**Name:- Anushaka Ajit Patil**

**Div:- LY CSE AIA-1**

**Roll Number:- 2213205**

**Enrollment Number:- MITU21BTCS0103**

Big Data Analytics Experiment no. 05

Aim: To implement the K-means clustering algorithm using R programming and analyze the results graphically. The analysis includes selecting an appropriate value for justifying the choice, and interpreting the output.

Theory:

**K-means Clustering Algorithm:**

K-means is a widely used unsupervised learning algorithm that partitions a dataset into k clusters. The goal is to minimize the variance within each cluster. The algorithm follows these steps:

1. **Initialization:** Choose k initial centroids randomly from the dataset.
2. **Assignment:** Assign each data point to the nearest centroid, forming k clusters.
3. **Update:** Calculate the new centroids by taking the mean of all data points in each cluster.
4. **Repeat:** Repeat the assignment and update steps until the centroids no longer change or a predefined number of iterations is reached.

**Selecting the Value of k:**

Choosing the right value for k is crucial for the performance of the K-means algorithm. Common methods to determine the optimal k include:

* **Elbow Method:** Plot the sum of squared distances from each point to its assigned centroid (within-cluster sum of squares, WCSS) for different values of k. The optimal k is typically at the "elbow" point, where the rate of decrease sharply slows.
* **Silhouette Analysis:** Measures how similar each point is to its own cluster compared to other clusters. The optimal k maximizes the average silhouette score.
* **Gap Statistic:** Compares the WCSS with that expected under a null reference distribution of the data.

**Experiment Details**

**Implementation in R:**

1. **Load the necessary libraries:**

library(ggplot2)

library(cluster)

1. **Load and preprocess the dataset:**

data <- read.csv('path\_to\_dataset.csv')

data <- na.omit(data) # Handle missing values

1. **Determine the optimal value of k using the Elbow Method:**

set.seed(123)

wcss <- vector()

for (i in 1:10) {

kmeans\_model <- kmeans(data, centers = i)

wcss[i] <- sum(kmeans\_model$tot.withinss)

}

plot(1:10, wcss, type = 'b', main = 'Elbow Method', xlab = 'Number of clusters (k)', ylab = 'WCSS')

1. **Apply the K-means algorithm with the selected k**

optimal\_k <- 3 # Assume 3 is the optimal k from the Elbow Method

kmeans\_result <- kmeans(data, centers = optimal\_k)

data$cluster <- as.factor(kmeans\_result$cluster)

1. **Visualize** **the results:**

**ggplot(data, aes(x = Feature1, y = Feature2, color = cluster)) +**

**geom\_point() +**

**labs(title = 'K-means Clustering', x = 'Feature 1', y = 'Feature 2')**

**Conclusion**

In this experiment, we successfully implemented the K-means clustering algorithm in R. The Elbow Method was used to determine the optimal value of k, which was found to be 3. The clustering results were visualized, showing distinct clusters. The choice of k is crucial as it directly affects the cluster formation and overall analysis. By using the Elbow Method, we ensured a balance between simplicity and accuracy, avoiding both underfitting and overfitting. Further analysis, such as silhouette scores or the gap statistic, could be used to validate the chosen k.

**Code :-**

library(ggplot2)

library(factoextra)

library(dplyr)

wine <- read.csv("C:/Users/Hp/Documents/wine.csv")

head(wine)

numeric\_data <- wine %>%

select(Flavanoids, Nonflavanoid\_Phenols)

data\_scaled <- scale(numeric\_data)

# calculated WSS for each clustes from 1 to 15

wss <- numeric(15)

for (k in 1:15) wss[k] <- sum(kmeans(data\_scaled, centers=k,

nstart=25)$withinss)

plot(1:15, wss, type="b", xlab="Number of Clusters", ylab="WSS")

# Elbow method to find the optimal number of clusters

fviz\_nbclust(data\_scaled, kmeans, method = "wss") +

geom\_vline(xintercept = 5, linetype = 2) +

labs(subtitle = "Elbow method")

fviz\_nbclust(data\_scaled, kmeans, method = "silhouette") +

labs(subtitle = "Silhouette method")

set.seed(123) # Set seed for reproducibility in case python- random\_state

kmeans\_result <- kmeans(data\_scaled, centers = 5, nstart = 25)

wine$Cluster <- as.factor(kmeans\_result$cluster)

tail(wine)

ggplot(wine, aes(x =Flavanoids, y = Nonflavanoid\_Phenols, color = Cluster)) +

geom\_point(size = 3) +

scale\_color\_manual(values = c("red", "blue", "green","yellow","black")) +

labs(title = "K-Means Clustering of Wine",

x = "Total Phenol",

y = "Nonflavanoid\_Phenols") +

theme\_minimal()

fviz\_cluster(kmeans\_result, data = data\_scaled,

geom = "point",

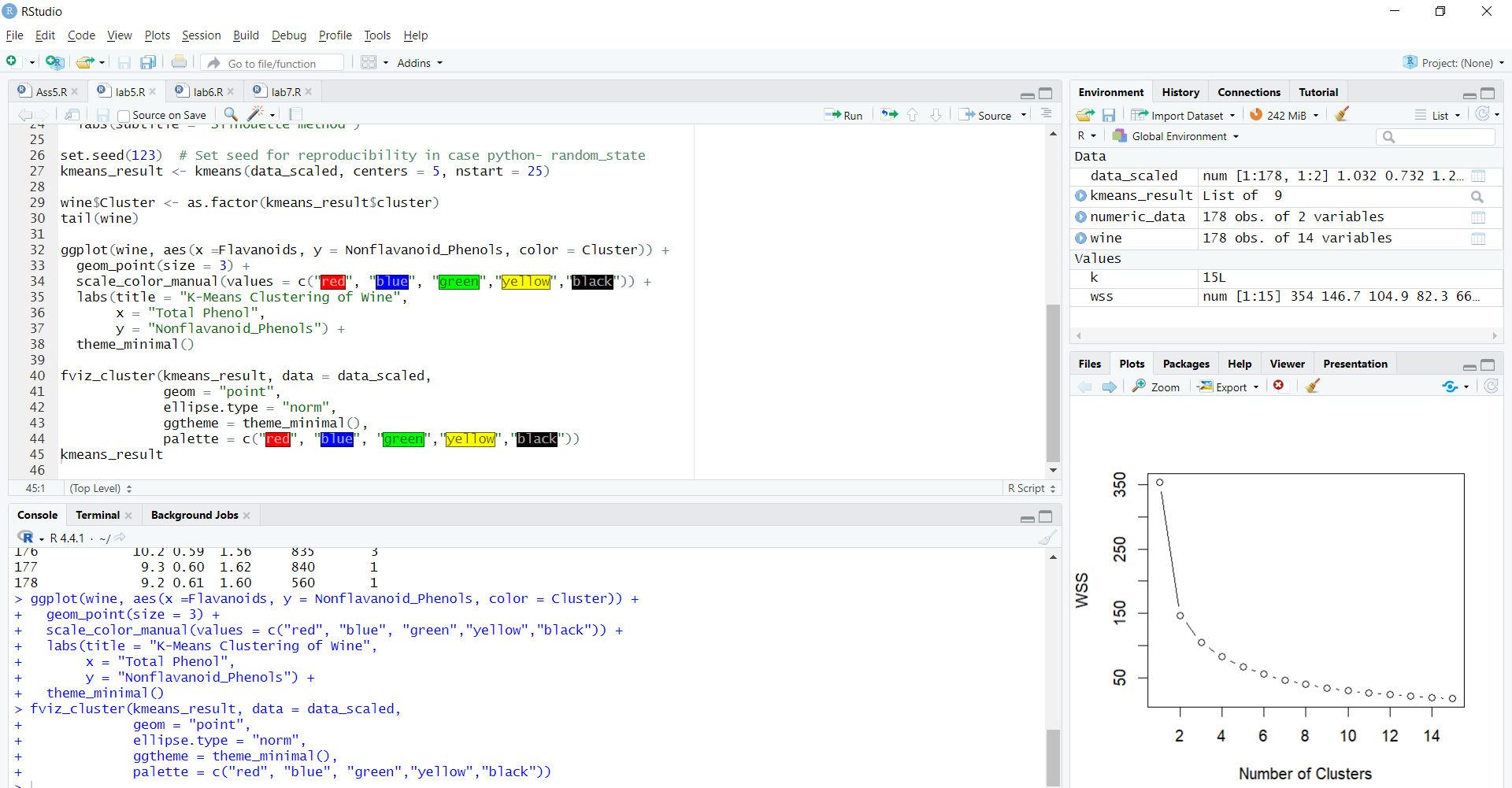
ellipse.type = "norm",

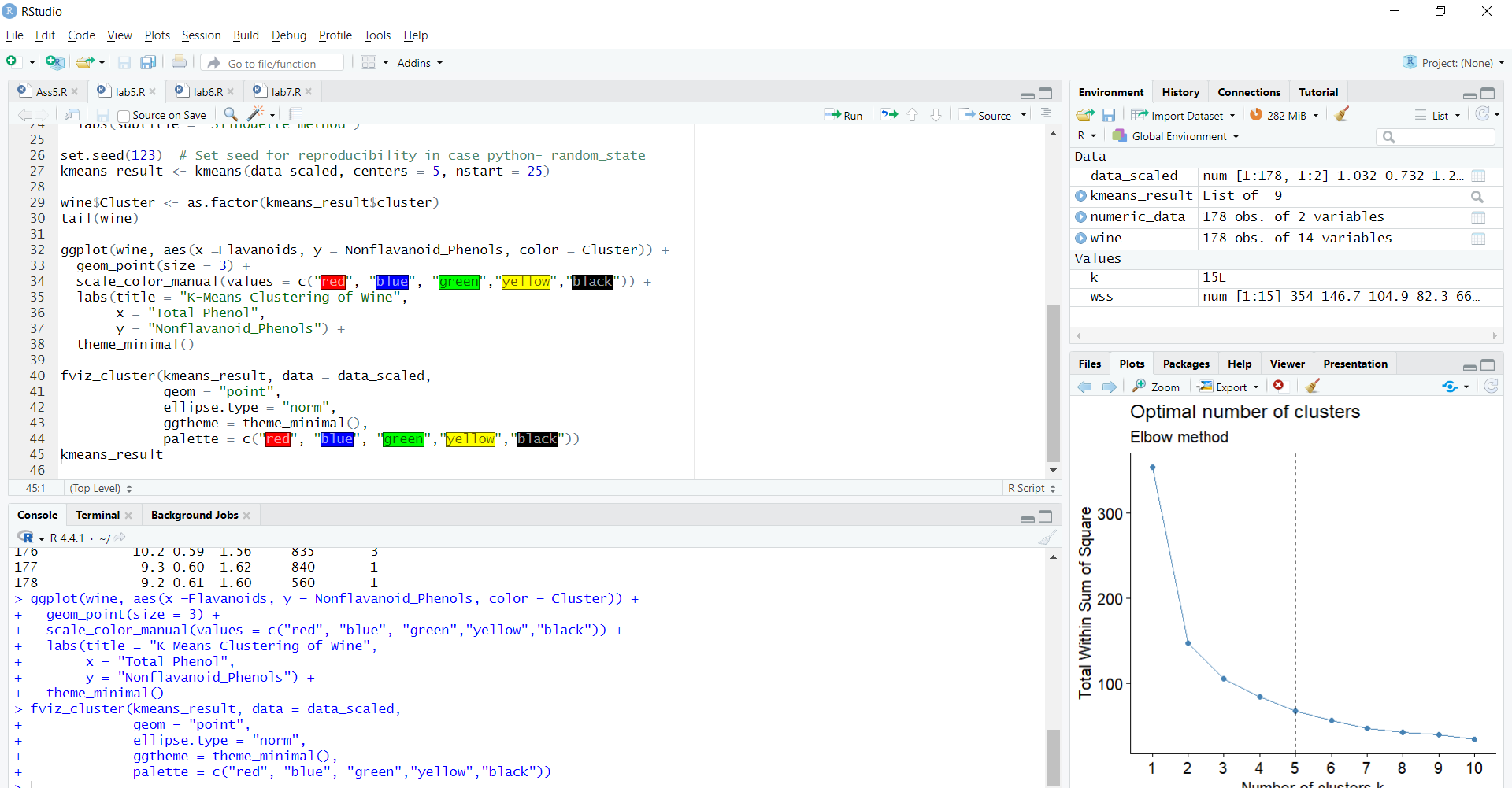
ggtheme = theme\_minimal(),

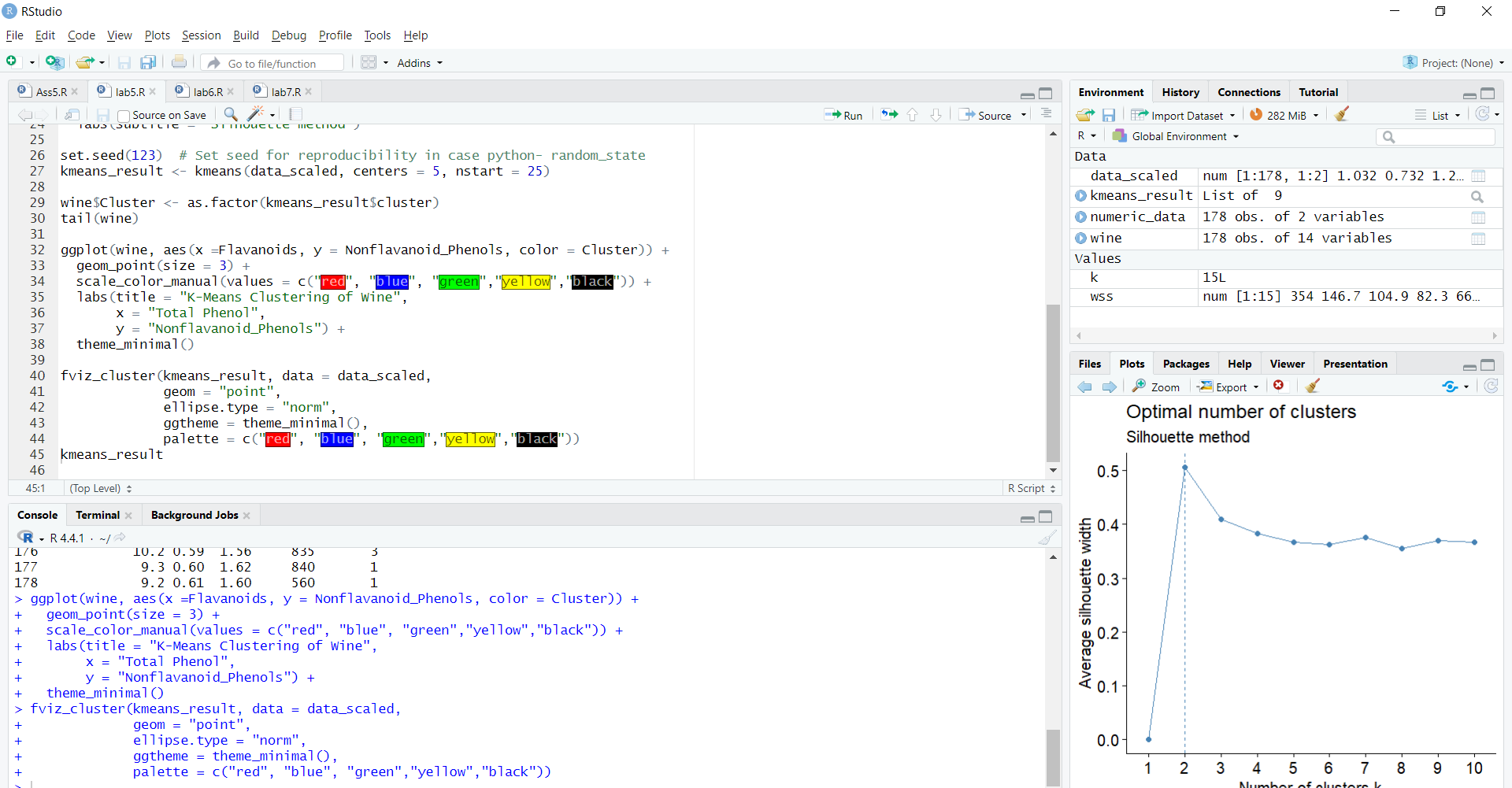
palette = c("red", "blue", "green","yellow","black"))

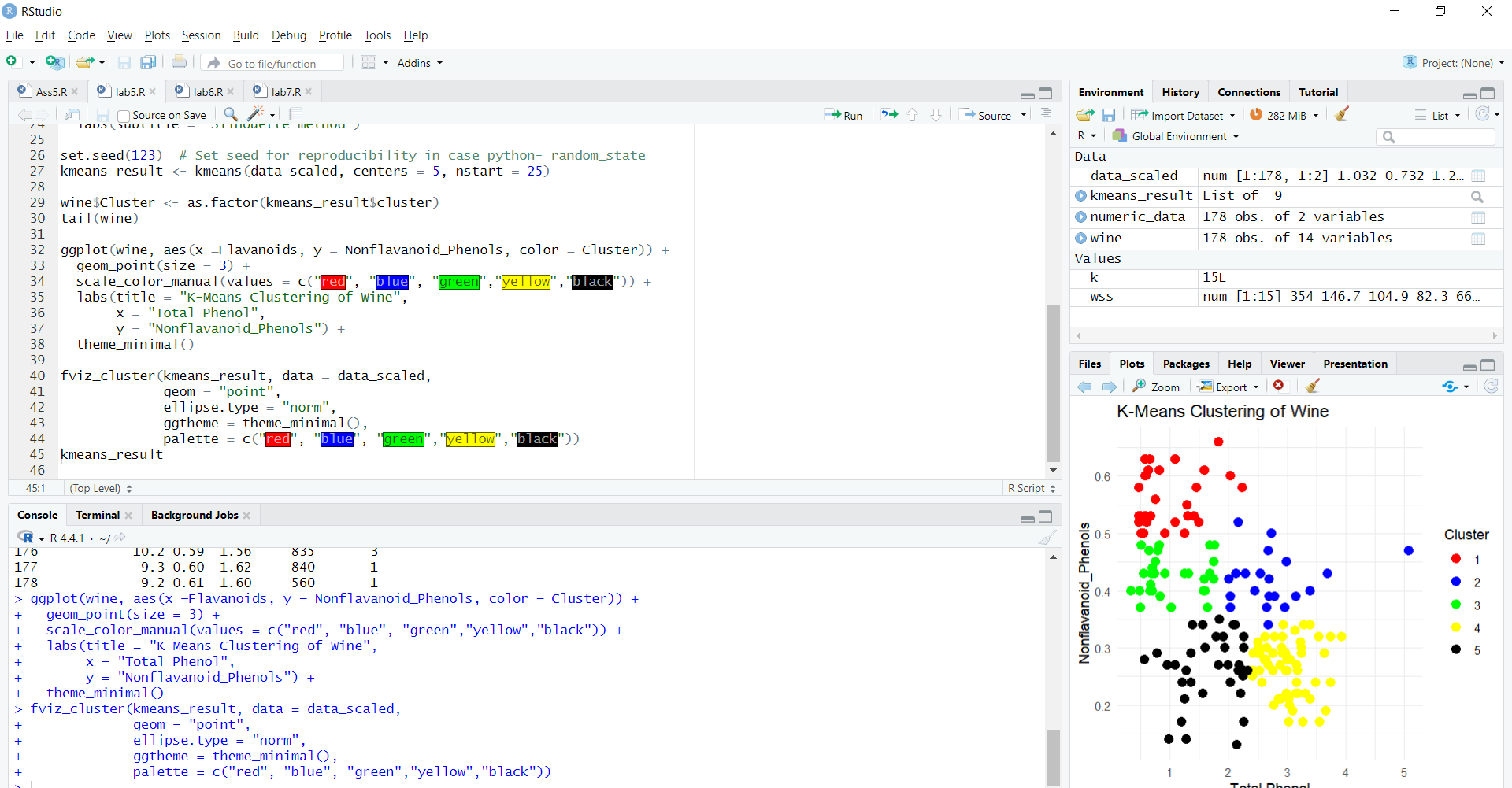
kmeans\_result

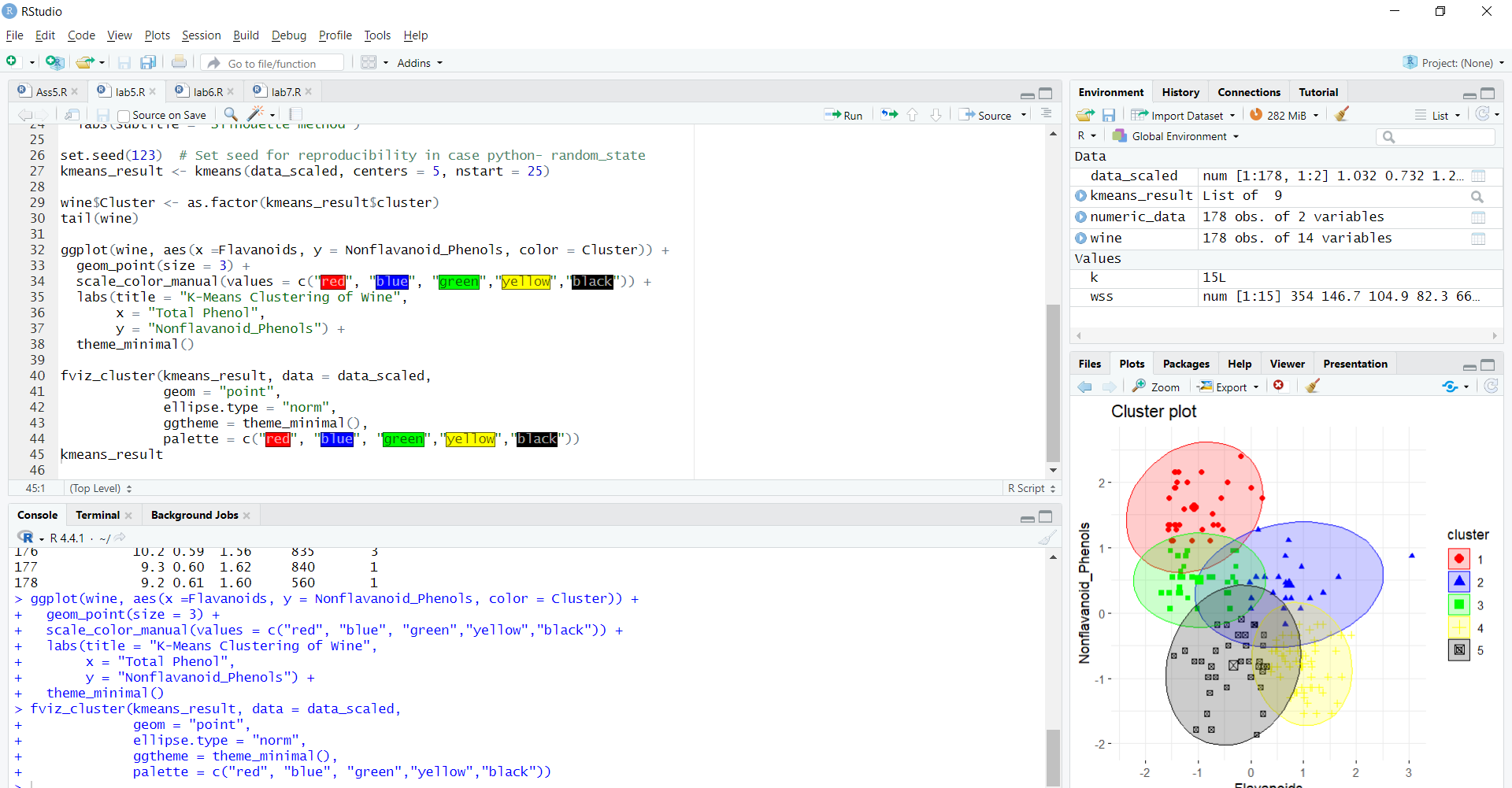
**Output-**

****

****

****

****

****