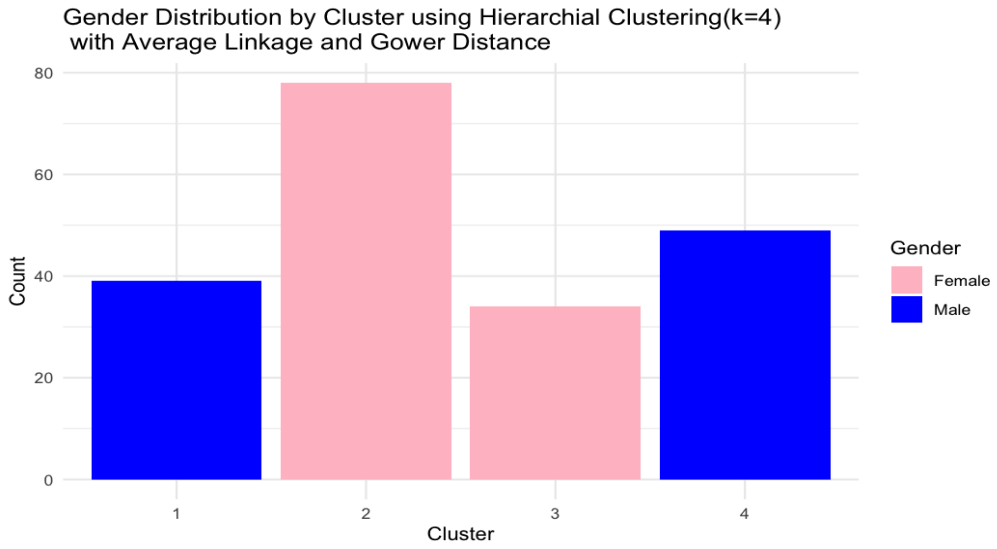


Fig 7: Bar graph with gender distribution with 4 clusters using Hierarchical Clustering and Gower Distance

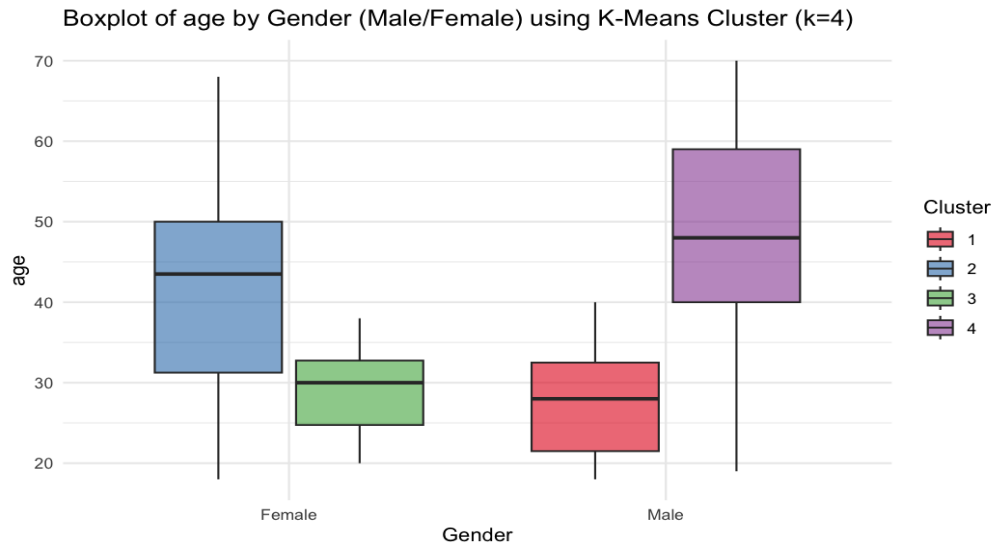


Fig 8: Box plot of gender v/s age using Hierarchical Clustering (4 clusters) using Gower Distance

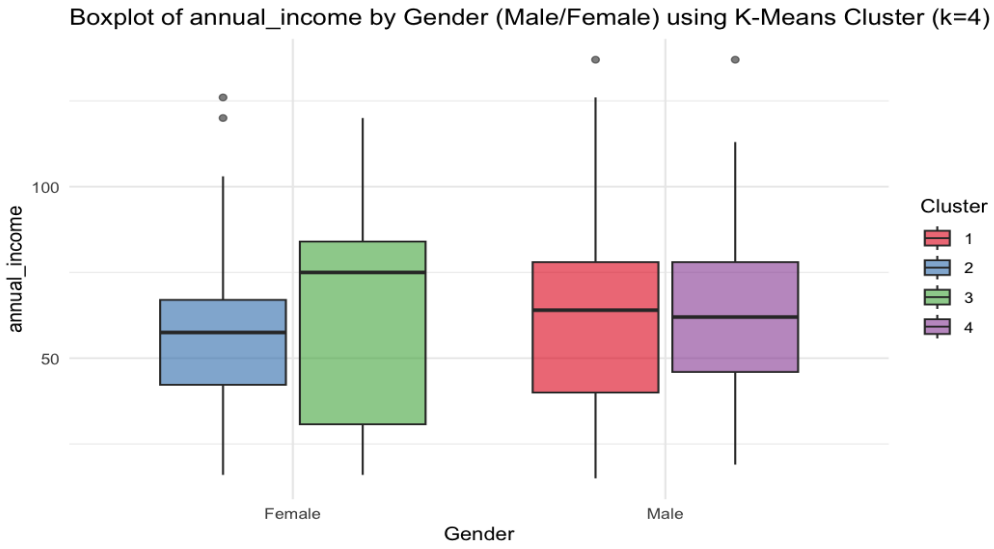


Fig 9: Box plot of gender v/s annual income using Hierarchical Clustering (4 clusters) using Gower Distance

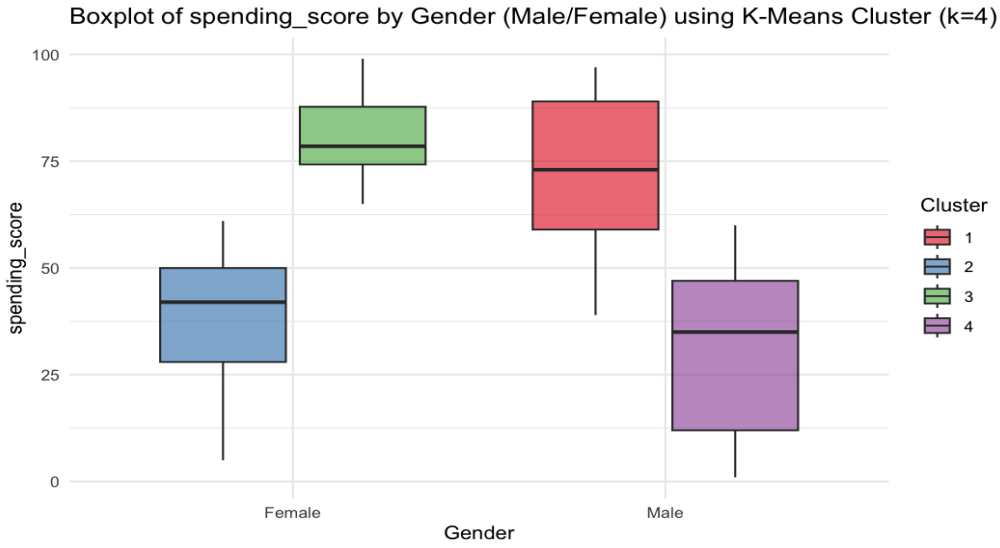


Fig 10: Box plot of gender v/s spending score using Hierarchical Clustering (4 clusters) using Gower Distance

Comparison between Hierarchical and K-means clustering detail

When analyzing the clustering results, we observed notable differences between K-means

clustering and hierarchical clustering using Gower distance. K-means clustering with four clusters resulted in sizes of 38, 40, 47, and 65 (in ascending order), showing a relatively uniform distribution of variables across clusters. In contrast, hierarchical clustering produced cluster sizes of 34, 39, 49, and 78, which were less balanced. This disparity is largely due to the Gower distance metric used in hierarchical clustering, which accommodates both quantitative and categorical variables. Consequently, hierarchical clustering distinctly separated gender, forming two clusters exclusively of male subjects and two exclusively of female subjects. This explains the uneven cluster sizes, as the dataset includes more females than males. On the other hand, K-means clustering provided a more balanced representation of variables, including gender, resulting in more uniformly sized clusters.

Further examination of scatter plots and box plots revealed additional distinctions between the two methods. In scatterplots of annual income vs. age, age vs. spending score, and annual income vs. spending score, K-means clustering achieved better separation, with well-defined clusters. In contrast, hierarchical clustering showed significant overlap in these quantitative variable relationships. However, hierarchical clustering excelled in separating clusters by gender, as demonstrated in the box plots. For instance, in age vs. gender and spending score vs. gender plots, hierarchical clustering formed non-overlapping clusters for males and females, capturing distinct age groups within each gender. In comparison, K-means clusters showed overlap in these relationships. These results highlight how the choice of clustering method and distance metric can influence the results, with K-means excelling in quantitative relationships and hierarchical clustering emphasizing categorical distinctions.

APPENDIX D

Post Cluster Analysis

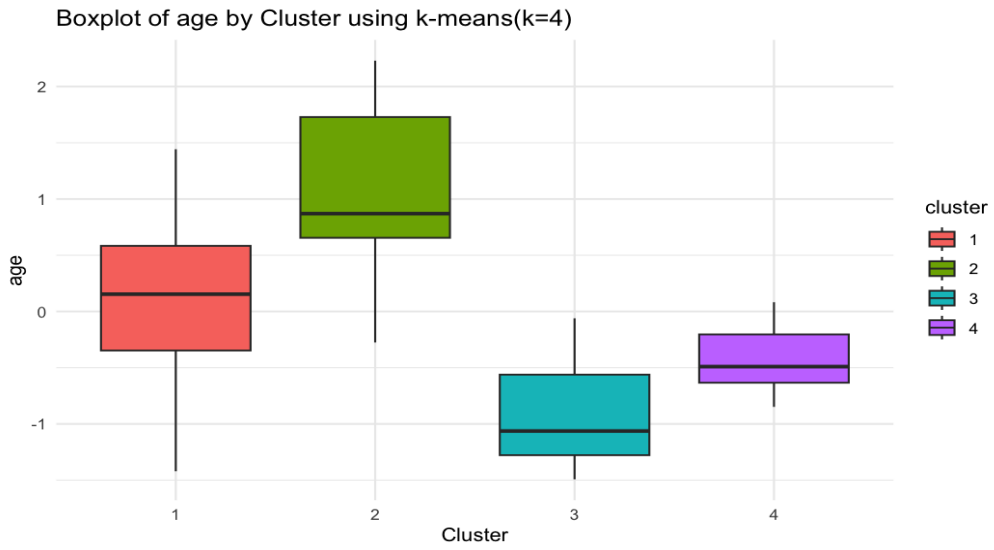


Fig 1: Box plot of Cluster v/s age using K-means (k=4)

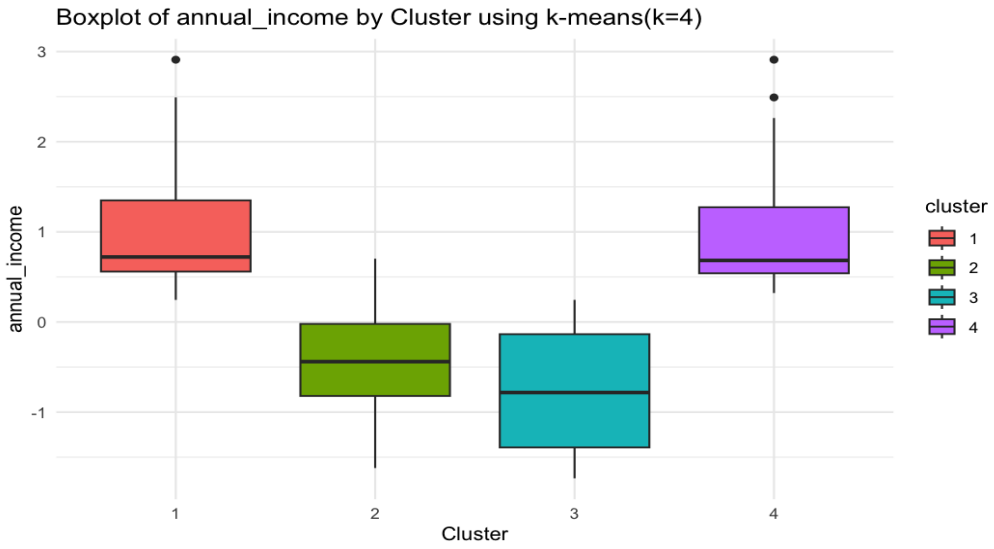


Fig 2: Box plot of Cluster v/s annual income using K-means (k=4)

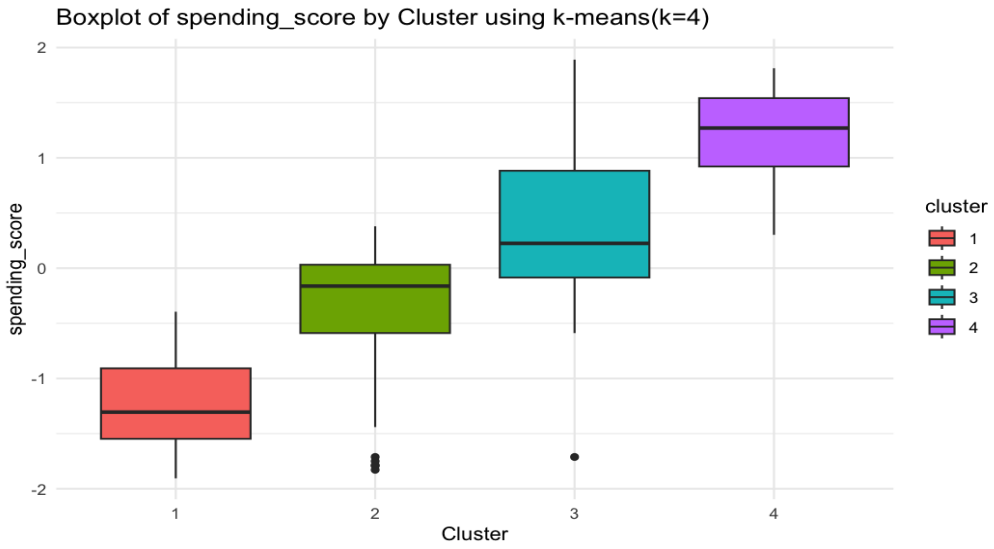
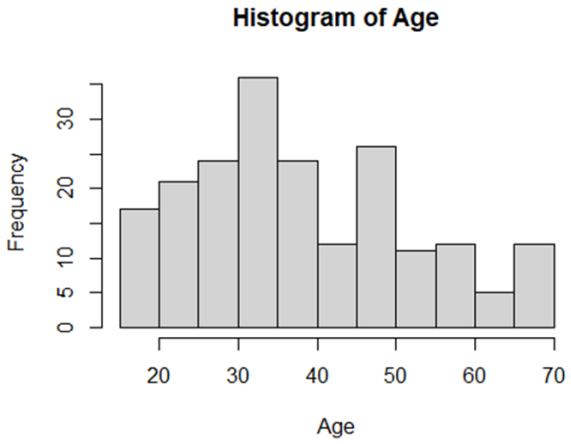


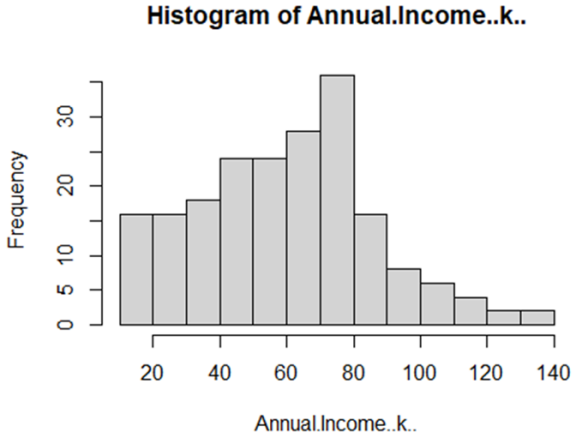
Fig 3: Box plot of Cluster v/s spending score using K-means (k=4)

APPENDIX E

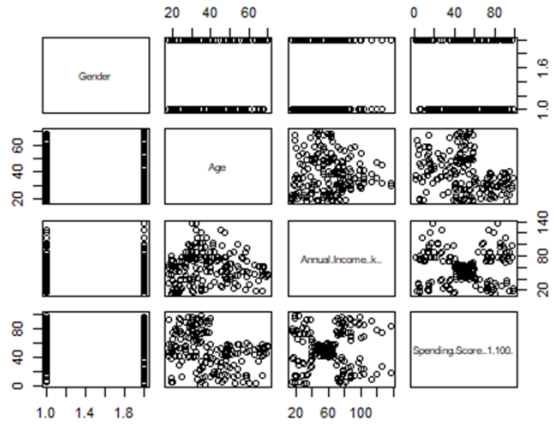
Code for Initial Data Exploration

```
Spending <- read.csv(file = file.choose(), header = TRUE)

Spending <- Spending[, -1]

 Spending$Gender <- factor(Spending$Gender)

 attach(Spending)

numeric_var <- Spending[, -1]

 summary(numeric_var)

##       Age         Annual.Income..k.. Spending.Score..1.100.

 ##  Min.   :18.00   Min.   : 15.00     Min.   : 1.00

##  1st Qu.:28.75   1st Qu.: 41.50     1st Qu.:34.75

##  Median :36.00   Median : 61.50     Median :50.00

##  Mean   :38.85   Mean   : 60.56     Mean   :50.20

##  3rd Qu.:49.00   3rd Qu.: 78.00     3rd Qu.:73.00

##  Max.   :70.00   Max.   :137.00     Max.   :99.00

table(Spending$Gender)

##

## Female   Male

##    112     88

hist(Age)
```

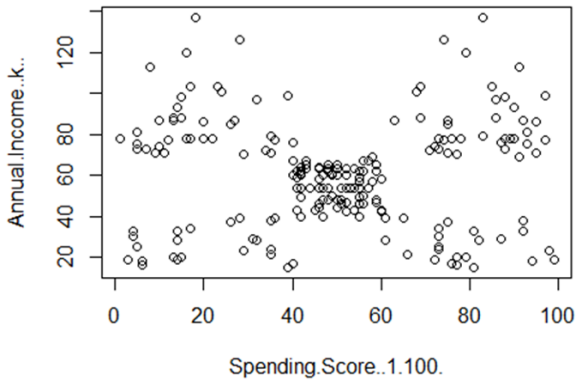


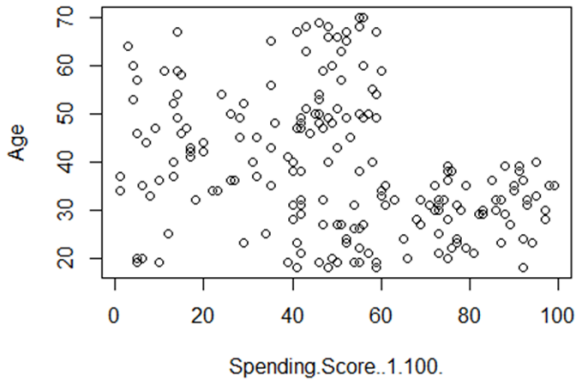



```
hist(Annual.Income..k..)

hist(Spending.Score..1.100.)


cor(numeric_var, use = "complete.obs")

##                                Age Annual.Income..k.. Spending.Score..1.100.

 ## Age                     1.00000000       -0.012398043           -0.327226846

 ## Annual.Income..k..     -0.01239804        1.000000000            0.009902848

 ## Spending.Score..1.100. -0.32722685        0.009902848            1.000000000
```




```
plot(Spending)

plot(Annual.Income..k.. ~ Spending.Score..1.100.)

plot(Age ~ Spending.Score..1.100.)
```


Annual.Income..k..
20
40
60
80
120
0
20
40
60
80
100
Spending.Score..1.100.

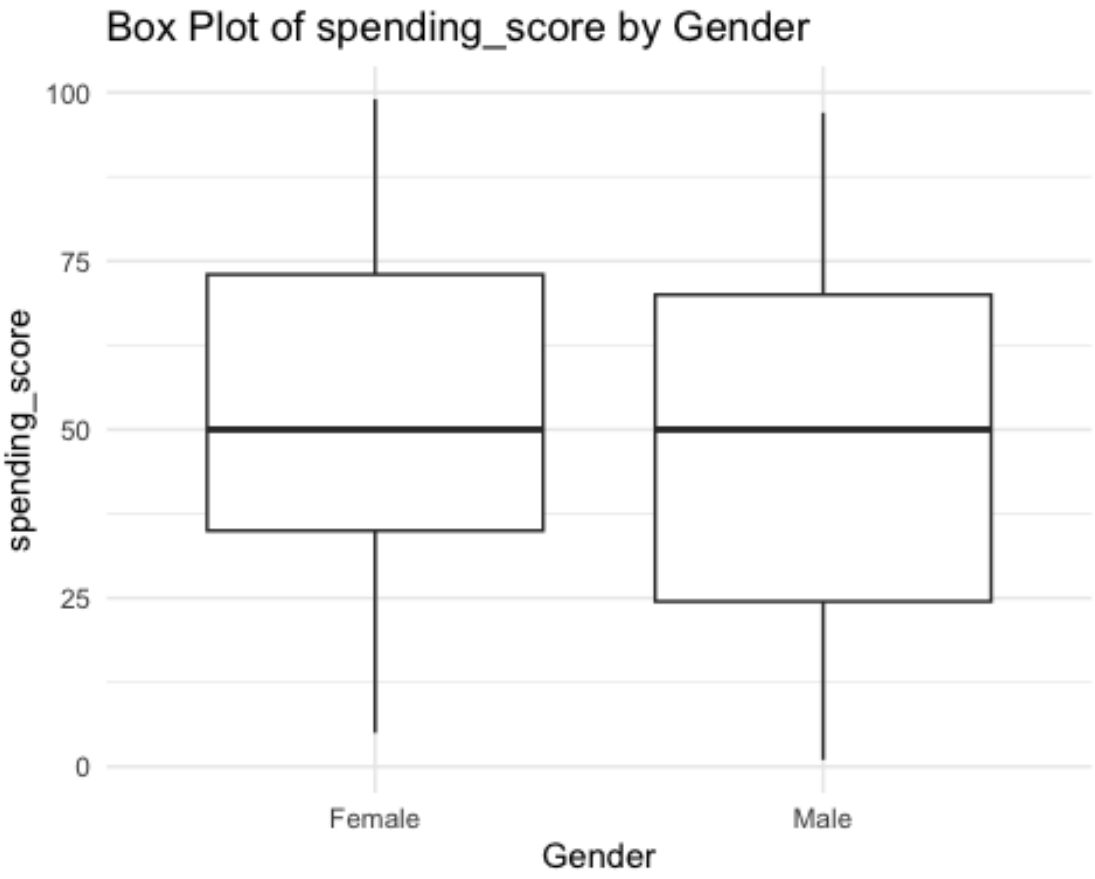

Age
20
30
40
50
60
70
0
20
40
60
80
100
Spending.Score..1.100.

APPENDIX F

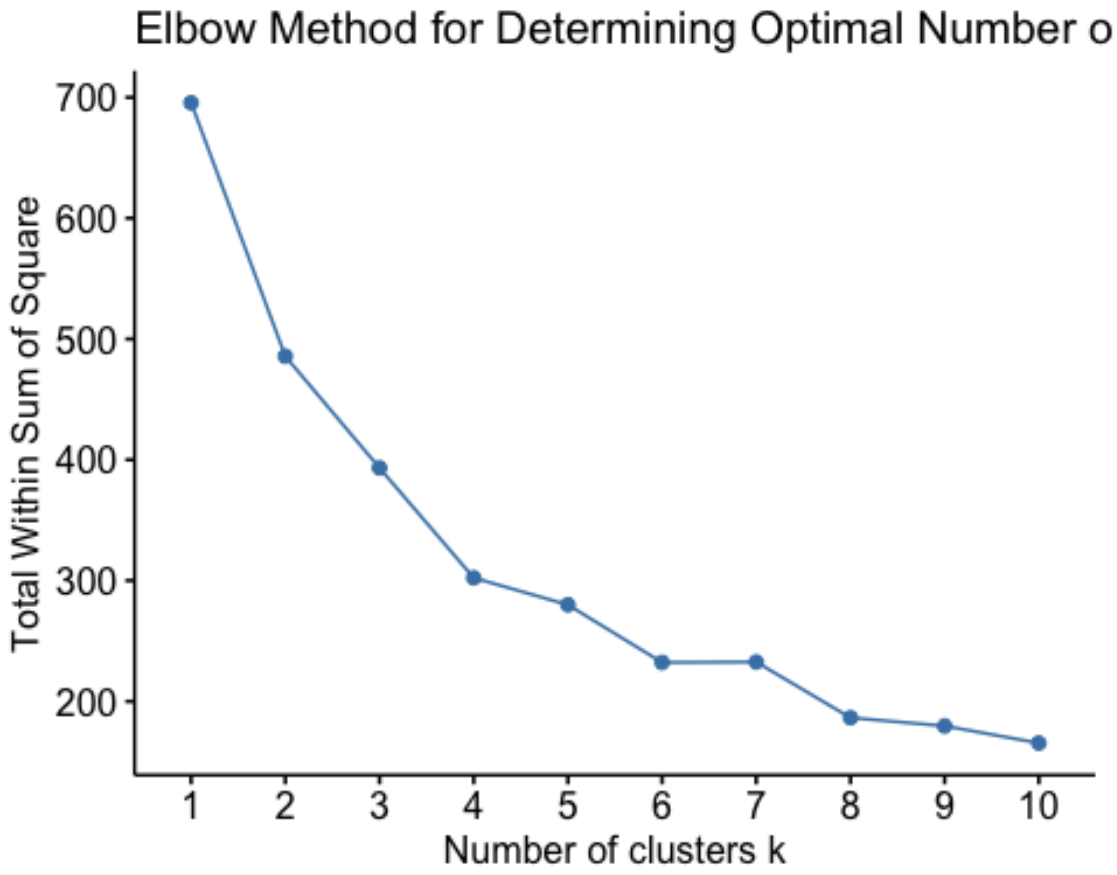
Code for K-means Clustering

```
library(readr)

data <- read_csv("Mall_Customers.csv")

#Exploration

print(str(data))

summary(data)

colSums(is.na(data))

##             CustomerID                 Gender                    Age

##                      0                      0                      0

##     Annual Income (k$) Spending Score (1-100)

##                      0                      0

library(ISLR2)


library(ggplot2)


# Ensure gender is treated as a categorical variable with appropriate labels



for (var in c("age", "annual_income", "spending_score")) {
```

```
  p <- ggplot(main_data, aes_string(x = "gender", y = var)) +
    geom_boxplot() +
    theme_minimal() +
    labs(title = paste("Box Plot of", var, "by Gender"),
         x = "Gender",
         y = var)

  print(p)
}
```

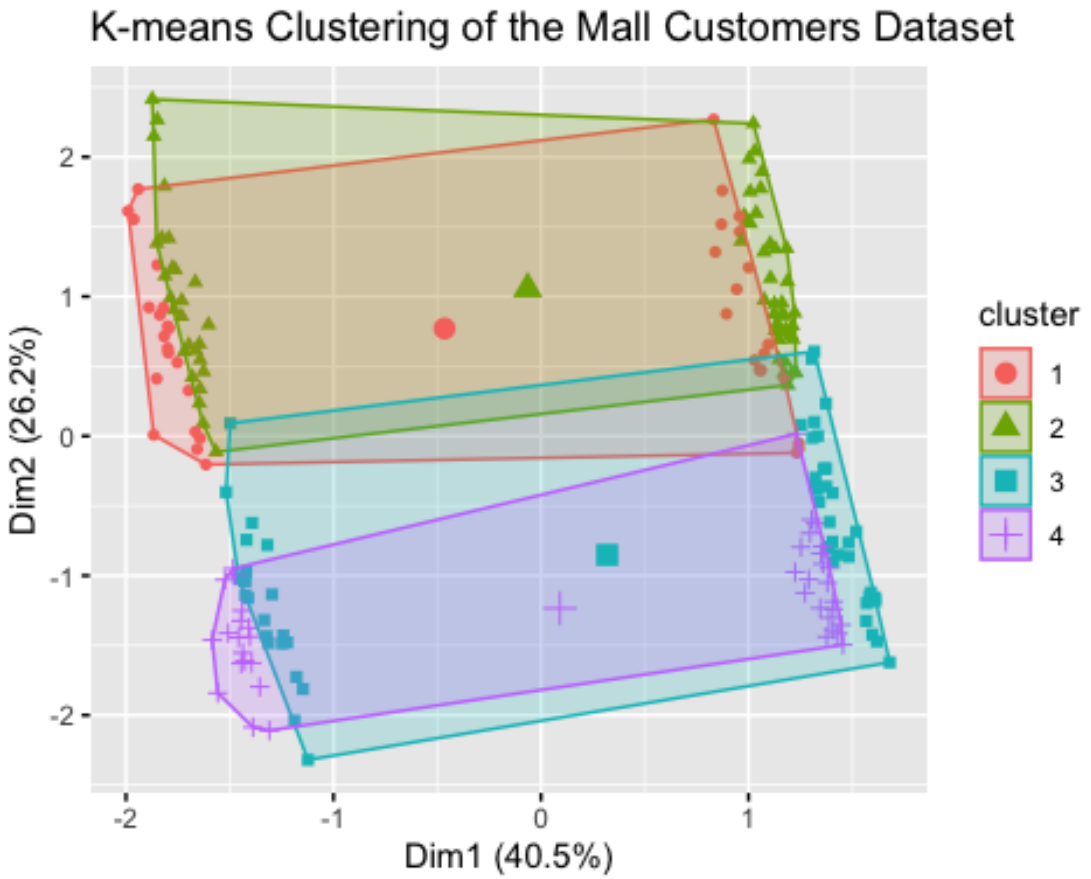


```
scaled_data <- as.data.frame(scale(main_data[, c("age", "annual_income", "spending_score

# One-hot encode the gender variable
one_hot_gender <- as.data.frame(model.matrix(~ gender - 1, data = main_data))

# Combine the scaled numerical data with the one-hot encoded gender variable
final_data <- cbind(one_hot_gender, scaled_data)

# View the result
head(final_data)
# Load the factoextra package for visualization

library(factoextra)
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WB
# Generate the elbow plot
fviz_nbclust(final_data, kmeans, method = "wss") +
  labs(title = "Elbow Method for Determining Optimal Number of Clusters")
```




```
# Set the number of clusters

k <- 4


# Perform k-means clustering

set.seed(123)  # Set seed for reproducibility

kmeans_result <- kmeans(final_data, centers = k, nstart = 25)


# View clustering results

kmeans_result
```

```
## K-means clustering with 4 clusters of sizes 38, 65, 57, 40
##
## Cluster means:
##   genderFemale genderMale        age annual_income spending_score
## 1    0.4736842  0.5263158  0.0766735     0.9946502     -1.2173724
## 2    0.5846154  0.4153846  1.0603143    -0.4934376     -0.3777117
## 3    0.5964912  0.4035088 -0.9600828    -0.7827991      0.3910484
## 4    0.5500000  0.4500000 -0.4277326     0.9724070      1.2130414
##
## Within cluster sum of squares by cluster:
## [1]  63.80312 105.87053  88.87074  43.71544
##  (between_SS / total_SS =  56.5 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
# Visualize the clusters with different colors
fviz_cluster(kmeans_result, data = final_data,
             geom = "point") +
  labs(title = "K-means Clustering of the Mall Customers Dataset")
```




```
kmeans_result$size
```

```
## [1] 38 65 57 40
```

```
library(tidyverse)
```

```
final_data_visualization = final_data
```

```
final_data_visualization$cluster <- as.factor(kmeans_result$cluster)
```

```
final_data_visualization
```

```
# # Create boxplots for numeric variables and bar graphs for gender variables
```

```
variables <- colnames(final_data_visualization)[-ncol(final_data_visualization)]  # Excl
variables
## [1] "genderFemale"   "genderMale"     "age"            "annual_income"
## [5] "spending_score"
#
for (var in variables) {
  if (var %in% c("genderMale", "genderFemale")) {


   print("Doing nothing")
    # Print the bar graph
  } else {
    # Create boxplot for numeric variables
    plot <- ggplot(final_data_visualization, aes_string(x = "cluster", y = var, fill = "
      geom_boxplot() +
      labs(title = paste("Boxplot of", var, "by Cluster using k-means(k=4)"),
           x = "Cluster",
           y = var) +
      theme_minimal()
       print(plot)
  }
```

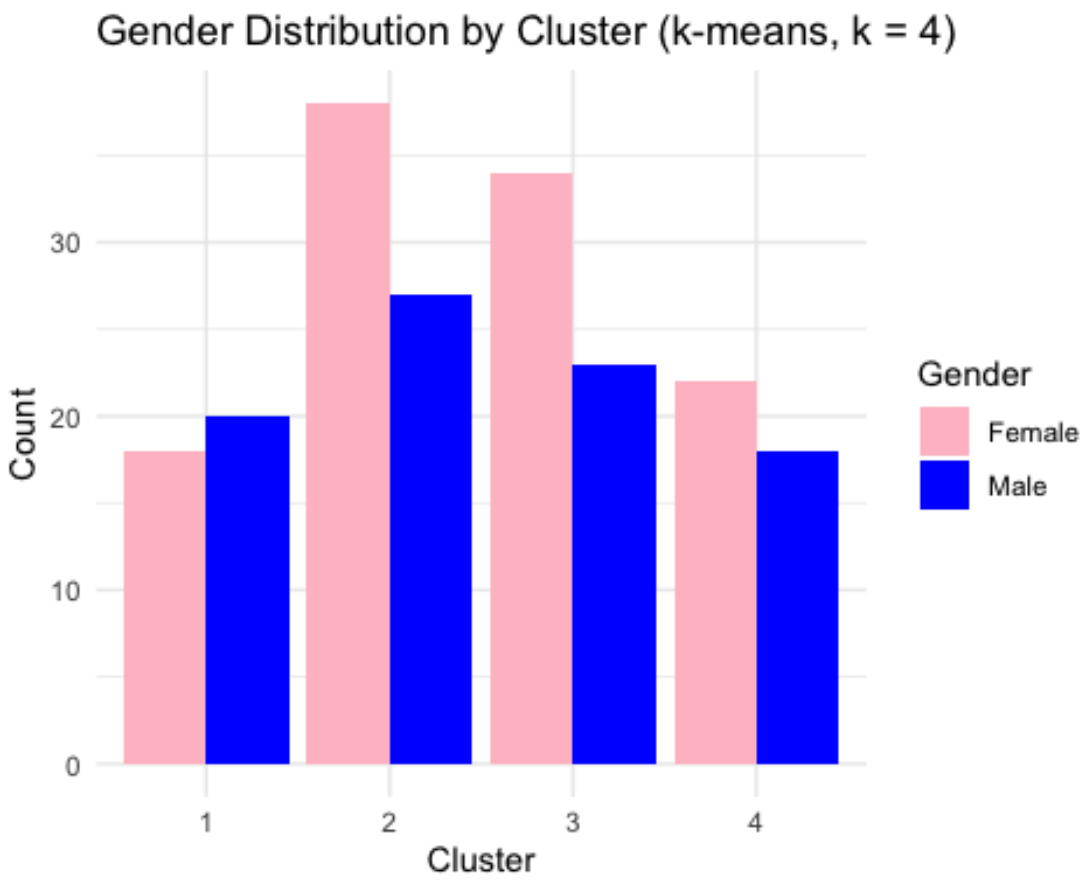
```
}
```

```
## [1] "Doing nothing"
```

```
## [1] "Doing nothing"
```

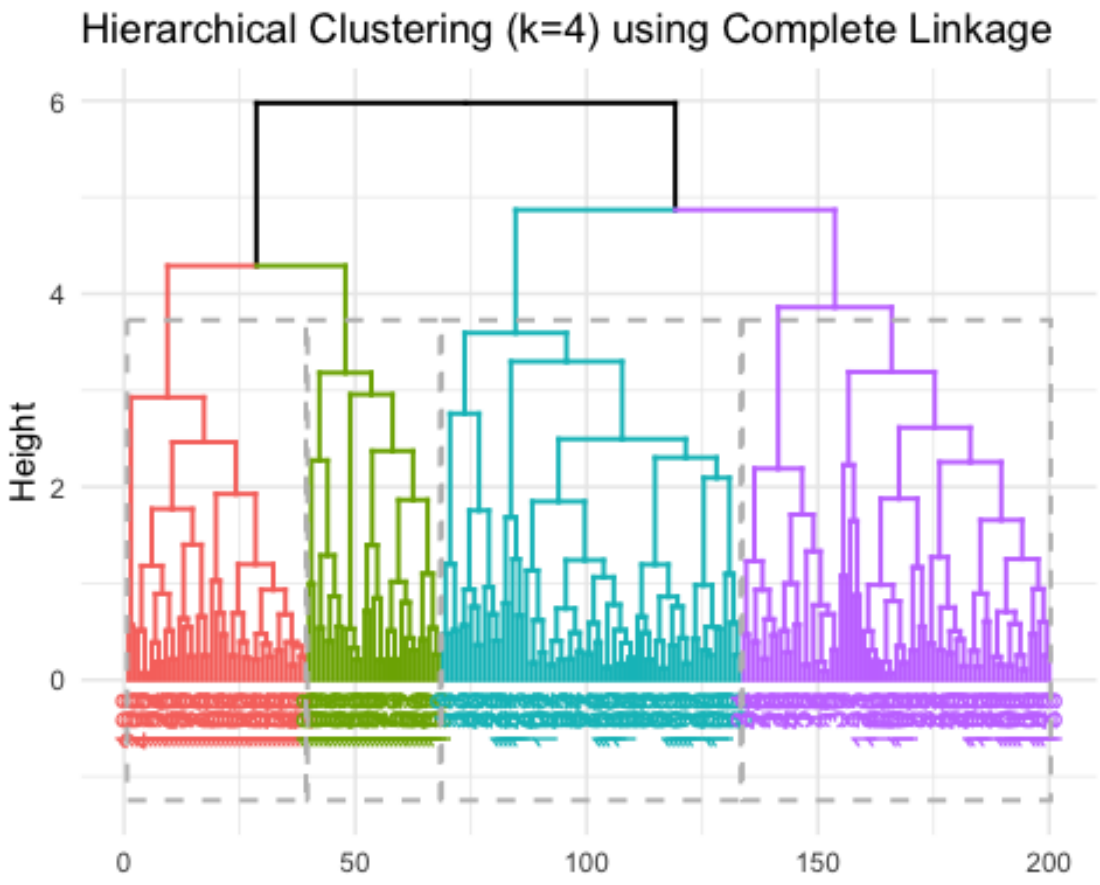

```
gender_data <- final_data_visualization %>%
  select(cluster, genderMale, genderFemale) %>% # Select cluster and gender columns
  pivot_longer(cols = c(genderMale, genderFemale), # Convert wide data to long format
               names_to = "Gender",
               values_to = "Count") %>%
  mutate(Gender = ifelse(Gender == "genderMale", "Male", "Female")) # Rename gender valu
```

```
# Summarize gender counts for each cluster

# Group by cluster and gender, then calculate total counts

gender_counts <- gender_data %>%

  group_by(cluster, Gender) %>%

  summarise(Count = sum(Count), .groups = "drop") # Drop grouping after summarizing


# Create a side-by-side bar chart to show gender distribution across clusters

plot <- ggplot(gender_counts, aes(x = cluster, y = Count, fill = Gender)) +

  geom_bar(stat = "identity", position = "dodge") + # Use dodge for side-by-side bars

  scale_fill_manual(values = c("Male" = "blue", "Female" = "pink")) + # Custom colors fo

  labs(title = "Gender Distribution by Cluster (k-means, k = 4)",

       x = "Cluster", # X-axis label

       y = "Count", # Y-axis label

       fill = "Gender") + # Legend title

  theme_minimal() # Apply a clean theme


# Display the plot

print(plot)
```

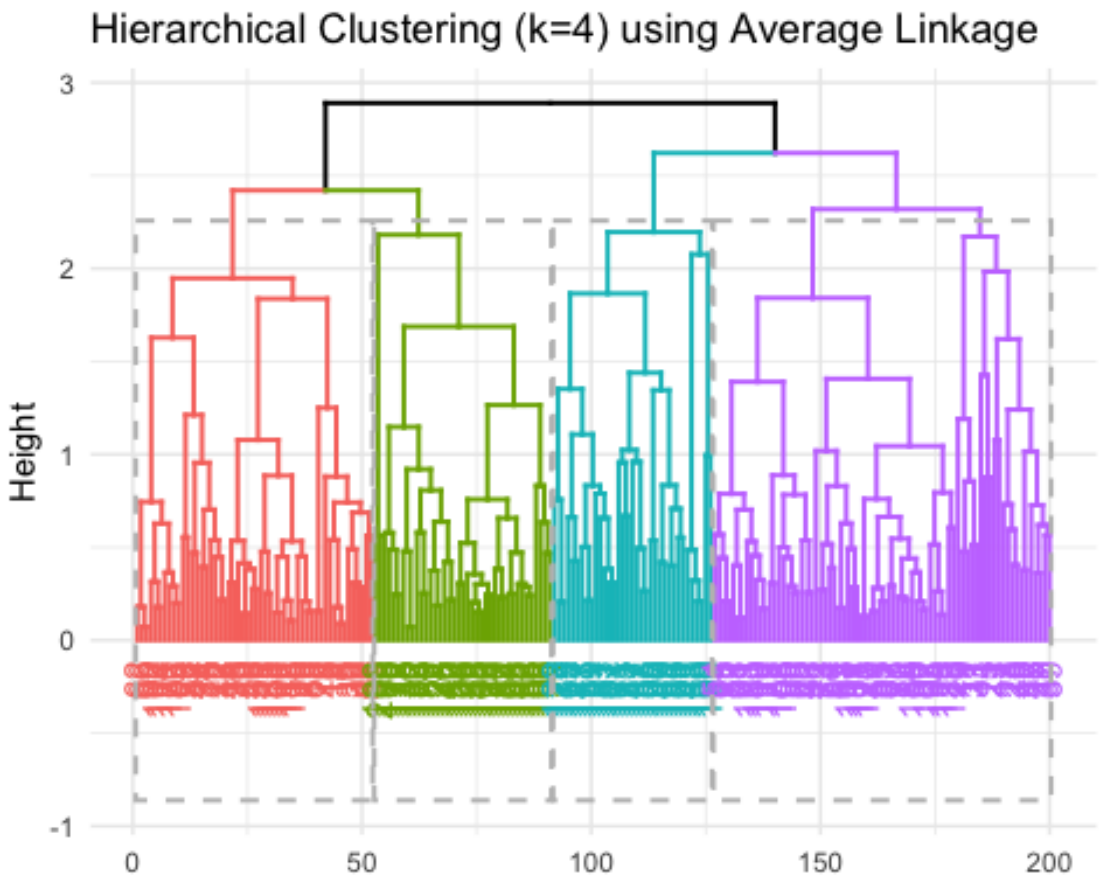



```
d<- dist(final_data, method="euclidean")

library(factoextra)


# Perform hierarchical clustering

hc_complete <- hclust(d, method = "complete")


# Visualize the dendrogram with a title

fviz_dend(hc_complete, k = 4, rect = TRUE) +

  labs(title = "Hierarchical Clustering (k=4) using Complete Linkage") +
```

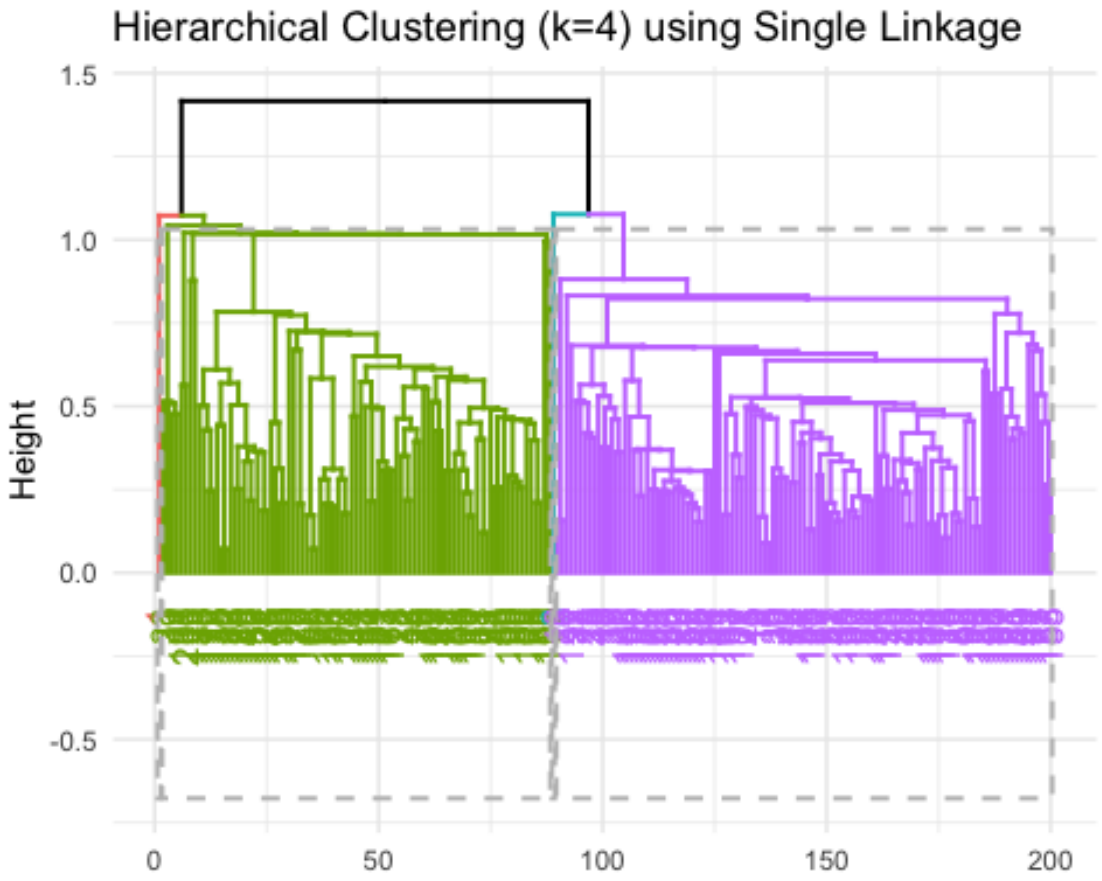
```
  theme_minimal()
```




```
# Perform hierarchical clustering
hc_complete <- hclust(d, method = "average")


# Visualize the dendrogram with a title
fviz_dend(hc_complete, k = 4, rect = TRUE) +
  labs(title = "Hierarchical Clustering (k=4) using Average Linkage") +
  theme_minimal()
```

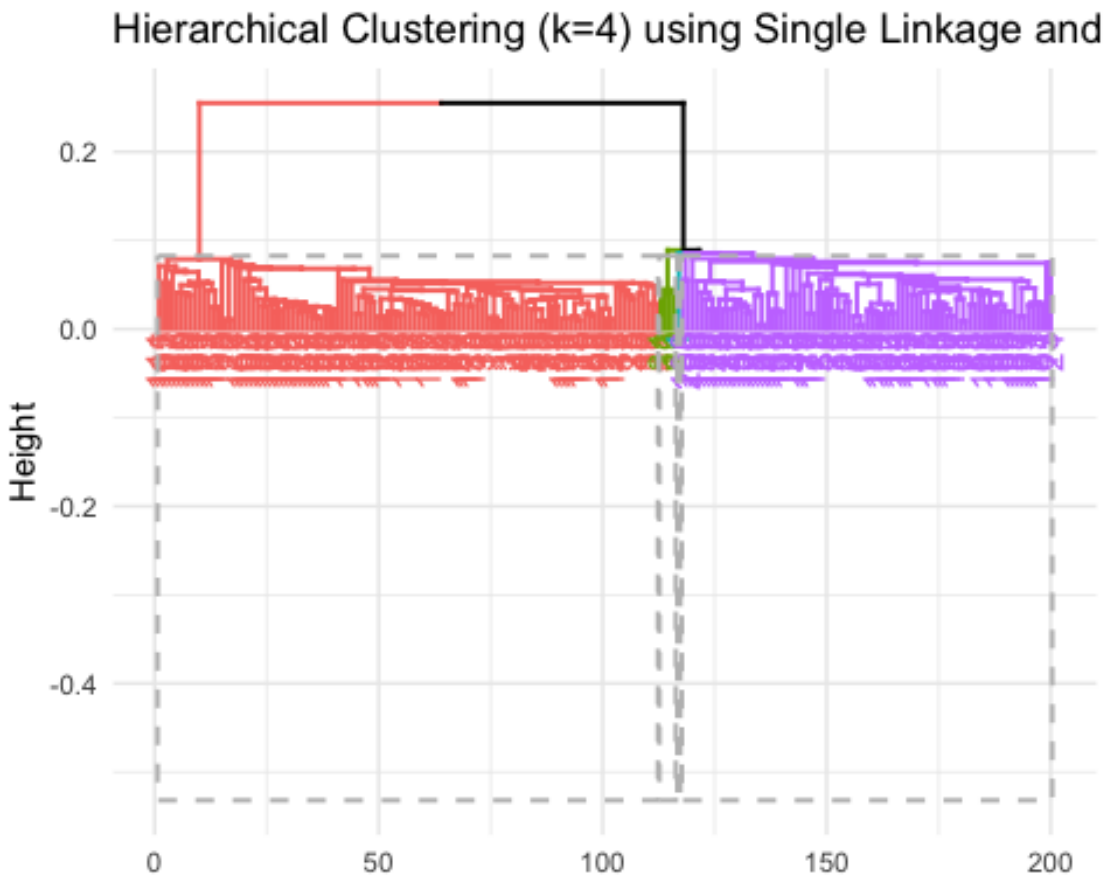



```
# Perform hierarchical clustering
hc_complete <- hclust(d, method = "single")


# Visualize the dendrogram with a title
fviz_dend(hc_complete, k = 4, rect = TRUE) +
  labs(title = "Hierarchical Clustering (k=4) using Single Linkage") +
  theme_minimal()
```



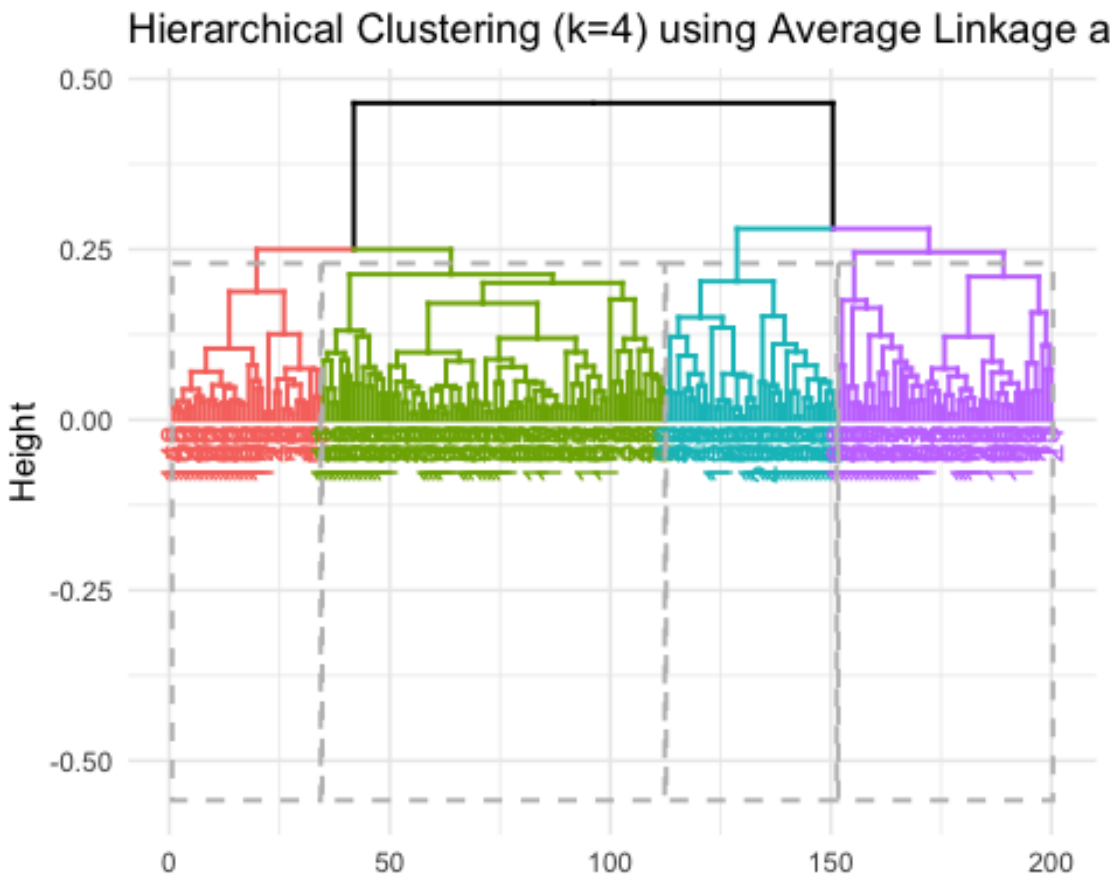

```
#Using Gower Distance

library(cluster)

library(factoextra)


# Compute Gower distance

gower_dist <- daisy(main_data, metric = "gower")


# Perform hierarchical clustering with single linkage

hc_complete <- hclust(gower_dist, method = "single")
```

```
# Visualize the dendrogram with improved scaling and spacing
fviz_dend(hc_complete, k = 4, rect = TRUE, cex = 0.8, horiz = FALSE) +
  labs(title = "Hierarchical Clustering (k=4) using Single Linkage and Gower Distance") +
  theme_minimal()
```

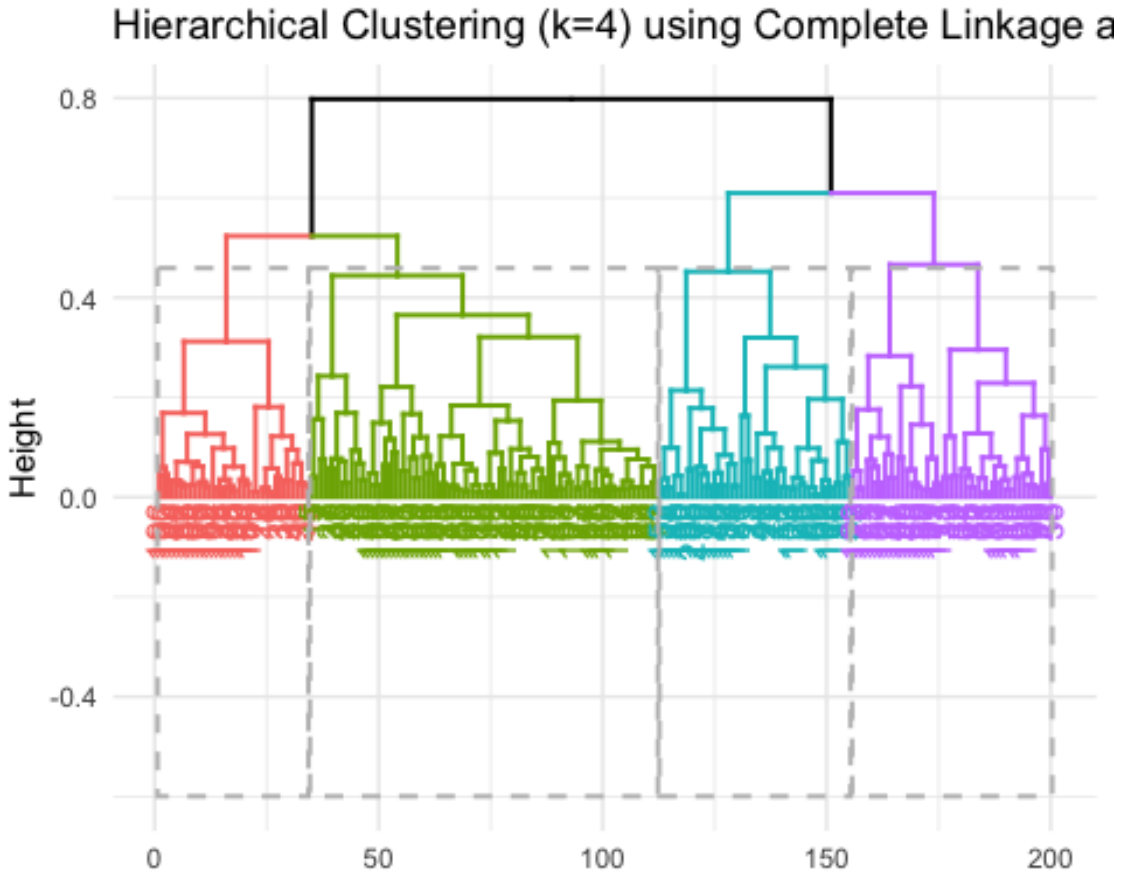



```
#Using Gower Distance
library(cluster)
library(factoextra)
```

```
# Compute Gower distance
gower_dist <- daisy(main_data, metric = "gower")

# Perform hierarchical clustering with single linkage
hc_complete <- hclust(gower_dist, method = "average")

# Visualize the dendrogram with improved scaling and spacing
fviz_dend(hc_complete, k = 4, rect = TRUE, cex = 0.8, horiz = FALSE) +
  labs(title = "Hierarchical Clustering (k=4) using Average Linkage and Gower Distance") +
  theme_minimal()
```

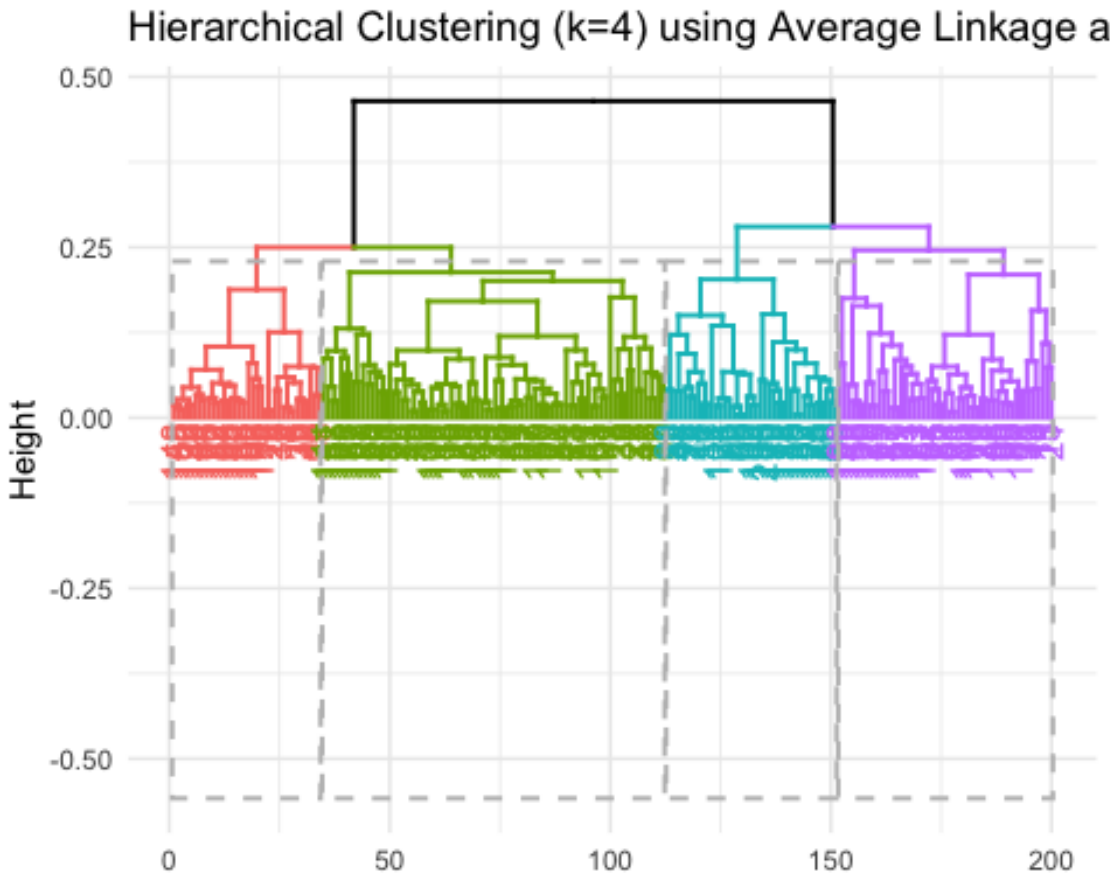


```
#Using Gower Distance

library(cluster)

library(factoextra)


# Compute Gower distance

gower_dist <- daisy(main_data, metric = "gower")


# Perform hierarchical clustering with single linkage

hc_complete <- hclust(gower_dist, method = "complete")


# Visualize the dendrogram with improved scaling and spacing

fviz_dend(hc_complete, k = 4, rect = TRUE, cex = 0.8, horiz = FALSE) +

  labs(title = "Hierarchical Clustering (k=4) using Complete Linkage and Gower Distance"

  theme_minimal()
```



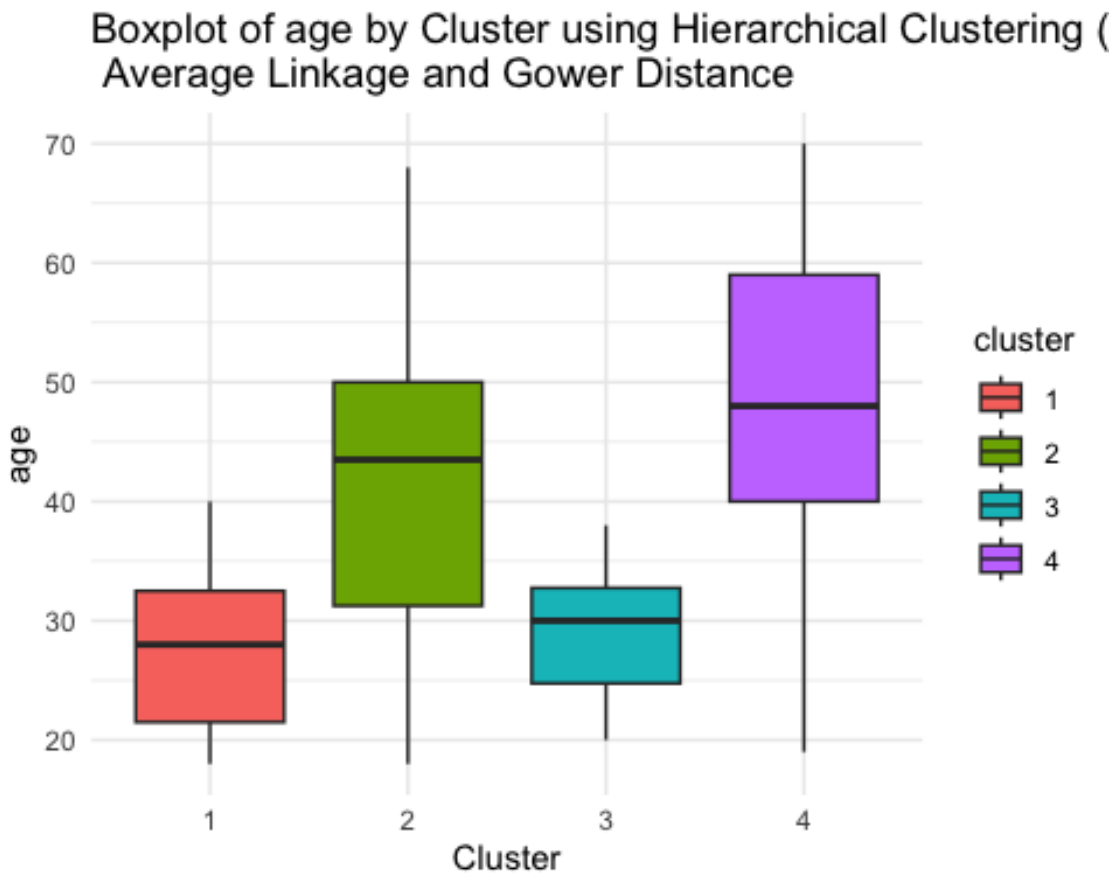

```
library(ggplot2)

library(dplyr)

library(tidyr)

library(cluster)

library(factoextra)


# Compute Gower distance

gower_dist <- daisy(main_data, metric = "gower")
```

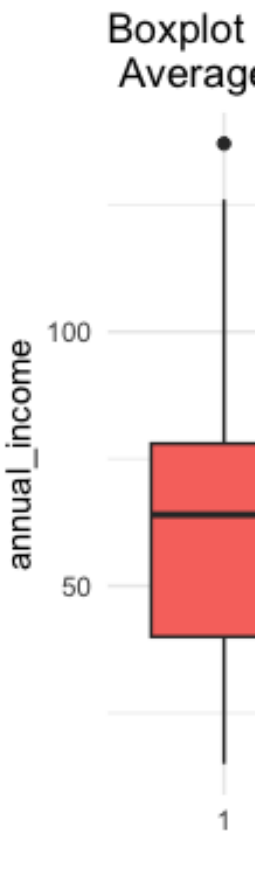
```
# Perform hierarchical clustering with average linkage
hc_complete <- hclust(gower_dist, method = "average")

# Assign cluster labels to the data
final_data_visualization <- main_data
final_data_visualization$cluster <- as.factor(cutree(hc_complete, k = 4))  # Cut tree in

final_data_visualization_c = final_data_visualization

# Visualize the dendrogram
fviz_dend(hc_complete, k = 4, rect = TRUE, cex = 0.8, horiz = FALSE) +
  labs(title = "Hierarchical Clustering (k=4) using Average Linkage and Gower Distance")
  theme_minimal()
```




```
# Create boxplots for numeric variables and a bar graph for gender variables
variables <- colnames(final_data_visualization)[-ncol(final_data_visualization)]  # Excl


for (var in variables) {
  if (var %in% c("gender")) {
    print("Doing nothing for gender variables in this loop")
  } else {
    # Create boxplot for numeric variables
    plot <- ggplot(final_data_visualization, aes_string(x = "cluster", y = var, fill = "
```

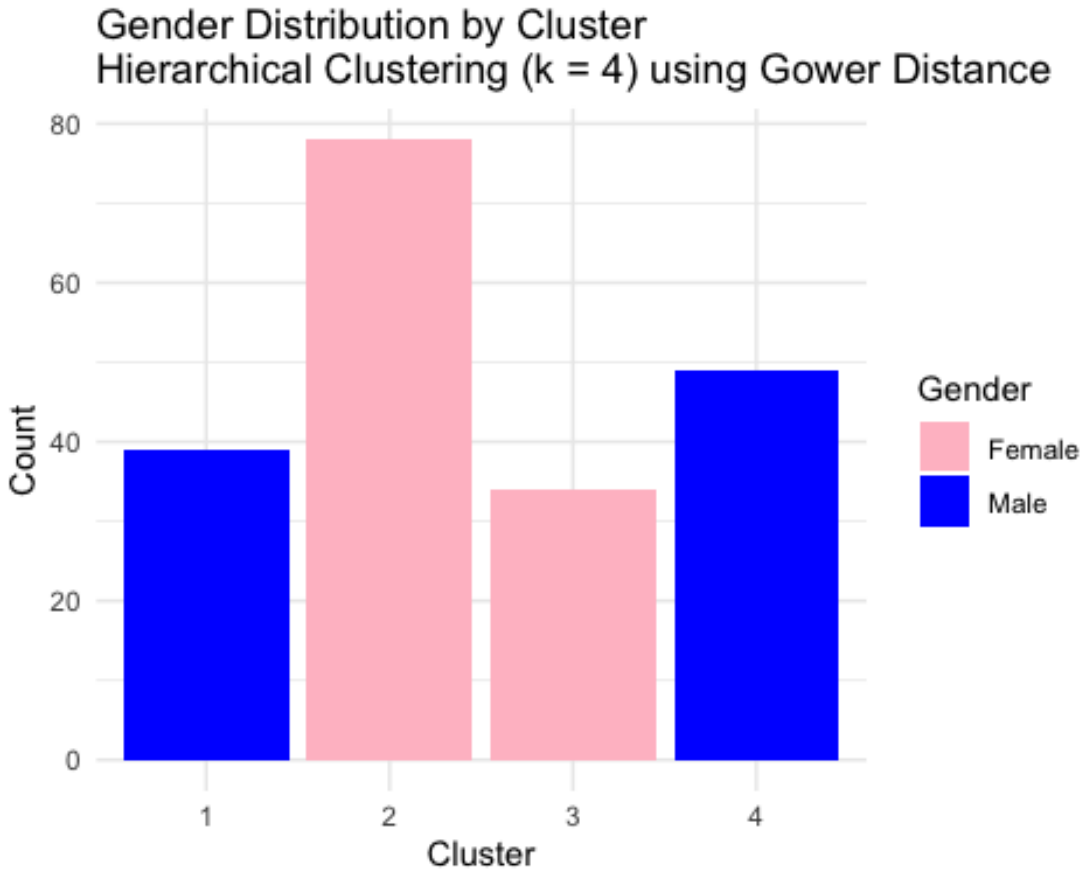
```
      geom_boxplot() +
      labs(title = paste("Boxplot of", var, "by Cluster using Hierarchical Clustering (k
           x = "Cluster",
           y = var) +
      theme_minimal()
    print(plot)
  }
}
##[1] "Doing nothing for gender variables in this loop"
```






```
# Load necessary library

library(tidyverse)


# Summarize gender counts for each cluster

# Group data by cluster and gender, then calculate the number of observations

gender_counts <- final_data_visualization_c %>%

  group_by(cluster, gender) %>%

  summarise(Count = n(), .groups = "drop") # Count the number of rows in each group
```

```
# Create a side-by-side bar graph to visualize gender distribution by cluster
plot <- ggplot(gender_counts, aes(x = cluster, y = Count, fill = gender)) +
  geom_bar(stat = "identity", position = "dodge") + # Side-by-side bars for better compa
  scale_fill_manual(values = c("Male" = "blue", "Female" = "pink")) + # Assign colors to
  labs(title = "Gender Distribution by Cluster\nHierarchical Clustering (k = 4) using Go
       x = "Cluster", # Label for the X-axis
       y = "Count", # Label for the Y-axis
       fill = "Gender") + # Title for the legend
  theme_minimal()

# Display the bar graph
print(plot)
```




```
# Print the gender counts for each cluster to inspect the data

print(gender_counts)

## # A tibble: 4 × 3

##   cluster gender Count

##   <fct>   <fct>  <int>

## 1 1       Male      39

## 2 2       Female    78

## 3 3       Female    34

## 4 4       Male      49
```

APPENDIX G

Medical Insurance Cost Data Exploration

The dataseet contains the following variables:

- Age: Age of the individual (Quantitative)
- Sex: Gender of the individual (Male/Female) (Qualitative)
- BMI: Body Mass Index of an individual (Quantitative)
- Children: Number of children an individual has (Quantitative)
- Smoker Status: yes/no on whether an individual regularly smoke's (Qualitative)
- Region: Region that an indvidual resides (northwest, northeast, southwest, southeast) (Qualitative)
- Charges (target variable): Cost of an individual's medical insurance (Quantitative)

```
## Loading required package:  ggplot2

## -- Attaching core tidyverse packages ----------------------- tidyverse 2.0.0
--

## v dplyr     1.1.4     v stringr   1.5.1

## v forcats   1.0.0     v tibble    3.2.1

## v lubridate 1.9.3     v tidyr     1.3.1
```

```
## v purrr     1.0.2

## -- Conflicts ---------------------------------------- tidyverse_conflicts()

--

## x dplyr::filter() masks stats::filter()

## x dplyr::lag()    masks stats::lag()

## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all

conflicts to become errors

## Loading required package:  lattice

##

##

## Attaching package:  'caret'

##

##

## The following object is masked from 'package:purrr':

##

##     lift
```

```
##  $ smoker  : chr  "yes" "no" "no" "no" ...
##  $ region  : chr  "southwest" "southeast" "southeast" "northwest" ...
##  $ expenses: num  16885 1726 4449 21984 3867 ...
##   age    sex  bmi children smoker    region expenses
## 1  19 female 27.9        0    yes southwest 16884.92
## 2  18   male 33.8        1     no southeast  1725.55
## 3  28   male 33.0        3     no southeast  4449.46
## 4  33   male 22.7        0     no northwest 21984.47
## 5  32   male 28.9        0     no northwest  3866.86
## 6  31 female 25.7        0     no southeast  3756.62
## [1] 0
##       age        sex          bmi            children     smoker
##  Min.   :18.00   1:676   Min.   :16.00   Min.   :0.000   no :1064
##  1st Qu.:27.00   0:662   1st Qu.:26.30   1st Qu.:0.000   yes: 274
##  Median :39.00           Median :30.40   Median :1.000
##  Mean   :39.21           Mean   :30.67   Mean   :1.095
##  3rd Qu.:51.00           3rd Qu.:34.70   3rd Qu.:2.000
##  Max.   :64.00           Max.   :53.10   Max.   :5.000
##        region       expenses
##  northeast:324   Min.   : 1122
##  northwest:325   1st Qu.: 4740
```

```
##  southeast:364   Median : 9382
##  southwest:325   Mean   :13270
##                  3rd Qu.:16640
##                  Max.   :63770
```

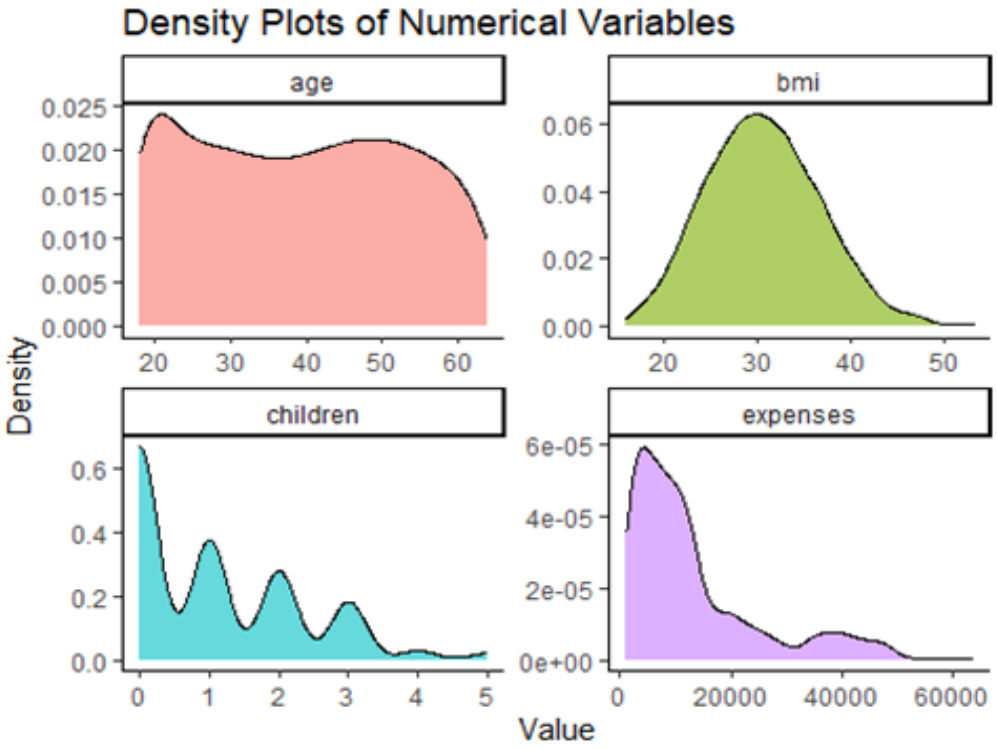


Fig 1: Medical Cost Data Numeric Variable Densities

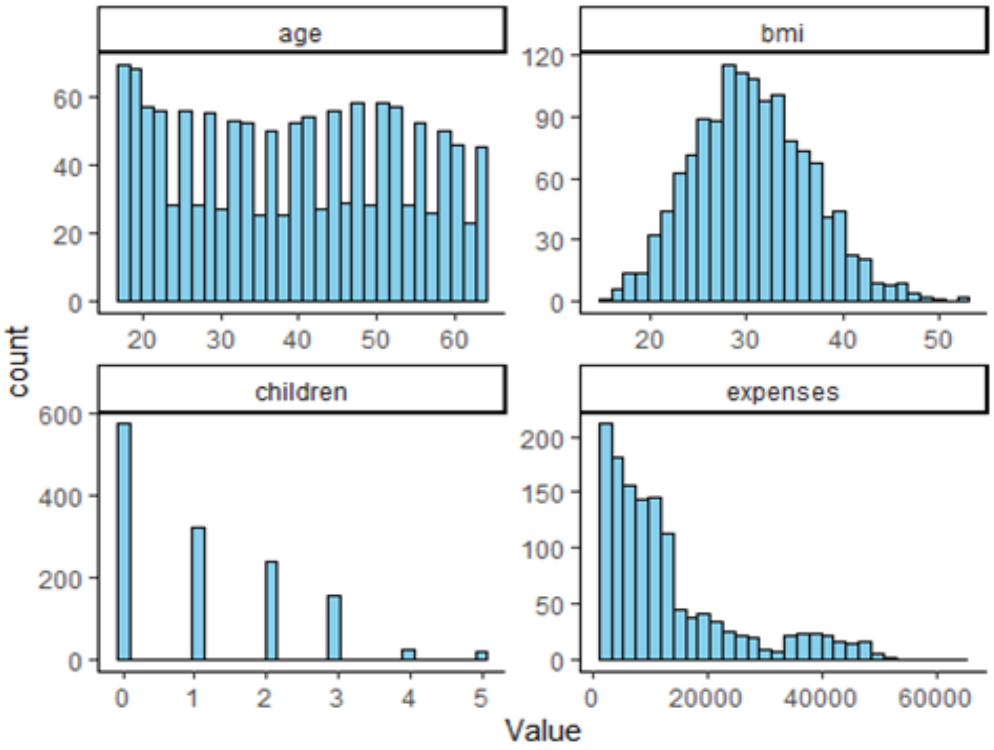


Fig 1: Medical Cost Data Numeric Variable Histograms

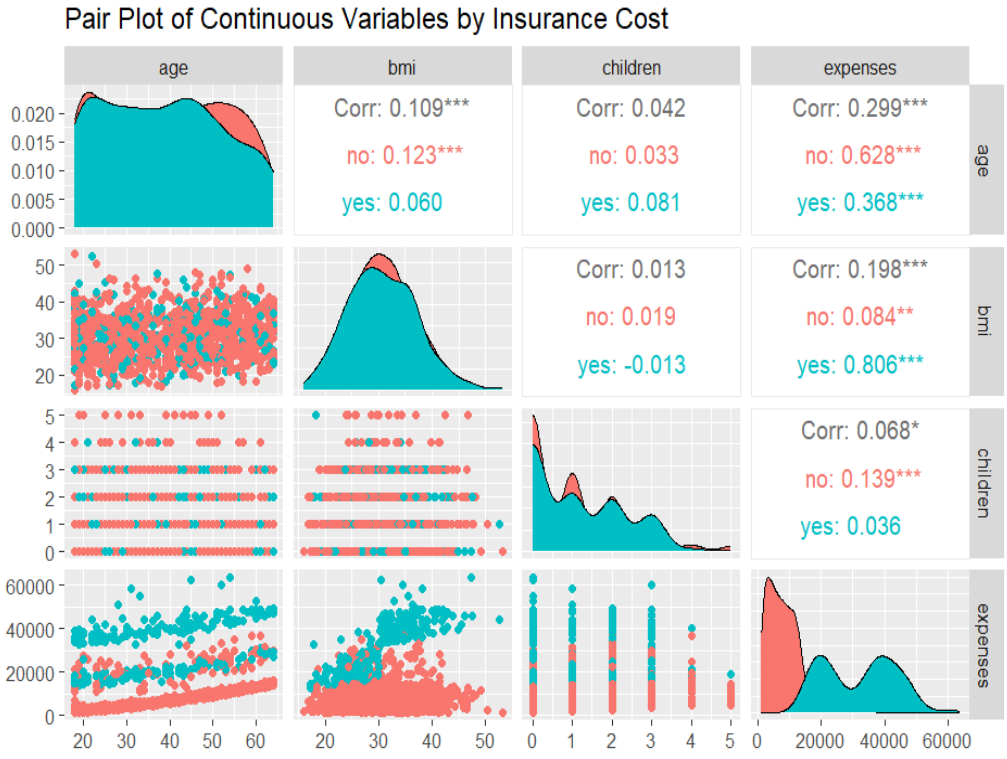


Fig 1: Medical Cost Data Numeric Pair Plot

APPENDIX H

Applying Each Statistical Learning Method

Multiple Linear Regression Model

```
## The following objects are masked from insurance (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
##
## The following objects are masked from insurance (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker

##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.92356 -0.24592 -0.08366  0.13076  2.47851
```

```
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.36707    0.03715  -9.881  < 2e-16 ***
## age              0.28284    0.01646  17.182  < 2e-16 ***
## sex0             0.02439    0.03247   0.751 0.452785
## bmi              0.18281    0.01695  10.783  < 2e-16 ***
## children         0.06056    0.01605   3.772 0.000171 ***
## smokeryes        1.98194    0.03949  50.189  < 2e-16 ***
## regionnorthwest -0.06328    0.04602  -1.375 0.169484
## regionsoutheast -0.07417    0.04650  -1.595 0.111062
## regionsouthwest -0.08523    0.04637  -1.838 0.066322 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5098 on 994 degrees of freedom
## Multiple R-squared:  0.7495, Adjusted R-squared:  0.7474
## F-statistic: 371.7 on 8 and 994 DF,  p-value: < 2.2e-16
```

Ridge Regression

```
## The following objects are masked from standardized_data:
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
## glmnet
##
## 1003 samples
##    6 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 901, 902, 903, 903, 903, 903, ...
## Resampling results across tuning parameters:
##
##   lambda      RMSE       Rsquared   MAE
##   0.00000000  0.5135750  0.7468254  0.3638328
##   0.01111111  0.5135750  0.7468254  0.3638328
##   0.02222222  0.5135750  0.7468254  0.3638328
##   0.03333333  0.5135750  0.7468254  0.3638328
##   0.04444444  0.5135750  0.7468254  0.3638328
##   0.05555556  0.5135750  0.7468254  0.3638328
##   0.06666667  0.5135750  0.7468254  0.3638328
##   0.07777778  0.5135750  0.7468254  0.3638328
##   0.08888889  0.5145348  0.7468086  0.3650187
##   0.10000000  0.5157486  0.7467919  0.3665509
```

```
##
## Tuning parameter 'alpha' was held constant at a value of 0
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were alpha = 0 and lambda = 0.07777778.
```

Lasso Model

```
## The following objects are masked from standardized_data (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 3):
```

```
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 901, 902, 903, 903, 903, 903, ...
## Resampling results across tuning parameters:
##
##   lambda      RMSE       Rsquared   MAE
##   0.00000000  0.5094412  0.7467932  0.3552108
##   0.01111111  0.5095888  0.7468890  0.3543900
##   0.02222222  0.5100988  0.7471238  0.3538109
##   0.03333333  0.5122851  0.7461917  0.3552571
##   0.04444444  0.5155790  0.7445914  0.3580541
##   0.05555556  0.5195977  0.7426245  0.3622056
##   0.06666667  0.5236682  0.7409630  0.3664994
##   0.07777778  0.5280760  0.7392842  0.3710620
##   0.08888889  0.5331523  0.7372027  0.3763201
##   0.10000000  0.5388386  0.7347019  0.3821629
##
## Tuning parameter 'alpha' was held constant at a value of 1
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were alpha = 1 and lambda = 0.
```

Elastic Net Model

```
## The following objects are masked from standardized_data (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 9):
##
```

```
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 9):
##
```

```
##     age, bmi, children, expenses, region, sex, smoker

## The following objects are masked from insurance (pos = 10):

##

##     age, bmi, children, expenses, region, sex, smoker
```

```
##   0.0000000  0.06666667  0.5135750  0.7468254  0.3638328
##   0.0000000  0.07777778  0.5135750  0.7468254  0.3638328
##   0.0000000  0.08888889  0.5145348  0.7468086  0.3650187
##   0.0000000  0.10000000  0.5157486  0.7467919  0.3665509
##   0.1111111  0.00000000  0.5094323  0.7468293  0.3557492
##   0.1111111  0.01111111  0.5095531  0.7468045  0.3560988
##   0.1111111  0.02222222  0.5099729  0.7467166  0.3569459
##   0.1111111  0.03333333  0.5106231  0.7465867  0.3579620
##   0.1111111  0.04444444  0.5114021  0.7464887  0.3590594
##   0.1111111  0.05555556  0.5122105  0.7465464  0.3602211
##   0.1111111  0.06666667  0.5131330  0.7466465  0.3615851
##   0.1111111  0.07777778  0.5141498  0.7467969  0.3630301
##   0.1111111  0.08888889  0.5153302  0.7469222  0.3645291
##   0.1111111  0.10000000  0.5166901  0.7469913  0.3661724
##   0.2222222  0.00000000  0.5094380  0.7468149  0.3555673
##   0.2222222  0.01111111  0.5096517  0.7467233  0.3558864
##   0.2222222  0.02222222  0.5102947  0.7464569  0.3567286
##   0.2222222  0.03333333  0.5107884  0.7465673  0.3576305
##   0.2222222  0.04444444  0.5113105  0.7468398  0.3584753
##   0.2222222  0.05555556  0.5120777  0.7470605  0.3595284
##   0.2222222  0.06666667  0.5131473  0.7471708  0.3608505
```

```
##   0.2222222  0.07777778  0.5144673  0.7472143  0.3624050
##   0.2222222  0.08888889  0.5159828  0.7472271  0.3640813
##   0.2222222  0.10000000  0.5177469  0.7471281  0.3659852
##   0.3333333  0.00000000  0.5094418  0.7468045  0.3554441
##   0.3333333  0.01111111  0.5097864  0.7466073  0.3557072
##   0.3333333  0.02222222  0.5102815  0.7465461  0.3564406
##   0.3333333  0.03333333  0.5105373  0.7469417  0.3568644
##   0.3333333  0.04444444  0.5111822  0.7471999  0.3576492
##   0.3333333  0.05555556  0.5122633  0.7472664  0.3588918
##   0.3333333  0.06666667  0.5137613  0.7471287  0.3605363
##   0.3333333  0.07777778  0.5155321  0.7469048  0.3623533
##   0.3333333  0.08888889  0.5175294  0.7466310  0.3644644
##   0.3333333  0.10000000  0.5197709  0.7462814  0.3668982
##   0.4444444  0.00000000  0.5094556  0.7467929  0.3553749
##   0.4444444  0.01111111  0.5099534  0.7464606  0.3556033
##   0.4444444  0.02222222  0.5100996  0.7467805  0.3559220
##   0.4444444  0.03333333  0.5103983  0.7472129  0.3561209
##   0.4444444  0.04444444  0.5114077  0.7472433  0.3571562
##   0.4444444  0.05555556  0.5129610  0.7469813  0.3587630
##   0.4444444  0.06666667  0.5148409  0.7466209  0.3605897
##   0.4444444  0.07777778  0.5170985  0.7461061  0.3629184
```

```
##   0.4444444  0.08888889  0.5196920  0.7454554  0.3657304
##   0.4444444  0.10000000  0.5225599  0.7447026  0.3688171
##   0.5555556  0.00000000  0.5094472  0.7467918  0.3553197
##   0.5555556  0.01111111  0.5099689  0.7464667  0.3554531
##   0.5555556  0.02222222  0.5098747  0.7470492  0.3552664
##   0.5555556  0.03333333  0.5104808  0.7472862  0.3556580
##   0.5555556  0.04444444  0.5119496  0.7469786  0.3569893
##   0.5555556  0.05555556  0.5138539  0.7465073  0.3588041
##   0.5555556  0.06666667  0.5163026  0.7457668  0.3612092
##   0.5555556  0.07777778  0.5191376  0.7448649  0.3642339
##   0.5555556  0.08888889  0.5223372  0.7437968  0.3676301
##   0.5555556  0.10000000  0.5257233  0.7427370  0.3711731
##   0.6666667  0.00000000  0.5094406  0.7467995  0.3552731
##   0.6666667  0.01111111  0.5099326  0.7465250  0.3552874
##   0.6666667  0.02222222  0.5097595  0.7472258  0.3547104
##   0.6666667  0.03333333  0.5108083  0.7471256  0.3554070
##   0.6666667  0.04444444  0.5126047  0.7466083  0.3569725
##   0.6666667  0.05555556  0.5150678  0.7457484  0.3592090
##   0.6666667  0.06666667  0.5180508  0.7446383  0.3623015
##   0.6666667  0.07777778  0.5214547  0.7433437  0.3659023
##   0.6666667  0.08888889  0.5250963  0.7420407  0.3697546
```

```
##   0.6666667  0.10000000  0.5286903  0.7410181  0.3736006
##   0.7777778  0.00000000  0.5094429  0.7467934  0.3552456
##   0.7777778  0.01111111  0.5098549  0.7466143  0.3550576
##   0.7777778  0.02222222  0.5097454  0.7473094  0.3542790
##   0.7777778  0.03333333  0.5112133  0.7468913  0.3552350
##   0.7777778  0.04444444  0.5134675  0.7460556  0.3571549
##   0.7777778  0.05555556  0.5164567  0.7448212  0.3599809
##   0.7777778  0.06666667  0.5199953  0.7433164  0.3636799
##   0.7777778  0.07777778  0.5237845  0.7418326  0.3676998
##   0.7777778  0.08888889  0.5275329  0.7406603  0.3717294
##   0.7777778  0.10000000  0.5316731  0.7393239  0.3760372
##   0.8888889  0.00000000  0.5094419  0.7467933  0.3552260
##   0.8888889  0.01111111  0.5097256  0.7467500  0.3547432
##   0.8888889  0.02222222  0.5098923  0.7472460  0.3540018
##   0.8888889  0.03333333  0.5116856  0.7465967  0.3551649
##   0.8888889  0.04444444  0.5144678  0.7453768  0.3575024
##   0.8888889  0.05555556  0.5180190  0.7437215  0.3609767
##   0.8888889  0.06666667  0.5219614  0.7419932  0.3651871
##   0.8888889  0.07777778  0.5258372  0.7406420  0.3693430
##   0.8888889  0.08888889  0.5301829  0.7390904  0.3738575
##   0.8888889  0.10000000  0.5350674  0.7372239  0.3788642
```

```
##   1.0000000  0.00000000  0.5094412  0.7467932  0.3552108
##   1.0000000  0.01111111  0.5095888  0.7468890  0.3543900
##   1.0000000  0.02222222  0.5100988  0.7471238  0.3538109
##   1.0000000  0.03333333  0.5122851  0.7461917  0.3552571
##   1.0000000  0.04444444  0.5155790  0.7445914  0.3580541
##   1.0000000  0.05555556  0.5195977  0.7426245  0.3622056
##   1.0000000  0.06666667  0.5236682  0.7409630  0.3664994
##   1.0000000  0.07777778  0.5280760  0.7392842  0.3710620
##   1.0000000  0.08888889  0.5331523  0.7372027  0.3763201
##   1.0000000  0.10000000  0.5388386  0.7347019  0.3821629
##
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were alpha = 0.1111111 and lambda = 0.
```

Subset Selection Model

```
## The following objects are masked from standardized_data (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 5):
```

```
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 10):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 3):
```

```
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 10):
```

```
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 12):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
##   4      0.5078715  0.74820938  0.3541606

##

## RMSE was used to select the optimal model using the smallest value.

## The final value used for the model was nvmax = 4.

## Subset selection object

## 8 Variables  (and intercept)

##                 Forced in Forced out

## age                 FALSE      FALSE

## sex0                FALSE      FALSE

## bmi                 FALSE      FALSE

## children            FALSE      FALSE

## smokeryes           FALSE      FALSE

## regionnorthwest     FALSE      FALSE

## regionsoutheast     FALSE      FALSE

## regionsouthwest     FALSE      FALSE

## 1 subsets of each size up to 4

## Selection Algorithm: 'sequential replacement'

##          age sex0 bmi children smokeryes regionnorthwest regionsoutheast

## 1  ( 1 ) " " " " " "  " " " " "      "*"       " "             " "

## 2  ( 1 ) "*" "*"  " " " " "      " "       " "             " "

## 3  ( 1 ) "*" " "  "*" " " "      "*"       " "             " "
```

```
## 4  ( 1 ) "*" " "  "*" "*"      "*"       " "              " "
##          regionsouthwest
## 1  ( 1 ) " "
## 2  ( 1 ) " "
## 3  ( 1 ) " "
## 4  ( 1 ) " "
```

PCR Model

```
## The following objects are masked from standardized_data (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 7):
##
```

```
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 10):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 12):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 13):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 3):
##
```

```
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 10):
##
```

```
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 12):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 13):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 14):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
## Resampling results across tuning parameters:
##
##   ncomp  RMSE       Rsquared   MAE
##   1      0.9583059  0.1073564  0.7595989
##   2      0.9612955  0.1036890  0.7630582
##   3      0.9592957  0.1077790  0.7644768
##   4      0.9492696  0.1270373  0.7451962
##   5      0.9406573  0.1423727  0.7273278
##   6      0.9433969  0.1380006  0.7292832
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was ncomp = 5.
## Data:  X dimension: 1003 8
##  Y dimension: 1003 1
## Fit method: svdpc
## Number of components considered: 5
## TRAINING: % variance explained
##           1 comps  2 comps  3 comps  4 comps  5 comps
## X           28.00    53.46    75.89    82.22    88.47
## .outcome    10.18    10.22    10.68    13.17    14.84
##                  Model      RMSE       MAE
```

```
## 1    Linear Regression 0.5093945 0.3554555
## 2                Ridge 0.5135750 0.3638328
## 3                Lasso 0.5123621 0.3560860
## 4          Elastic Net 0.5094323 0.3557492
## 5 Subset \n  Selection 0.5089586 0.3542164
## 6                  PCR 0.9406573 0.7292832
```

Model	RMSE	MAE
Linear Regression	0.5093945	0.3554555
Ridge	0.5135750	0.3638328
Lasso	0.5123621	0.3560860
Elastic Net	0.5094323	0.3557492
Subset Selection	0.5089586	0.3542164
PCR	0.9406573	0.7292832

APPENDIX I

Apply Methods to Insurance Cost Test Data

```
## The following objects are masked from standardized_data (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
## The following objects are masked from standardized_data (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 10):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 12):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 13):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 14):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 15):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
## The following objects are masked from insurance (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
## The following objects are masked from standardized_data (pos = 10):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 12):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 13):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 14):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 15):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 16):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
##                   Model Test_RMSE  Test_MAE
## 1    Linear Regression 0.4759808 0.3303843
## 2                 Ridge 0.4778313 0.3398511
## 3                 Lasso 0.4746840 0.3289926
## 4           Elastic Net 0.4756728 0.3305701
## 5 Subset \n  Selection 0.4760174 0.3301431
## 6                   PCR 0.8537596 0.6633826
```

Model	Test_RMSE	Test_MAE
Linear Regression	0.4759808	0.3303843
Ridge	0.4778313	0.3398511
Lasso	0.4746840	0.3289926
Elastic Net	0.4756728	0.3305701
Subset Selection	0.4760174	0.3301431
PCR	0.8537596	0.6633826

APPENDIX J

Final Model Decision and Results

```
## The following objects are masked from standardized_data (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
## The following objects are masked from standardized_data (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 10):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 12):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 13):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 14):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 15):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
## The following objects are masked from insurance (pos = 16):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 17):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
## The following objects are masked from standardized_data (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 10):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 12):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 13):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 14):
##
##     age, bmi, children, expenses, region, sex, smoker
```

```
## The following objects are masked from insurance (pos = 15):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 16):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 17):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 18):
##
##     age, bmi, children, expenses, region, sex, smoker
## glmnet
##
## 1338 samples
##    6 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1204, 1205, 1203, 1203, 1204, 1206, ...
## Resampling results across tuning parameters:
```

```
##
##   lambda      RMSE       Rsquared   MAE
##   0.00000000  0.5001185  0.7519482  0.3465158
##   0.01111111  0.5004022  0.7518251  0.3461569
##   0.02222222  0.5016541  0.7514147  0.3465395
##   0.03333333  0.5039351  0.7504498  0.3485089
##   0.04444444  0.5071123  0.7489974  0.3518130
##   0.05555556  0.5105043  0.7477464  0.3555191
##   0.06666667  0.5143049  0.7464248  0.3596476
##   0.07777778  0.5187544  0.7447582  0.3644586
##   0.08888889  0.5238351  0.7427108  0.3697657
##   0.10000000  0.5295291  0.7402413  0.3755733
##
## Tuning parameter 'alpha' was held constant at a value of 1
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were alpha = 1 and lambda = 0.
```

APPENDIX K

Code for Regression Analysis

```
set.seed(123)

library(readr)

library(readxl)

library(ggpubr)

library(tidyverse)

library(dplyr)

library(ggplot2)

library(caret)

library(rsample)

library(patchwork)



insurance=read.csv("insurance.csv")

str(insurance)

head(insurance)

sum(is.na(insurance))

attach(insurance)

## The following objects are masked from standardized_data (pos = 3):

##
```

```
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 6):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 10):
##
```

```
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 12):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 13):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 14):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 15):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 16):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 17):
##
```

```
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 18):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 19):
##
##     age, bmi, children, expenses, region, sex, smoker

insurance = insurance %>%
  mutate(sex = factor(sex, levels = c("male", "female"),
          labels = c(1, 0)), smoker = factor(smoker),
          region = factor(region,levels = c("northeast", "northwest",
          "southeast", "southwest")))


summary(insurance)
insurance %>%
  select(age, bmi, children, expenses) %>%
  pivot_longer(everything(), names_to = "Variable", values_to = "Value") %>%
  ggplot(aes(x = Value)) +
  geom_histogram(bins = 10, fill = "skyblue", color = "black") +
  facet_wrap(~ Variable, scales = "free") +
  theme_classic()
```

```
insurance %>%
  select(age, bmi, children, expenses) %>%
  summary()


#Plots
ggplot(insurance, aes(x = bmi, y = expenses)) +
  geom_point(aes(color = smoker), alpha = 0.6) +
  labs(title = "Insurance Costs vs. BMI", x = "BMI", y = "Expenses") +
  theme_classic()
ggplot(insurance, aes(x = age, y = expenses)) +
  geom_point(aes(color = smoker), alpha = 0.6) +
  labs(title = "Insurance Costs vs. Age", x = "Age", y = "Expenses") +
  theme_classic()
#Correlation Matrix
correlation_matrix <- cor(insurance %>% select(age, bmi, children, expenses))
ggcorrplot::ggcorrplot(correlation_matrix, lab =T)
# Density Plot
insurance %>%
  select(age, bmi, children, expenses) %>%
  pivot_longer(cols = everything(), names_to = "Variable", values_to =
  "Value") %>%
```

```
  ggplot(aes(x = Value, fill = Variable)) +
  geom_density(alpha = 0.6) +
  facet_wrap(~ Variable, scales = "free") +
  labs(title = "Density Plots of Numerical Variables", x = "Value", y =
  "Density") +
  theme_classic() +
  theme(legend.position = "none")
preproc <- preProcess(insurance %>% select(age, bmi, children, expenses),
                      method = c("center", "scale"))
standardized_data <- predict(preproc, insurance)
attach(standardized_data)

## The following objects are masked from insurance (pos = 3):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 4):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 5):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 6):
```

```
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 7):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 8):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 9):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 10):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 11):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 12):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 13):
```

```
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 14):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 15):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 16):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 17):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from standardized_data (pos = 18):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 19):
##
##     age, bmi, children, expenses, region, sex, smoker
## The following objects are masked from insurance (pos = 20):
```

```
##
##     age, bmi, children, expenses, region, sex, smoker


split = initial_split(standardized_data, prop = 0.75)

train_data = training(split)

test_data = testing(split)

train_control <- trainControl(method = "cv", number = 10,

                              savePredictions = "final",

                              summaryFunction = defaultSummary)


# Linear regression model

linear_model <- train(expenses ~ ., data = train_data, method = "lm",

                      trControl = train_control)

linear_rmse <- linear_model$results$RMSE[1]

linear_mae <- linear_model$results$MAE[1]


# Ridge regression model

ridge_model <- train(expenses ~ ., data = train_data, method = "glmnet",

                     trControl = train_control,

                     tuneGrid = expand.grid(alpha = 0, lambda = seq(0, 0.1,

                     length = 10)))

ridge_rmse <- 0.5135750
```

```
ridge_mae <- 0.3638328


# Lasso regression model
lasso_model <- train(expenses ~ ., data = train_data, method = "glmnet",
                     trControl = train_control,
                     tuneGrid = expand.grid(alpha = 1, lambda = seq(0, 0.1,
                     length = 10)))
lasso_rmse <- 0.5094412
lasso_mae <- 0.3552108


# Elastic Net Model
elastic_model <- train(expenses ~ ., data = train_data, method = "glmnet",
                       trControl = train_control,
                       tuneGrid = expand.grid(alpha = seq(0, 1,
                                              length = 10),
                       lambda = seq(0, 0.1, length = 10)))
elastic_rmse <- 0.5094323
elastic_mae <- 0.3557492


# Subset Selection using Stepwise Regression
subset_model <- train(expenses ~ ., data = train_data, method = "leapSeq",
```

```
                    trControl = train_control)

subset_rmse <- 0.5078715

subset_mae <- 0.3541606


# PCR Model

pcr_model <- train(expenses ~ ., data = train_data, method = "pcr",

                   trControl = train_control,

                   tuneGrid = data.frame(ncomp = seq(1,ncol(train_data)-1,

                   by = 1)))


# Performance for PCR

pcr_rmse <- 0.9406573

pcr_mae <- 0.7292832


# Model Comparison

model_comparison <- data.frame(

  Model = c("Linear Regression", "Ridge", "Lasso", "Elastic Net", "Subset

  Selection", "PCR"),

  RMSE = c(linear_rmse, ridge_rmse, lasso_rmse, elastic_rmse, subset_rmse,

  pcr_rmse),

  MAE = c(linear_mae, ridge_mae, lasso_mae, elastic_mae, subset_mae, pcr_mae
```

```
  )
)


# Print comparison table
print(model_comparison)
(table_plot <- ggtexttable(model_comparison,
                           rows = NULL, # Remove row names
                           theme = ttheme("mBlue")))
# Predict on Test Data
linear_preds <- predict(linear_model, newdata = test_data)
ridge_preds <- predict(ridge_model, newdata = test_data)
lasso_preds <- predict(lasso_model, newdata = test_data)
elastic_preds <- predict(elastic_model, newdata = test_data)
subset_preds <- predict(subset_model, newdata = test_data)
pcr_preds <- predict(pcr_model, newdata = test_data)


# Compute RMSE and MAE on Test Data
test_results <- data.frame(
  Model = c("Linear Regression", "Ridge", "Lasso", "Elastic Net", "Subset
  Selection", "PCR"),
  Test_RMSE = c(
```

```
    RMSE(linear_preds, test_data$expenses),
    RMSE(ridge_preds, test_data$expenses),
    RMSE(lasso_preds, test_data$expenses),
    RMSE(elastic_preds, test_data$expenses),
    RMSE(subset_preds, test_data$expenses),
    RMSE(pcr_preds, test_data$expenses)
  ),
  Test_MAE = c(
    MAE(linear_preds, test_data$expenses),
    MAE(ridge_preds, test_data$expenses),
    MAE(lasso_preds, test_data$expenses),
    MAE(elastic_preds, test_data$expenses),
    MAE(subset_preds, test_data$expenses),
    MAE(pcr_preds, test_data$expenses)
  ))


# Print test results
print(test_results)
(table_plot <- ggtexttable(test_results,
                           rows = NULL,
                           theme = ttheme("mBlue")))
```

```r
# Final model Lasso

final_lasso_model <- train(expenses ~ ., data = standardized_data,

                           method = "glmnet", trControl = train_control,

                           tuneGrid = expand.grid(alpha = 1,

                                      lambda = seq(0, 0.1, length = 10)))
```