Part 2: Leveraging Machine Learning and Hyperspectral Imaging for Predicting Adulteration in Ground Arabica Coffee with Robusta: Principal Component Analysis and Clustering

AUTHOR
Derick Malavi

Introduction

Principal Component Analysis (PCA)

PCA reduces dimensionality in hyperspectral data by transforming the original variables into a smaller set of uncorrelated principal components. This process highlights the most significant spectral features driving variance and helps reveal relationships or adulteration patterns, even without prior labels.

PCA was applied to NIR-HSI spectral data from pure and adulterated Arabica coffee samples, reducing dimensionality while preserving critical information. The goal is to visualize the data and prepare it for further analysis using unsupervised methods like K-Means clustering.

K-Means Clustering

K-Means clustering groups data points based on spectral similarities, offering an efficient way to identify clusters in large hyperspectral datasets. The objective is to observe whether distinct clusters for pure and adulterated coffee samples can be detected.

By applying K-Means to the PCA-transformed data, clusters corresponding to pure and adulterated coffee samples are identified based on their spectral features. This unsupervised method provides preliminary insight into the separation between pure and adulterated samples, serving as a foundation for further supervised analysis.

```
#Load libraries

warning = FALSE
  suppressWarnings(suppressMessages({
  library(FactoMineR)
  library(factoextra)
  library(FactoInvestigate)
  library(ggplot2)
  library(dplyr)
  library(readxl)
  library(readr)
  library(janitor)
  library(mdatools)
  library(rmarkdown)
  library(knitr)
  library(quarto)
  library(pander)
  library(plotly)
```

```
  library(gridExtra)
  library(caret) #for generating confusion matrix
  }))
```

```
# Load data preprocessed by different sepctral techniques

hsi_raw <- read.csv("coffee_data_pycleaned.csv")
hsi_snv <- read.csv("coffee_ground_snv.csv")
hsi_snv_1d <- read.csv("coffee_ground_snv_sg_1d.csv")
hsi_snv_2d <- read.csv("coffee_ground_snv_sg_2d.csv")
hsi_msc <- read.csv("coffee_ground_msc.csv")
hsi_msc_1d <- read.csv("coffee_ground_msc_sg_1d.csv")
hsi_msc_2d <- read.csv("coffee_ground_msc_sg_2d.csv")
```

```
# Let us ensure we have the same number of dimensions and the range of our coulumns/covariates

dim(hsi_raw)
```

```
[1] 1469  229
```

```
dim(hsi_snv)
```

```
[1] 1469  229
```

```
dim(hsi_snv_1d)
```

```
[1] 1469  229
```

```
dim(hsi_snv_2d)
```

```
[1] 1469  229
```

```
dim(hsi_msc)
```

```
[1] 1469  229
```

```
dim(hsi_msc_1d)
```

```
[1] 1469  229
```

```
dim(hsi_msc_2d)
```

```
[1] 1469  229
```

```
# Check a few column names
col_check <- colnames(hsi_msc[,c(1:6)])
col_check
```

```
[1] "sample_id"     "binary_class"  "three_class"   "adult_percent"
[5] "cal_val"       "X935.609985"
```

```
# The spectral values start from column # 6-------> 229

columns <- hsi_raw[,c(1:5)]
```

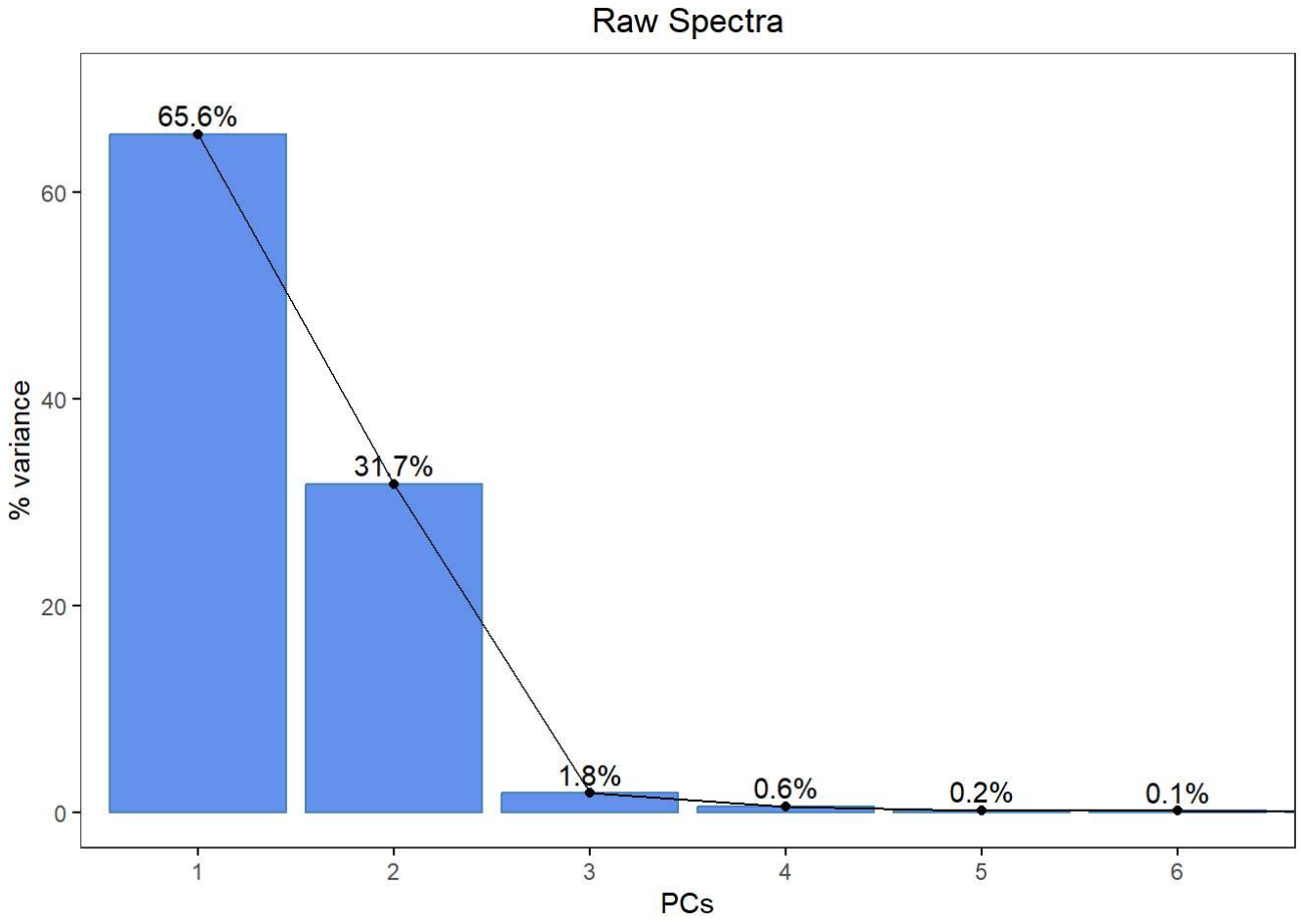
Raw Spectra (Unprocessed)

PCA Analysis: HSI Raw Spectral Data

```
pca_raw <- PCA(hsi_raw[,c(6:229)],ncp = 6, graph = FALSE)

pca_raw$eig[1:5,] # Extract the first 5 component eigenvalues
```

```
       eigenvalue percentage of variance cumulative percentage of variance
comp 1 146.9369529             65.5968540                           65.59685
comp 2  71.0849617             31.7343579                           97.33121
comp 3   4.0610077              1.8129499                           99.14416
comp 4   1.2632289              0.5639415                           99.70810
comp 5   0.3386989              0.1512049                           99.85931
```

```
# Scree Plot: Check the number of components to keep

fviz_eig(pca_raw, addlabels = TRUE, ylim = c(0, 70),
         xlim=c(1,6), main = 'Raw Spectra',barfill = "cornflowerblue",
         hjust = 0.5,
         ggtheme = theme_bw(),xlab = "PCs", ylab = "% variance")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(panel.grid = element_blank())
```




```
# The elbow is at PC2-----> To retain PC1 and PC2. However, we will most of the PCs and subseq

# Save the PCs in a new data frame
Pcs_hsi_raw <- as.data.frame(pca_raw$ind$coord[,c(1:6)])
colnames(Pcs_hsi_raw) <- c("PC1","PC2","PC3","PC4","PC5","PC6")
Pcs_hsi_raw <- cbind(columns, Pcs_hsi_raw) # Bind with the columns from initial data--> with s
kable(head(Pcs_hsi_raw))
```

sample_id	binary_class	three_class	adult_percent	cal_val	PC1	PC2	PC3	PC4
Pure_Arabica_10	pure_arabica	pure_arabica	0	1	9.551566	13.26732	0.6525435	1.2566445
Pure_Arabica_3	pure_arabica	pure_arabica	0	1	8.780026	20.03452	0.6924415	0.3597407
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	16.737292	18.74044	2.9525753	0.2477756
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	4.069887	11.88874	1.1476004	1.3918031
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	11.655970	16.62836	1.9071946	0.5583164
Pure_Arabica_25	pure_arabica	pure_arabica	0	2	8.907335	15.70847	1.0928471	-0.1436962

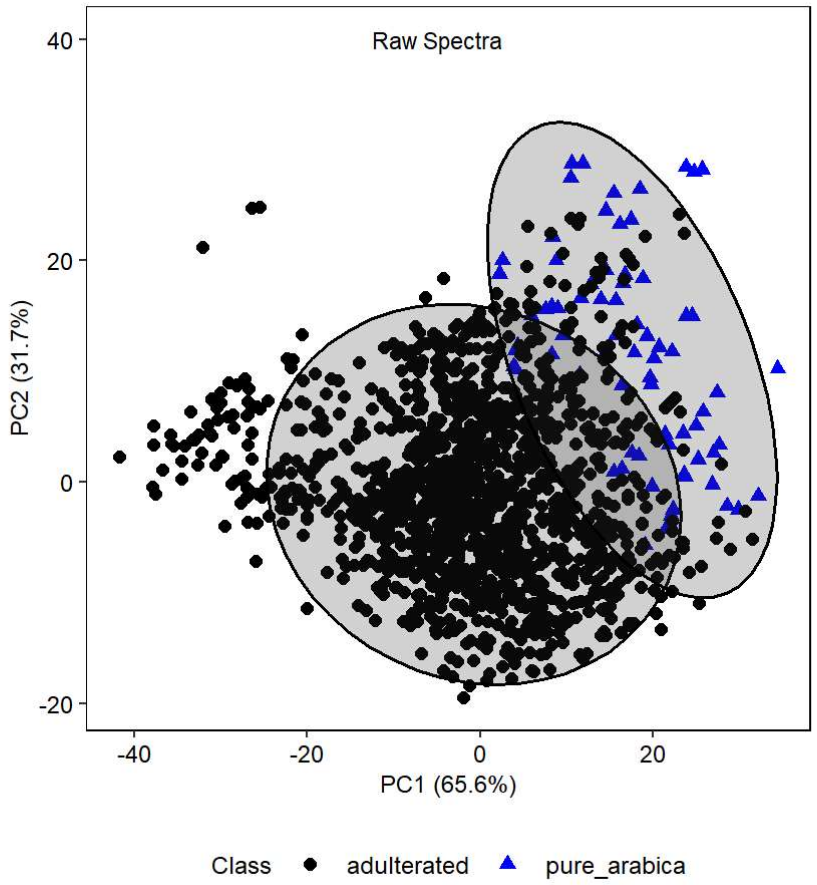
```
#write.csv(Pcs_hsi_raw, file = "pcs_hsi_raw.csv", row.names = FALSE) #save PC data
```

Insights

- The first two principal components, **PC1 and PC2**, account for the vast majority of the variance (**97.33%**), making them critical for analysis.

PCA Plot for HSI Raw Spectra

```
# Raw spectra PCA Plot

Pcs_hsi_raw %>%
  ggplot(mapping = aes(x = PC1, y = PC2,
                       shape = as.factor(binary_class), color = binary_class)) +
  geom_point(size= 2) +
  labs(x = "PC1 (65.6%)", y = "PC2 (31.7%)",
       title = "",shape = "Class", color = "Class") +
  theme_bw() +
  theme(
    panel.border = element_rect(color = 'black', fill = NA),
    panel.grid = element_blank(),
    axis.text.x = element_text(color = 'black', size = 9),
    axis.text.y = element_text(color = 'black', size = 9),
    aspect.ratio = 1,
    axis.title.x = element_text(size = 9),
    axis.title.y = element_text(size = 9),
    plot.title = element_text(size = 9, hjust = 0.5),
    legend.title = element_text(size = 9),
    legend.text = element_text(size = 9),
    legend.position = "bottom"
  ) +
  scale_color_manual(values = c("pure_arabica" = "blue", "adulterated" = "black")) +
  stat_ellipse(aes(group = binary_class),
               level = 0.95,
               geom = "polygon", alpha = 0.2,
               color = 'black', linewidth = 0.6)+
   annotate("text", x = -5, y = 40, label = "Raw Spectra", size = 3, color = "black")
```



Check Important variables contributing to PC1 and PC2 for HSI Raw Spectra

```
# Let us extract the loadings from the PCA
loadings <- pca_raw$var$coord

# Sort the loadings for PC1 and PC2
top_PC1 <- head(sort(abs(loadings[, 1]), decreasing = TRUE), 10) # Top 10 for PC1
top_PC2 <- head(sort(abs(loadings[, 2]), decreasing = TRUE), 10)  # Top 10 for PC2

# Combine into a data frame for easy viewing
top_PC1 <- as.data.frame(top_PC1)
top_PC2 <- as.data.frame(top_PC2)

# Display the top contributing variables
kable(top_PC1)
```

	top_PC1
X1364.920044	0.9902321
X1368.439941	0.9900957
X1361.390015	0.9900693
X1371.969971	0.9897523
X1357.869995	0.9894548
X1375.5	0.9891530

	top_PC1
X1354.349976	0.9884458
X1379.02002	0.9880565
X1350.819946	0.9868517
X1382.550049	0.9853808

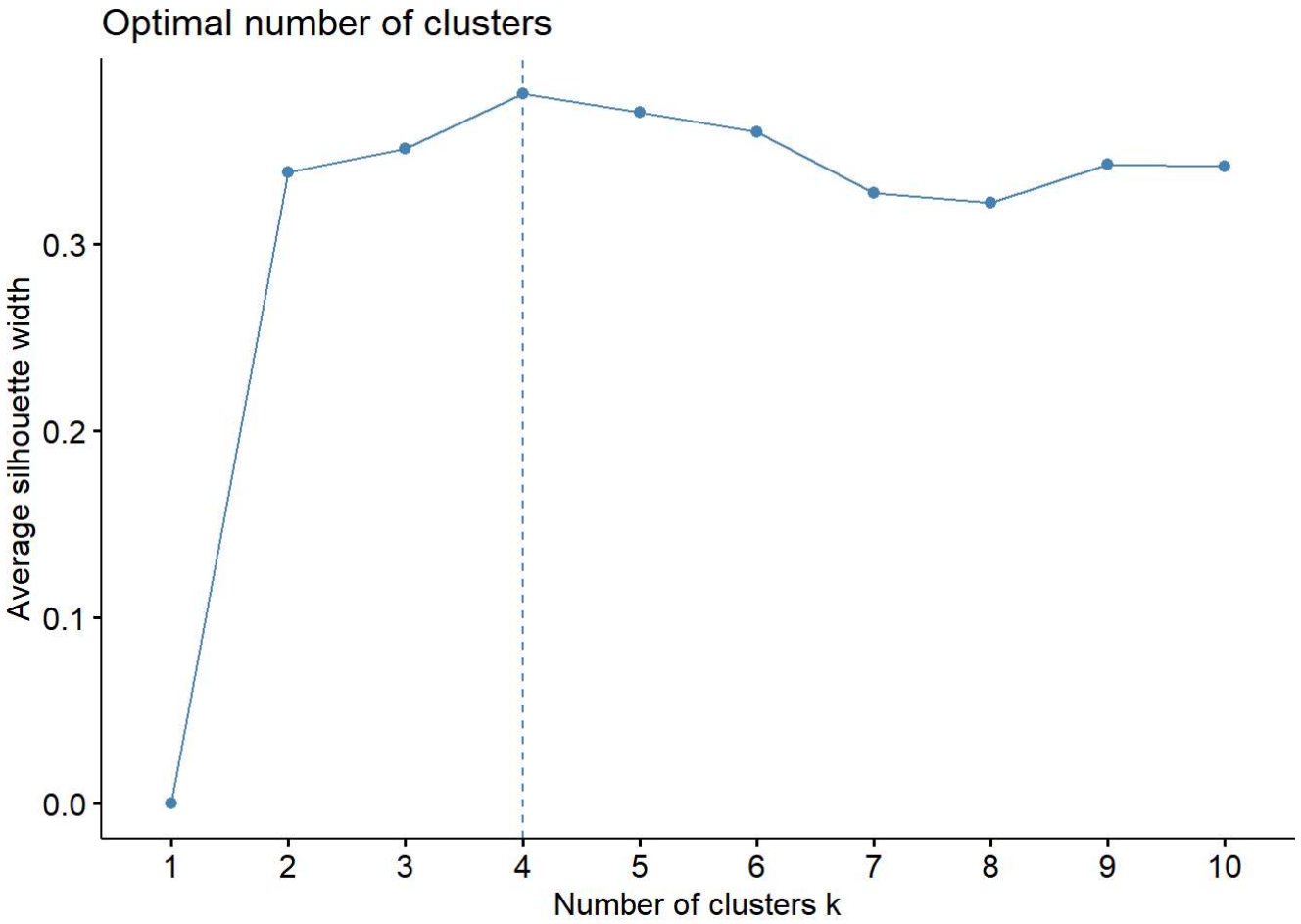
```
kable(top_PC2)
```

	top_PC2
X1580.98999	0.8077017
X1584.550049	0.8073871
X1577.430054	0.8073722
X1573.869995	0.8071899
X1588.109985	0.8067845
X1570.310059	0.8062990
X1591.670044	0.8053972
X1595.22998	0.8044535
X1566.75	0.8043830
X1563.199951	0.8025520

- The region from **1350 to 1382 nm** contributes most to the variation in PC1, while the region from **1563 to 1595 nm** contributes to PC2.

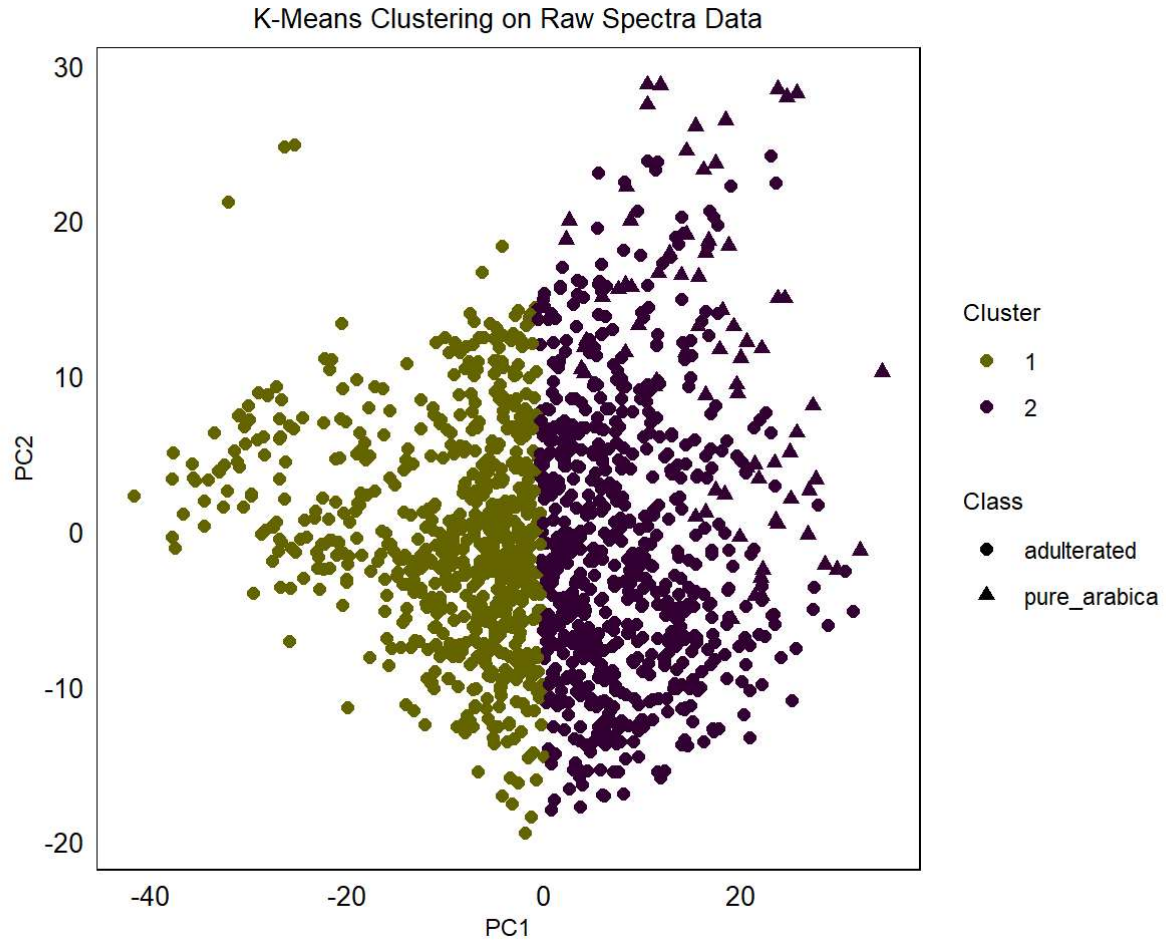
K-Means Clustering: Raw Spectra

```
# Use PCA reduced data to first determine the number of clusters  by silhouette method

# optimal number of clusters for HSI
raw_clust<-fviz_nbclust(Pcs_hsi_raw[,c(6,7)],
                        kmeans, method = "silhouette", k.max=10)

print(raw_clust) #determine the number of clusters
```

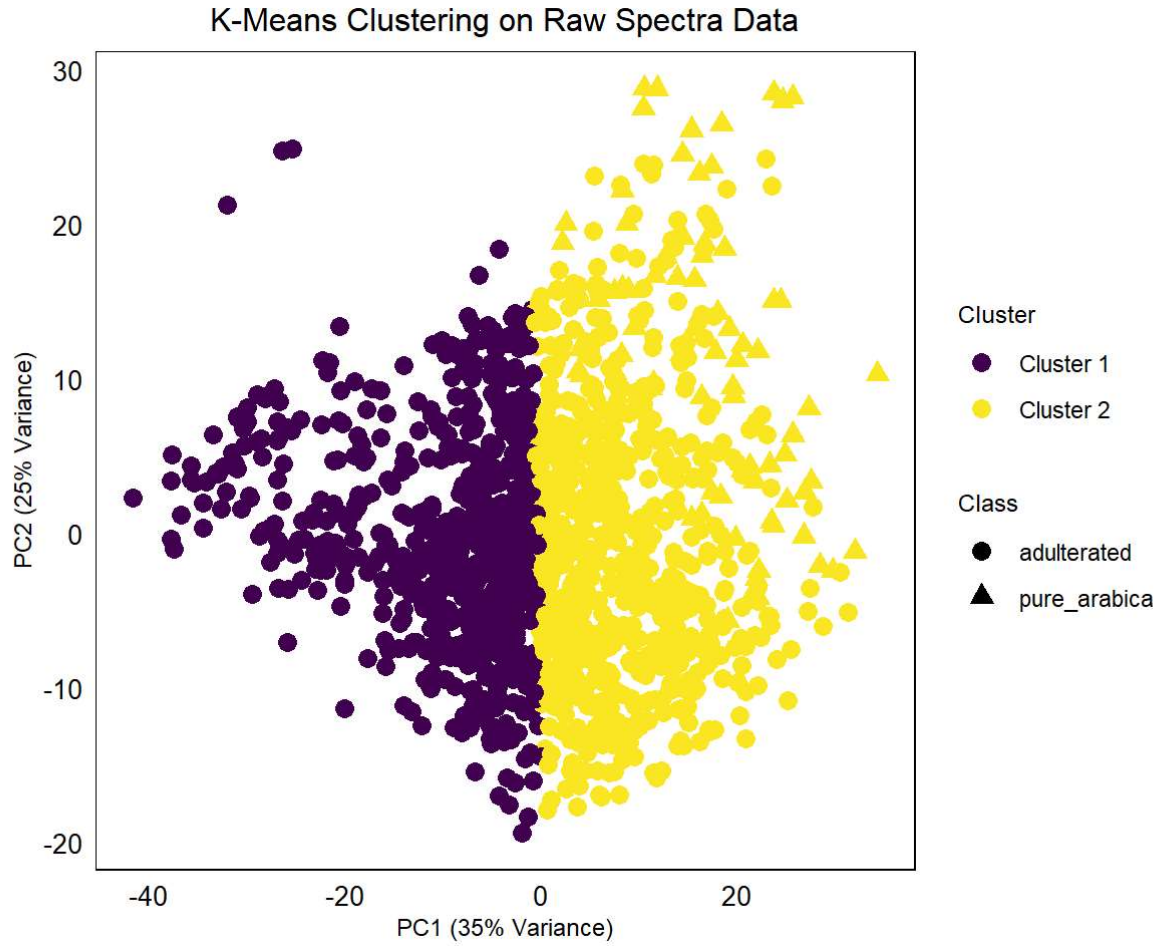



```r
# The method shows 4 as the optimal number of clusters. Due to the known goal of the study, we

# Perform K-Means Classification using the optimal number of clusters
raw_kmeans <- kmeans(Pcs_hsi_raw[,c(6,7)],centers = 2, nstart = 25)
raw_clusters <- as.factor(raw_kmeans$cluster) # Extract clusters
```

```r
# Add the cluster assignments to the data frame
Pcs_hsi_raw$Cluster <- raw_clusters

# Visualize the clusters
ggplot(Pcs_hsi_raw, aes(x = PC1, y = PC2, color = Cluster, shape = as.factor(binary_class))) +
  geom_point(size = 2) +
  labs(x = "PC1", y = "PC2", title = "K-Means Clustering on Raw Spectra Data", shape = 'Class'
 scale_color_manual(values = c("#660", "#330033")) +
  theme_minimal() +
  theme(panel.border = element_rect(color = 'black', fill = NA),
        panel.grid = element_blank(),
        axis.text.x = element_text(color = 'black', size = 10),
        axis.text.y = element_text(color = 'black', size = 10),
        aspect.ratio = 1,
        axis.title.x = element_text(size = 9),
        axis.title.y = element_text(size = 9),
        legend.title = element_text(size = 9),
        plot.title = element_text(size = 10, hjust = 0.5))
```




```
ggplot(Pcs_hsi_raw, aes(x = PC1, y = PC2, color = Cluster, shape = as.factor(binary_class))) +
  geom_point(size = 3) +
  labs(x = "PC1 (35% Variance)", y = "PC2 (25% Variance)", title = "K-Means Clustering on Raw S
  scale_color_viridis_d(option = "viridis", labels = c("Cluster 1", "Cluster 2")) +
  theme_minimal() +
  theme(panel.border = element_rect(color = 'black', fill = NA),
        panel.grid = element_blank(),
        axis.text.x = element_text(color = 'black', size = 10),
        axis.text.y = element_text(color = 'black', size = 10),
        aspect.ratio = 1,
        axis.title.x = element_text(size = 9),
        axis.title.y = element_text(size = 9),
        legend.title = element_text(size = 9),
        plot.title = element_text(size = 12, hjust = 0.5))
```



Confusion Matrix - Raw Spectra

```
# Make labels for the confusion matrix table
class_no <- as.factor(hsi_raw$binary_class) #select the binary class column and convert to a f
table(class_no) # confirm the number per class
```

```
class_no
 adulterated pure_arabica
        1394           75
```

```
class_k <- ifelse(class_no == 'adulterated',1, 2) # assign numbers to the groups

# Create a confusion matrix
cfmatrix_raw<-confusionMatrix(as.factor(raw_clusters),as.factor(class_k))

cfmatrix_raw #display the confusion matrix
```

```
Confusion Matrix and Statistics

          Reference
Prediction   1   2
         1 706   0
         2 688  75

               Accuracy : 0.5317
```

```
                 95% CI : (0.5058, 0.5574)
    No Information Rate : 0.9489
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0948

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.5065
            Specificity : 1.0000
         Pos Pred Value : 1.0000
         Neg Pred Value : 0.0983
             Prevalence : 0.9489
         Detection Rate : 0.4806
   Detection Prevalence : 0.4806
      Balanced Accuracy : 0.7532

       'Positive' Class : 1
```

```
kable(cfmatrix_raw$byClass)
```

	x
Sensitivity	0.5064562
Specificity	1.0000000
Pos Pred Value	1.0000000
Neg Pred Value	0.0982962
Precision	1.0000000
Recall	0.5064562
F1	0.6723810
Prevalence	0.9489449
Detection Rate	0.4805990
Detection Prevalence	0.4805990
Balanced Accuracy	0.7532281

Standard Normal Variate (SNV) Spectral Data

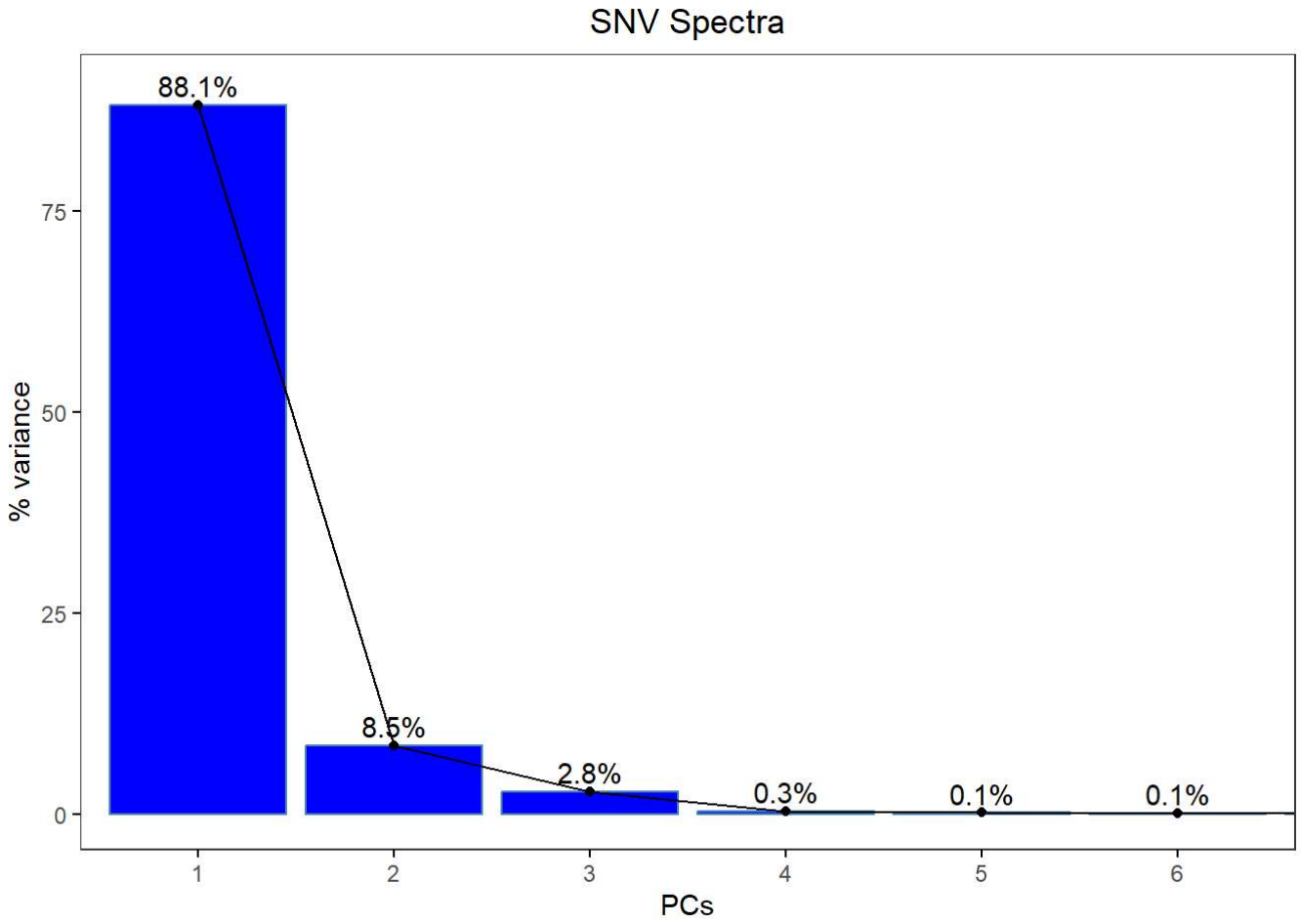
PCA Analysis: SNV pre-treated data

```
pca_snv <- PCA(hsi_snv[,c(6:229)],ncp = 5, graph = FALSE)

pca_snv$eig[1:5,] # Extract the first 5 component eigenvalues
```

```
        eigenvalue percentage of variance cumulative percentage of variance
comp 1 197.3480795             88.1018212                            88.10182
comp 2  19.0827700              8.5190938                            96.62091
```

```
comp 3   6.2034840   2.7694125   99.39033
comp 4   0.6399516   0.2856927   99.67602
comp 5   0.3010730   0.1344076   99.81043
```

```
# Scree Plot: Check the number of components to keep

fviz_eig(pca_snv, addlabels = TRUE, ylim = c(0, 90),
         xlim=c(1,6), main = 'SNV Spectra',barfill = "blue",
         hjust = 0.5,
         ggtheme = theme_bw(),xlab = "PCs", ylab = "% variance")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(panel.grid = element_blank())
```




```
# Save the PCs in a new data frame
Pcs_hsi_snv <- as.data.frame(pca_snv$ind$coord[,c(1:5)])
colnames(Pcs_hsi_snv) <- c("PC1","PC2","PC3","PC4","PC5")
Pcs_hsi_snv <- cbind(columns, Pcs_hsi_snv) # Bind with the columns from initial data--> with s
kable(head(Pcs_hsi_snv))
```

sample_id	binary_class	three_class	adult_percent	cal_val	PC1	PC2	PC3	PC4
Pure_Arabica_10	pure_arabica	pure_arabica	0	1	-10.92845	0.0692437	3.032866	1.4293193
Pure_Arabica_3	pure_arabica	pure_arabica	0	1	-21.17899	-4.4378327	2.244970	0.6628035
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-13.47325	-0.7823829	2.137894	1.8928154

sample_id	binary_class	three_class	adult_percent	cal_val	PC1	PC2	PC3	PC4
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-13.13640	-0.1849247	2.077591	1.9630751
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-14.32925	-0.8975206	2.268559	1.6703217
Pure_Arabica_25	pure_arabica	pure_arabica	0	2	-15.41867	-1.5177495	2.502160	1.7023439

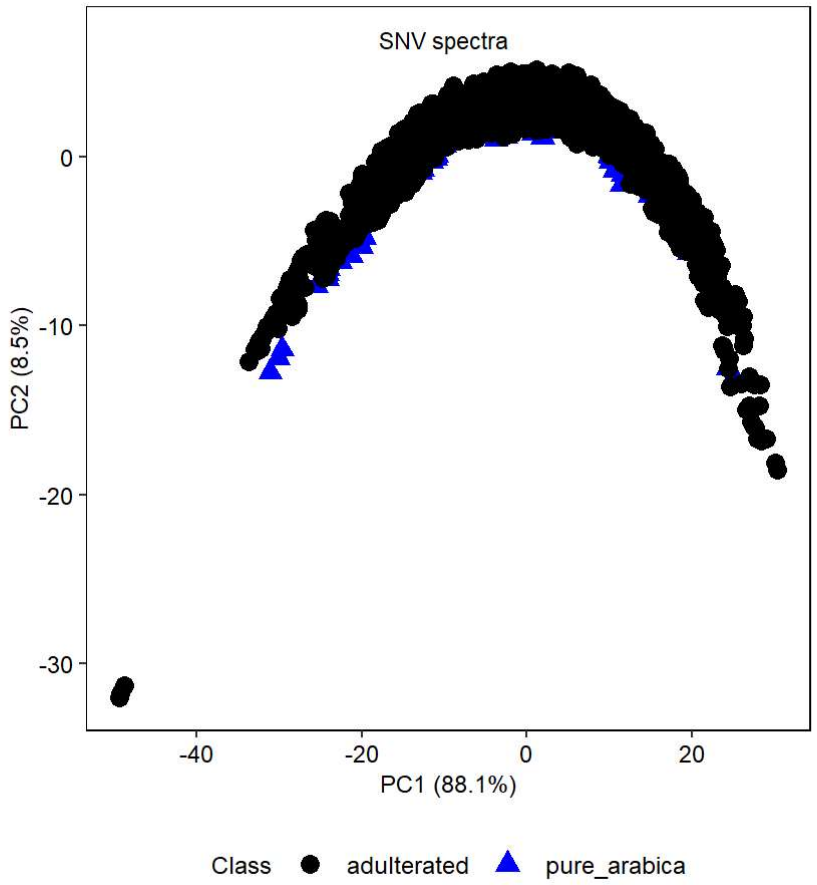
```
#write.csv(Pcs_hsi_snv, file = "pcs_hsi_snv.csv", row.names = FALSE) #save PC data
```

Insights: PCA-SNV spectral treatment

- The first two principal components, **PC1 and PC2**, account for the most the variance (**96.6%**).

PCA Plot for SNV Spectra

```
# SNV spectra PCA Plot

Pcs_hsi_snv %>%
  ggplot(mapping = aes(x = PC1, y = PC2,
                       shape = as.factor(binary_class), color = binary_class)) +
  geom_point(size= 3) +
  labs(x = "PC1 (88.1%)", y = "PC2 (8.5%)",
       title = "",shape = "Class", color = "Class") +
  theme_bw() +
  theme(
    panel.border = element_rect(color = 'black', fill = NA),
    panel.grid = element_blank(),
    axis.text.x = element_text(color = 'black', size = 9),
    axis.text.y = element_text(color = 'black', size = 9),
    aspect.ratio = 1,
    axis.title.x = element_text(size = 9),
    axis.title.y = element_text(size = 9),
    plot.title = element_text(size = 9, hjust = 0.5),
    legend.title = element_text(size = 9),
    legend.text = element_text(size = 9),
    legend.position = "bottom"
  ) +
  scale_color_manual(values = c("pure_arabica" = "blue", "adulterated" = "black")) +
   annotate("text", x = -10, y = 7, label = "SNV spectra", size = 3, color = "black")
```



Check Important variables contributing to PC1 and PC2 for HSI SNV Spectra

```
# Let us extract the loadings from the PCA
loadings <- pca_snv$var$coord

# Sort the loadings for PC1 and PC2
top_PC1 <- head(sort(abs(loadings[, 1]), decreasing = TRUE), 10) # Top 10 for PC1
top_PC2 <- head(sort(abs(loadings[, 2]), decreasing = TRUE), 10)  # Top 10 for PC2

# Combine into a data frame for easy viewing
top_PC1 <- as.data.frame(top_PC1)
top_PC2 <- as.data.frame(top_PC2)

# Display the top contributing variables
kable(top_PC1)
```

	top_PC1
X1548.97998	0.9976597
X1552.530029	0.9976566
X1545.420044	0.9976289
X1556.089966	0.9975457
X1541.869995	0.9975425
X1538.319946	0.9974280

	top_PC1
X1559.640015	0.9974233
X1534.76001	0.9973988
X1531.209961	0.9972933
X1563.199951	0.9972036

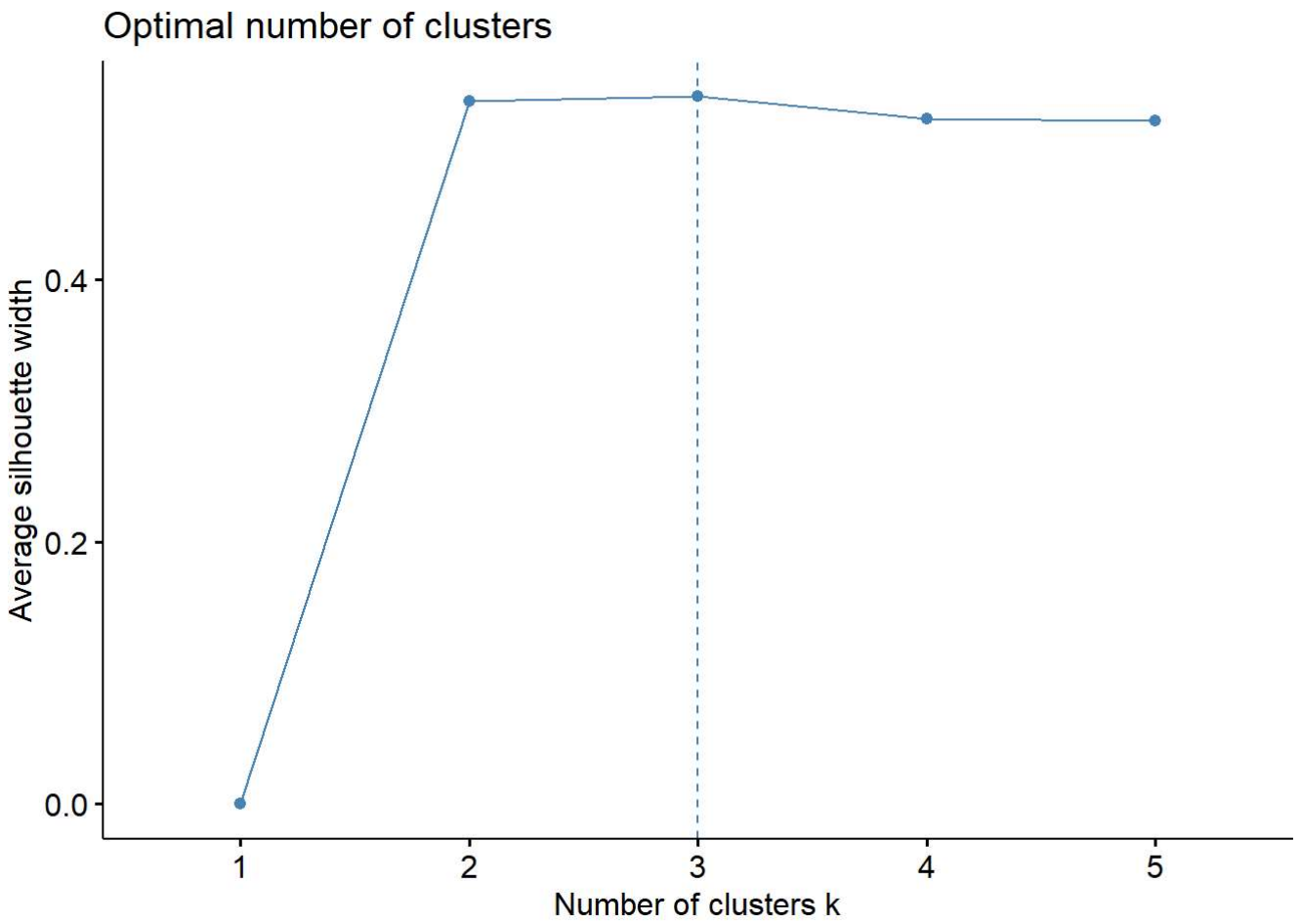
```
kable(top_PC2)
```

	top_PC2
X1347.300049	0.9469037
X1343.780029	0.9463100
X1340.26001	0.9240159
X1350.819946	0.9189437
X1336.73999	0.8868445
X1354.349976	0.8647280
X1333.219971	0.8431986
X1329.699951	0.7981270
X1357.869995	0.7981151
X1326.180054	0.7539149

- Similar to the raw spectra, spectral range of **1531 to 1563 nm*** and **1326 nm to 1357 nm**, contribute to vast of the variance in PC1 and PC2, respectively.

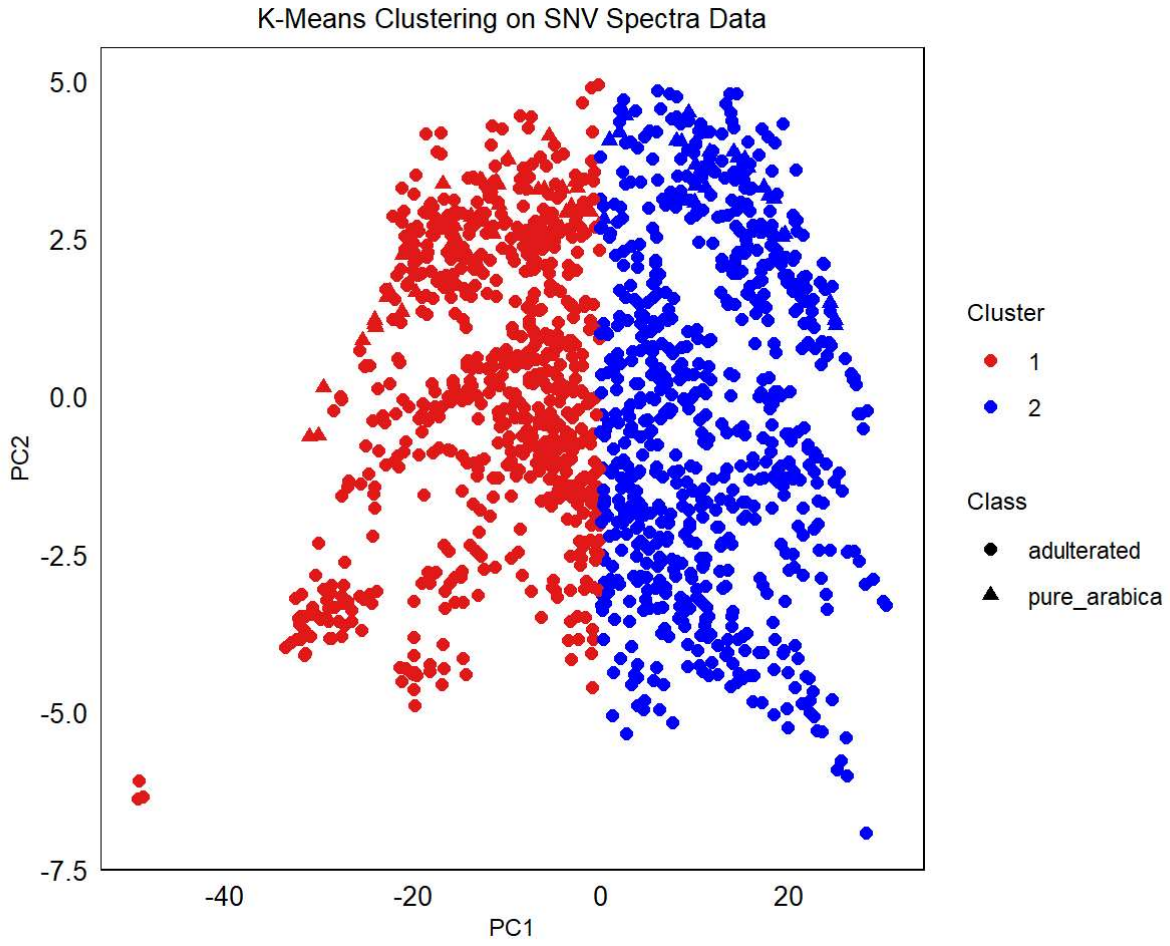
K-Means Clustering: SNV Spectra

```
# Use PCA reduced data to first determine the number of clusters  by silhouette method

# optimal number of clusters for HSI
snv_clust<-fviz_nbclust(Pcs_hsi_snv[,c(6,7)],
                        kmeans, method = "silhouette", k.max=5)

# Optimal number of cluster selected by kmeans = 3
print(snv_clust) #determine the number of clusters
```




```r
# Perform K-Means Classification using the optimal number of clusters
snv_kmeans <- kmeans(Pcs_hsi_snv[,c(6,7)],centers = 2, nstart = 25)
snv_clusters <- as.factor(snv_kmeans$cluster) # Extract clusters
```

```r
# Add the cluster assignments to the data frame
Pcs_hsi_snv$Cluster <- snv_clusters

# Visualize the clusters
ggplot(Pcs_hsi_snv, aes(x = PC1, y = PC3, color = Cluster, shape = as.factor(binary_class))) +
  geom_point(size = 2) +
  labs(x = "PC1", y = "PC2", title = "K-Means Clustering on SNV Spectra Data", shape = 'Class'
 scale_color_manual(values = c("#e31a1c","blue")) +
  theme_minimal() +
  theme(panel.border = element_rect(color = 'black', fill = NA),
        panel.grid = element_blank(),
        axis.text.x = element_text(color = 'black', size = 10),
        axis.text.y = element_text(color = 'black', size = 10),
        aspect.ratio = 1,
        axis.title.x = element_text(size = 9),
        axis.title.y = element_text(size = 9),
        legend.title = element_text(size = 9),
        plot.title = element_text(size = 10, hjust = 0.5))
```



Confusion Matrix - SNV Spectra

```
# Create a confusion matrix
cfmatrix_snv<-confusionMatrix(as.factor(snv_clusters),as.factor(class_k))

cfmatrix_snv #display the confusion matrix
```

```
Confusion Matrix and Statistics

          Reference
Prediction   1   2
         1 679  44
         2 715  31

               Accuracy : 0.4833
                 95% CI : (0.4575, 0.5092)
    No Information Rate : 0.9489
    P-Value [Acc > NIR] : 1

                  Kappa : -0.019

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.48709
            Specificity : 0.41333
         Pos Pred Value : 0.93914
```

```
          Neg Pred Value : 0.04155
              Prevalence : 0.94894
          Detection Rate : 0.46222
    Detection Prevalence : 0.49217
       Balanced Accuracy : 0.45021

        'Positive' Class : 1
```

```
kable(cfmatrix_snv$byClass)
```

	x
Sensitivity	0.4870875
Specificity	0.4133333
Pos Pred Value	0.9391425
Neg Pred Value	0.0415550
Precision	0.9391425
Recall	0.4870875
F1	0.6414738
Prevalence	0.9489449
Detection Rate	0.4622192
Detection Prevalence	0.4921715
Balanced Accuracy	0.4502104

Standard Normal Variate (SNV), Savitzky_Golay and 1st Derivative Spectral Data

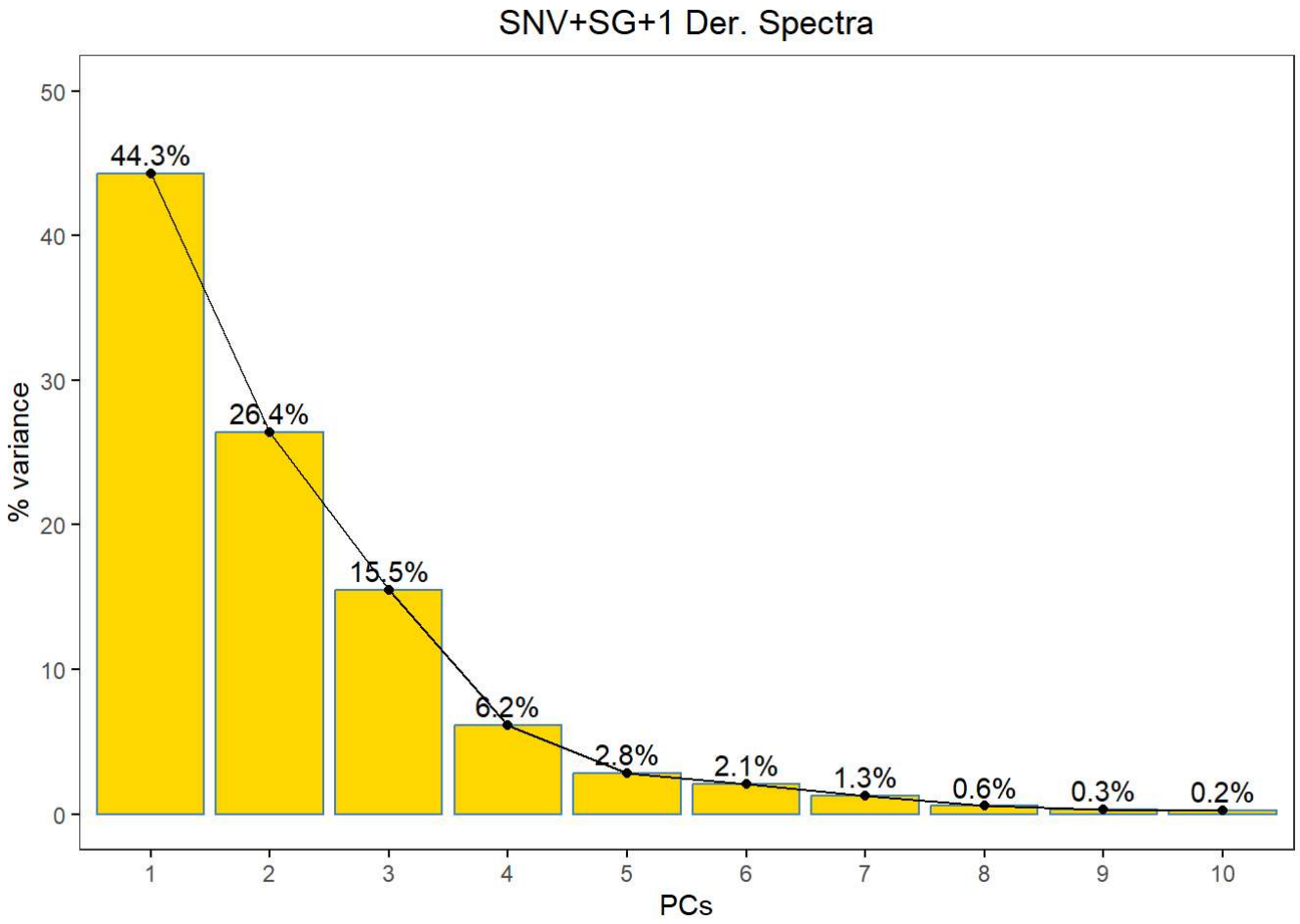
PCA Analysis: SNV+SG+1D pre-treated data

```
pca_snv_sg_1d <- PCA(hsi_snv_1d[,c(6:229)],ncp = 10, graph = FALSE)

pca_snv_sg_1d$eig[1:5,] # Extract the first 5 component eigenvalues
```

```
       eigenvalue percentage of variance cumulative percentage of variance
comp 1  99.175475              44.274766                           44.27477
comp 2  59.157127              26.409432                           70.68420
comp 3  34.743219              15.510365                           86.19456
comp 4  13.801326               6.161306                           92.35587
comp 5   6.301664               2.813243                           95.16911
```

```
# Scree Plot: Check the number of components to keep

fviz_eig(pca_snv_sg_1d, addlabels = TRUE, ylim = c(0, 50),
         xlim=c(1,10), main = 'SNV+SG+1 Der. Spectra',barfill = "gold",
         hjust = 0.5,
         ggtheme = theme_bw(),xlab = "PCs", ylab = "% variance")+
```

```
  theme(plot.title = element_text(hjust = 0.5))+
  theme(panel.grid = element_blank())
```




```
# Save the PCs in a new data frame
Pcs_hsi_snv_sg_1d <- as.data.frame(pca_snv_sg_1d$ind$coord[,c(1:10)])
colnames(Pcs_hsi_snv_sg_1d) <- c("PC1","PC2","PC3","PC4","PC5","PC6","PC7","PC8","PC9","PC10")
Pcs_hsi_snv_sg_1d <- cbind(columns, Pcs_hsi_snv_sg_1d) # Bind with the columns from initial da
kable(head(Pcs_hsi_snv_sg_1d))
```

sample_id	binary_class	three_class	adult_percent	cal_val	PC1	PC2	PC3	PC4
Pure_Arabica_10	pure_arabica	pure_arabica	0	1	7.816656	-7.249043	4.4910550	9.475175
Pure_Arabica_3	pure_arabica	pure_arabica	0	1	12.807400	-10.541332	-4.1325183	9.525875
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	8.763490	-6.795419	1.4471280	13.189943
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	9.057916	-5.966264	1.3586940	13.678044
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	9.434077	-6.938138	1.0048629	11.198098
Pure_Arabica_25	pure_arabica	pure_arabica	0	2	10.012251	-8.053628	0.4709614	12.676888

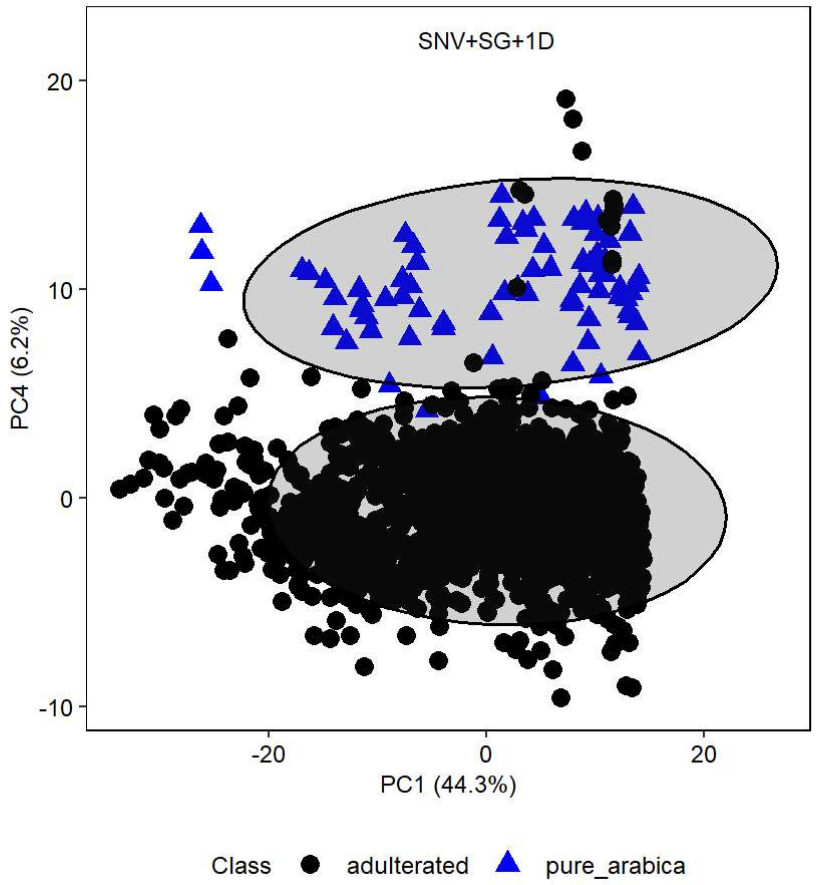
```
#write.csv(Pcs_hsi_snv_sg_1d, file = "pcs_hsi_snv_sg_1d.csv", row.names = FALSE) #save PC data
```

Insights: PCA-SNV+SG+1D spectral treatment

- The first four principal components account for 92.4% of the variation in the data.

PCA Plot for SNV+SG+1D Spectra

```
# SNV+SG+1D spectra PCA Plot

Pcs_hsi_snv_sg_1d %>%
  ggplot(mapping = aes(x = PC1, y = PC4,
                       shape = as.factor(binary_class), color = binary_class)) +
  geom_point(size=3) +
  labs(x = "PC1 (44.3%)", y = "PC4 (6.2%)",
       title = "",shape = "Class", color = "Class") +
  theme_bw() +
  theme(
    panel.border = element_rect(color = 'black', fill = NA),
    panel.grid = element_blank(),
    axis.text.x = element_text(color = 'black', size = 9),
    axis.text.y = element_text(color = 'black', size = 9),
    aspect.ratio = 1,
    axis.title.x = element_text(size = 9),
    axis.title.y = element_text(size = 9),
    plot.title = element_text(size = 9, hjust = 0.5),
    legend.title = element_text(size = 9),
    legend.text = element_text(size = 9),
    legend.position = "bottom"
  ) +
  stat_ellipse(aes(group = binary_class),
               level = 0.95,
               geom = "polygon", alpha = 0.2,
               color = 'black', linewidth = 0.6)+
  scale_color_manual(values = c("pure_arabica" = "blue", "adulterated" = "black")) +
   annotate("text", x = 0, y = 22, label = "SNV+SG+1D", size = 3, color = "black")
```



Check Important variables contributing to PC1 and PC2 for HSI SNV+SG+1st Derivative Spectra

```
# Let us extract the loadings from the PCA
loadings <- pca_snv_sg_1d$var$coord

# Sort the loadings for PC1 and PC2
top_PC1 <- head(sort(abs(loadings[, 1]), decreasing = TRUE), 10) # Top 10 for PC1
top_PC2 <- head(sort(abs(loadings[, 2]), decreasing = TRUE), 10)  # Top 10 for PC2

# Combine into a data frame for easy viewing
top_PC1 <- as.data.frame(top_PC1)
top_PC2 <- as.data.frame(top_PC2)

# Display the top contributing variables
kable(top_PC1)
```

	top_PC1
X1143.959961	0.9840986
X1140.469971	0.9824947
X1147.449951	0.9807038
X1276.969971	0.9801263
X1273.459961	0.9798308

	top_PC1
X1280.47998	0.9785021
X1269.949951	0.9782354
X1136.98999	0.9753663
X1266.439941	0.9751961
X1283.98999	0.9751264

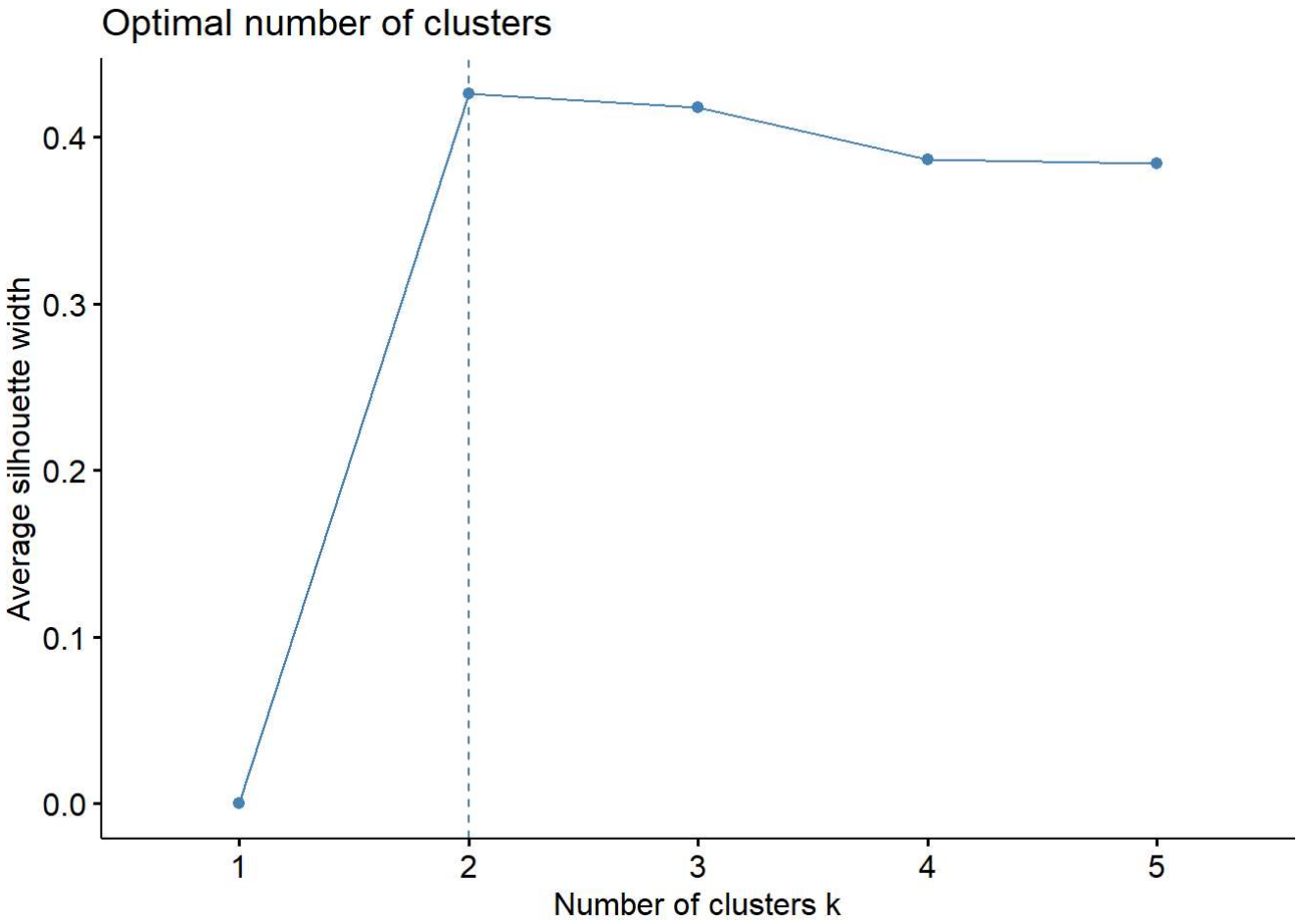
```
kable(top_PC2)
```

	top_PC2
X1573.869995	0.9817624
X1570.310059	0.9779056
X1577.430054	0.9735114
X1566.75	0.9680412
X1563.199951	0.9572951
X1559.640015	0.9472885
X1580.98999	0.9382606
X1556.089966	0.9379681
X1552.530029	0.9310056
X1548.97998	0.9277183

- According to the x-loadings, Spectral wavelengths in the regions **1136-1147 nm**, and **1266 nm to 1283 nm** contribute to the variance in PC1. On the other hand, the vast of the variation in PC2 is contributed by regions **1548 nm to 1580 nm**.

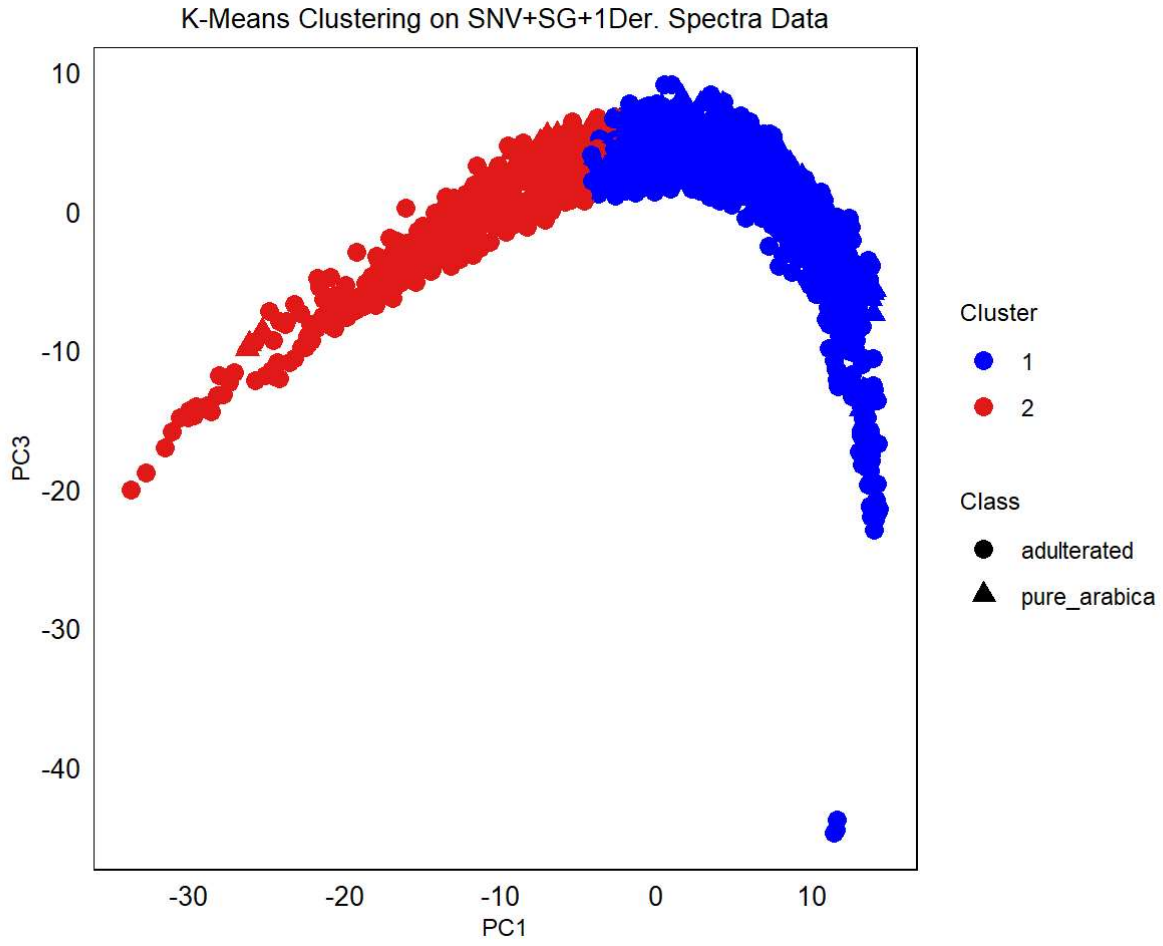
K-Means Clustering: SNV+SG+1st Derivative Spectra

```
# Use PCA reduced data to first determine the number of clusters  by silhouette method

# optimal number of clusters for HSI
snv_sg_1d_clust<-fviz_nbclust(Pcs_hsi_snv_sg_1d[,c(6,7)],
                        kmeans, method = "silhouette", k.max=5)

print(snv_sg_1d_clust) #determine the number of clusters
```




```r
# Perform K-Means Classification using the optimal number of clusters
snv_sg_1d_kmeans <- kmeans(Pcs_hsi_snv_sg_1d[,c(6,7)], centers = 2, nstart = 25)
snv_sg_1d_clusters <- as.factor(snv_sg_1d_kmeans$cluster) # Extract clusters
```

```r
# Add the cluster assignments to the data frame
Pcs_hsi_snv_sg_1d$Cluster <- snv_sg_1d_clusters

# Visualize the clusters
ggplot(Pcs_hsi_snv_sg_1d, aes(x = PC1, y = PC3, color = Cluster, shape = as.factor(binary_clas
  geom_point(size = 3) +
  labs(x = "PC1", y = "PC3", title = "K-Means Clustering on SNV+SG+1Der. Spectra Data", shape =
 scale_color_manual(values = c("blue", "#e31a1c")) +
  theme_minimal() +
  theme(panel.border = element_rect(color = 'black', fill = NA),
        panel.grid = element_blank(),
        axis.text.x = element_text(color = 'black', size = 10),
        axis.text.y = element_text(color = 'black', size = 10),
        aspect.ratio = 1,
        axis.title.x = element_text(size = 9),
        axis.title.y = element_text(size = 9),
        legend.title = element_text(size = 9),
        plot.title = element_text(size = 10, hjust = 0.5))
```



Confusion Matrix - SNV+SG+1D Spectra

```
# Create a confusion matrix
cfmatrix_snv_sg_1d<-confusionMatrix(as.factor(snv_sg_1d_clusters),as.factor(class_k))

cfmatrix_snv_sg_1d #display the confusion matrix
```

```
Confusion Matrix and Statistics

          Reference
Prediction   1   2
         1 961  48
         2 433  27

               Accuracy : 0.6726
                 95% CI : (0.6479, 0.6965)
    No Information Rate : 0.9489
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0144

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.6894
            Specificity : 0.3600
         Pos Pred Value : 0.9524
```

```
          Neg Pred Value : 0.0587
              Prevalence : 0.9489
          Detection Rate : 0.6542
    Detection Prevalence : 0.6869
       Balanced Accuracy : 0.5247

        'Positive' Class : 1
```

```
kable(cfmatrix_snv_sg_1d$byClass)
```

	x
Sensitivity	0.6893831
Specificity	0.3600000
Pos Pred Value	0.9524281
Neg Pred Value	0.0586957
Precision	0.9524281
Recall	0.6893831
F1	0.7998335
Prevalence	0.9489449
Detection Rate	0.6541865
Detection Prevalence	0.6868618
Balanced Accuracy	0.5246915

Standard Normal Variate (SNV), Savitzky_Golay and 2nd Derivative Spectral Data

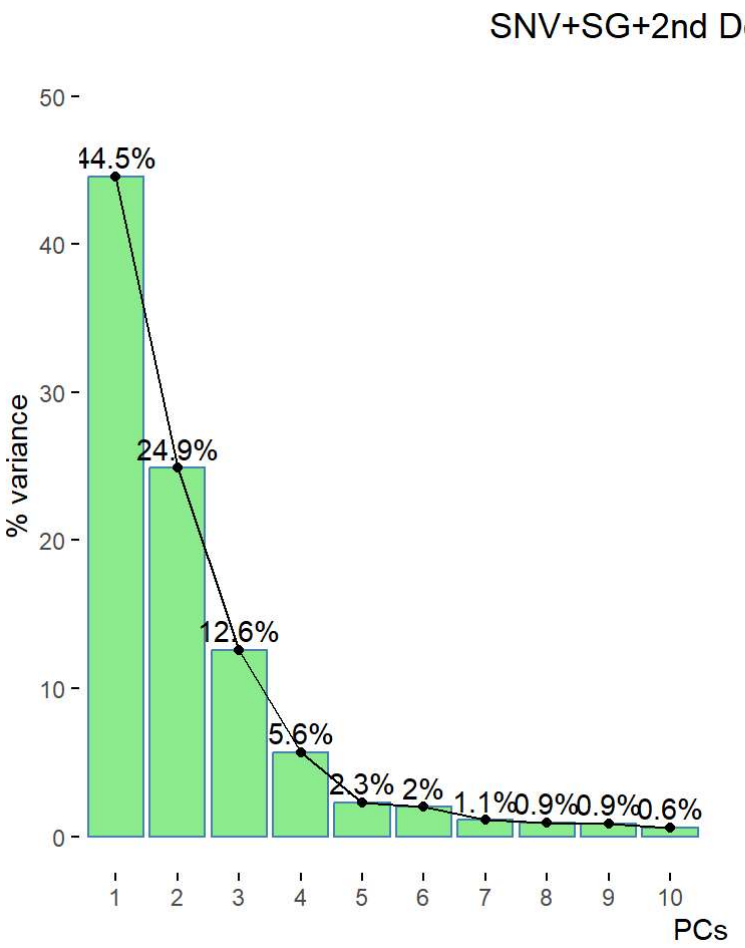
PCA Analysis: SNV+SG+2D pre-treated data

```
pca_snv_sg_2d <- PCA(hsi_snv_2d[,c(6:229)],ncp = 50, graph = FALSE)

pca_snv_sg_2d$eig[1:50,] # Extract the first 5 component eigenvalues
```

```
         eigenvalue percentage of variance cumulative percentage of variance
comp 1  99.79195521            44.54998001                           44.54998
comp 2  55.73163021            24.88019206                           69.43017
comp 3  28.22255368            12.59935432                           82.02953
comp 4  12.64085330             5.64323808                           87.67276
comp 5   5.08994917             2.27229874                           89.94506
comp 6   4.43325826             1.97913315                           91.92420
comp 7   2.46040577             1.09839543                           93.02259
comp 8   2.07181388             0.92491691                           93.94751
comp 9   1.96506015             0.87725899                           94.82477
comp 10  1.31491285             0.58701466                           95.41178
comp 11  0.99850436             0.44576088                           95.85754
comp 12  0.91175266             0.40703244                           96.26458
```

comp 13	0.80113423	0.35764921	96.62222
comp 14	0.75571210	0.33737147	96.95960
comp 15	0.62943043	0.28099573	97.24059
comp 16	0.53595949	0.23926763	97.47986
comp 17	0.49454487	0.22077896	97.70064
comp 18	0.44144627	0.19707423	97.89771
comp 19	0.37781196	0.16866605	98.06638
comp 20	0.33265309	0.14850584	98.21488
comp 21	0.32517360	0.14516678	98.36005
comp 22	0.31494294	0.14059953	98.50065
comp 23	0.29039349	0.12963995	98.63029
comp 24	0.24971028	0.11147780	98.74177
comp 25	0.22095142	0.09863903	98.84041
comp 26	0.19729152	0.08807657	98.92848
comp 27	0.17644570	0.07877040	99.00725
comp 28	0.14806544	0.06610064	99.07336
comp 29	0.14382659	0.06420830	99.13756
comp 30	0.13080471	0.05839496	99.19596
comp 31	0.11748936	0.05245061	99.24841
comp 32	0.11549756	0.05156141	99.29997
comp 33	0.10467121	0.04672822	99.34670
comp 34	0.09184069	0.04100031	99.38770
comp 35	0.08976135	0.04007203	99.42777
comp 36	0.08325133	0.03716577	99.46494
comp 37	0.07449675	0.03325748	99.49819
comp 38	0.06991812	0.03121345	99.52941
comp 39	0.05883016	0.02626346	99.55567
comp 40	0.05733964	0.02559805	99.58127
comp 41	0.05282051	0.02358058	99.60485
comp 42	0.05122582	0.02286867	99.62772
comp 43	0.04810602	0.02147590	99.64919
comp 44	0.04408047	0.01967878	99.66887
comp 45	0.04367733	0.01949881	99.68837
comp 46	0.03960714	0.01768176	99.70605
comp 47	0.03601857	0.01607972	99.72213
comp 48	0.03439203	0.01535358	99.73749
comp 49	0.03314331	0.01479612	99.75228
comp 50	0.02919453	0.01303327	99.76532

```
# Scree Plot: Check the number of components to keep

fviz_eig(pca_snv_sg_2d, addlabels = TRUE, ylim = c(0, 50),
         xlim=c(1,20), main = 'SNV+SG+2nd Der. Spectra',barfill = "lightgreen",
         hjust = 0.5,xlab = "PCs", ylab = "% variance")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(panel.grid = element_blank())
```




```
# Save the PCs in a new data frame
Pcs_hsi_snv_sg_2d <- as.data.frame(pca_snv_sg_2d$ind$coord[,c(1:50)])
colnames(Pcs_hsi_snv_sg_2d) <- c("PC1","PC2","PC3","PC4","PC5",
                                 "PC6","PC7","PC8","PC9","PC10",
                                 "PC11","PC12","PC13","PC14","PC15",
                                 "PC16","PC17","PC18","PC19","PC20",
                                 "P21","P22","P23","P24","PC25",
                                 "P26","PC27","PC28","PC29","PC30",
                                 "PC31","PC32","PC33","PC34","PC35",
                                 "PC36","PC37","PC38","PC39","PC40",
                                 "PC41","PC42","PC43","PC44","PC45",
                                 "PC46","PC47","PC48","PC49","PC50")
Pcs_hsi_snv_sg_2d <- cbind(columns, Pcs_hsi_snv_sg_2d) # Bind with the columns from initial da
kable(head(Pcs_hsi_snv_sg_2d))
```

sample_id	binary_class	three_class	adult_percent	cal_val	PC1	PC2	PC3	PC4
Pure_Arabica_10	pure_arabica	pure_arabica	0	1	-4.1719257	12.50206	12.60210	0.9371088
Pure_Arabica_3	pure_arabica	pure_arabica	0	1	3.7379458	15.96940	14.78799	-3.5886841
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-0.7500140	13.79452	17.61945	0.5415258
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-0.3999200	14.13544	18.61445	0.5027155
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	0.1146333	13.76127	14.44767	1.2300733

sample_id	binary_class	three_class	adult_percent	cal_val	PC1	PC2	PC3	PC4	
Pure_Arabica_25	pure_arabica	pure_arabica	0	2	-0.0848800	15.46835	17.89272	-2.1046567	-

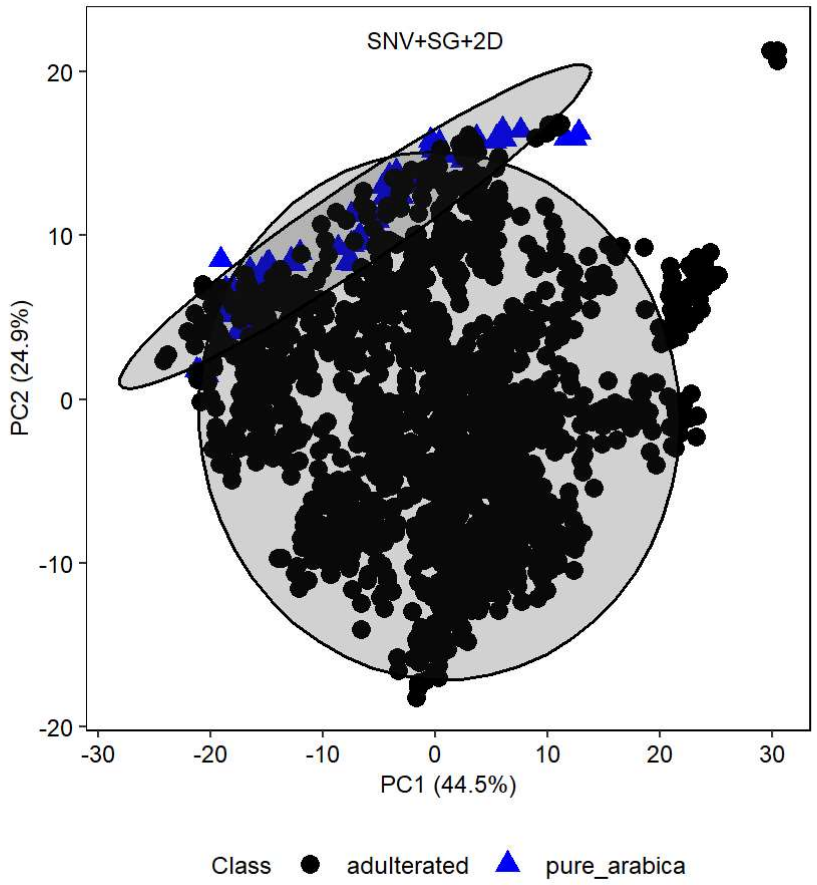
```
#write.csv(Pcs_hsi_snv_sg_2d, file = "pcs_hsi_snv_sg_2d.csv", row.names = FALSE) #save PC data
```

Insights: PCA-SNV+SG+2D spectral treatment

- The first five principal components account for about 90% of the variation in the data.

PCA Plot for SNV+SG+2D Spectra

```
# SNV+SG+1D spectra PCA Plot

Pcs_hsi_snv_sg_2d %>%
  ggplot(mapping = aes(x = PC1, y = PC2,
                       shape = as.factor(binary_class), color = binary_class)) +
  geom_point(size=3) +
  labs(x = "PC1 (44.5%)", y = "PC2 (24.9%)",
       title = "",shape = "Class", color = "Class") +
  theme_bw() +
  theme(
    panel.border = element_rect(color = 'black', fill = NA),
    panel.grid = element_blank(),
    axis.text.x = element_text(color = 'black', size = 9),
    axis.text.y = element_text(color = 'black', size = 9),
    aspect.ratio = 1,
    axis.title.x = element_text(size = 9),
    axis.title.y = element_text(size = 9),
    plot.title = element_text(size = 9, hjust = 0.5),
    legend.title = element_text(size = 9),
    legend.text = element_text(size = 9),
    legend.position = "bottom"
  ) +
  stat_ellipse(aes(group = binary_class),
               level = 0.95,
               geom = "polygon", alpha = 0.2,
               color = 'black', linewidth = 0.6)+
  scale_color_manual(values = c("pure_arabica" = "blue", "adulterated" = "black")) +
   annotate("text", x = 0, y = 22, label = "SNV+SG+2D", size = 3, color = "black")
```



Check Important variables contributing to PC1 and PC2 for HSI SNV+SG+2nd Derivative Spectra

```
# Let us extract the loadings from the PCA
loadings <- pca_snv_sg_2d$var$coord

# Sort the loadings for PC1 and PC2
top_PC1 <- head(sort(abs(loadings[, 1]), decreasing = TRUE), 10) # Top 10 for PC1
top_PC2 <- head(sort(abs(loadings[, 2]), decreasing = TRUE), 10)  # Top 10 for PC2

# Combine into a data frame for easy viewing
top_PC1 <- as.data.frame(top_PC1)
top_PC2 <- as.data.frame(top_PC2)

# Display the top contributing variables
kable(top_PC1)
```

	top_PC1
X1333.219971	0.9605254
X1224.369995	0.9525710
X1252.410034	0.9511852
X1255.920044	0.9507312
X1336.73999	0.9484054

	top_PC1
X1220.869995	0.9424500
X973.650024	0.9398907
X1248.900024	0.9390240
X1199.869995	0.9369854
X1259.420044	0.9369443

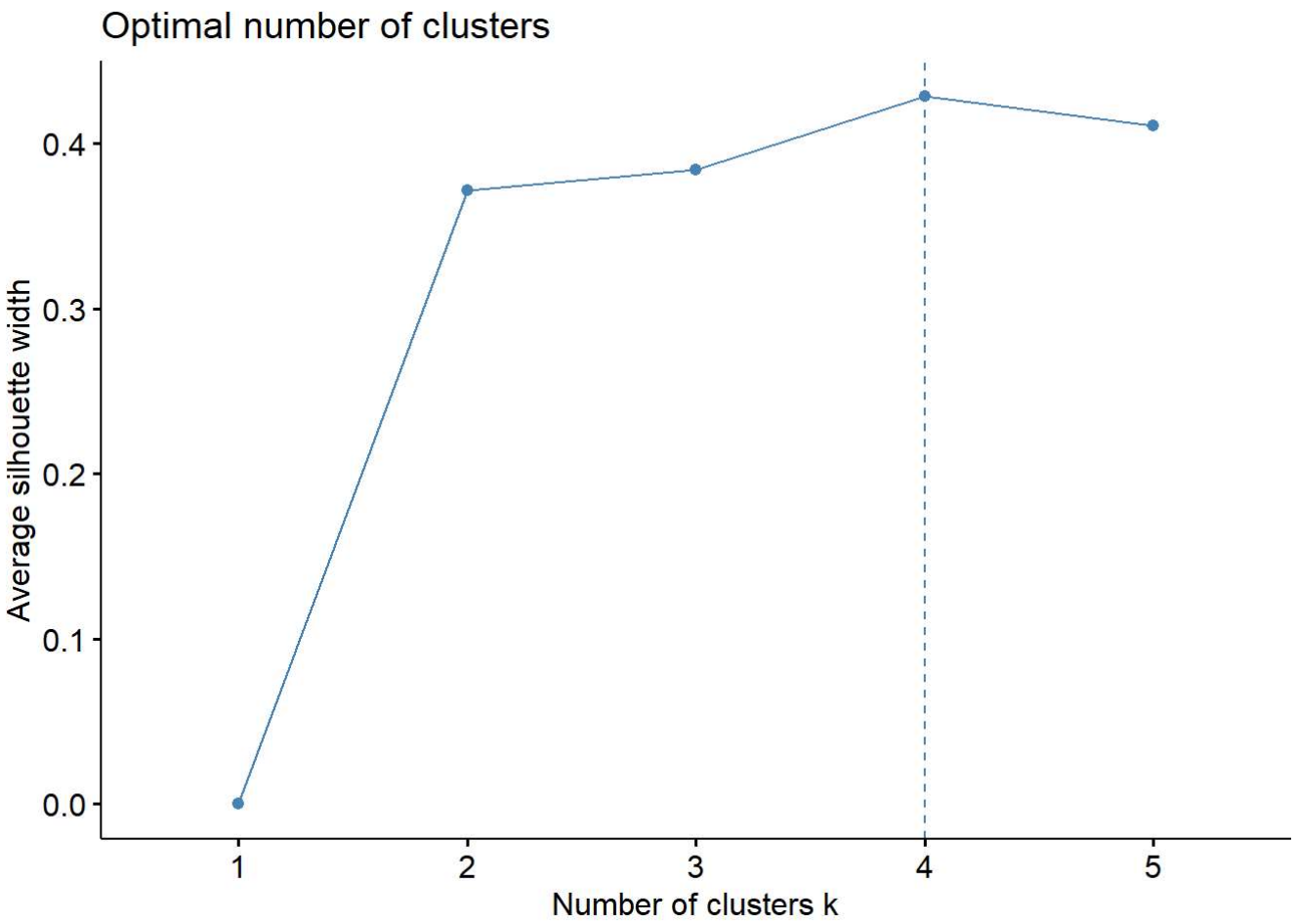
```
kable(top_PC2)
```

	top_PC2
X1424.920044	0.9637522
X1421.390015	0.9494860
X1287.51001	0.9397534
X1283.98999	0.9335110
X1389.609985	0.9235275
X1291.02002	0.9210477
X1305.079956	0.9109596
X1308.589966	0.9091320
X1294.530029	0.9076710
X1298.050049	0.9057625

- The variance in **PC1** is largely contributed by wavelengths regions around **1220-1259 nm** and **1333-1336 nm**, while regions **1287 - 1308 nm** and **1389 - 1424 nm** contribute to PC2.

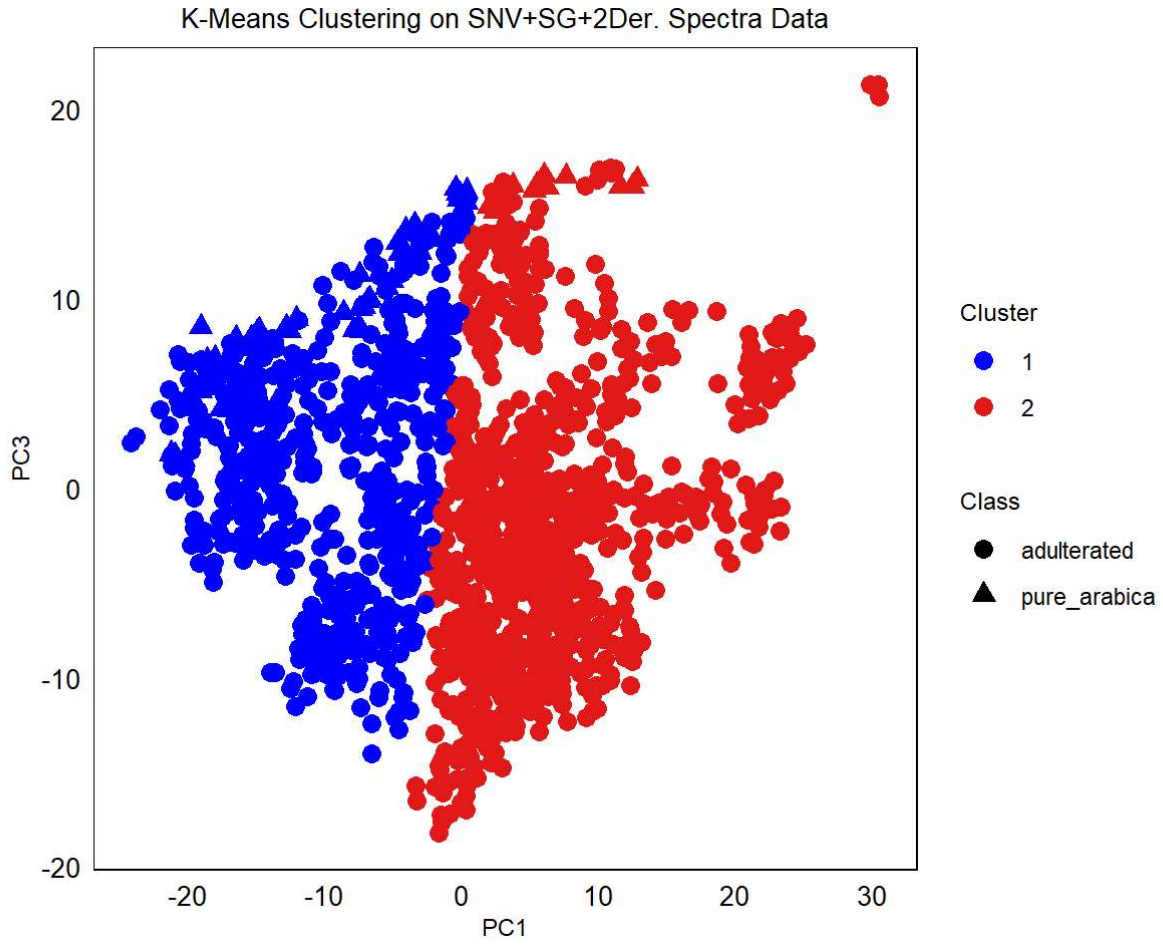
K-Means Clustering: SNV+SG+2nd Derivative Spectra

```
# Use PCA reduced data to first determine the number of clusters  by silhouette method

# optimal number of clusters for HSI
snv_sg_2d_clust<-fviz_nbclust(Pcs_hsi_snv_sg_2d[,c(6,7)],
                        kmeans, method = "silhouette", k.max=5)

print(snv_sg_2d_clust) #determine the number of clusters
```




```r
# Perform K-Means Classification using the optimal number of clusters
snv_sg_2d_kmeans <- kmeans(Pcs_hsi_snv_sg_2d[,c(6,7)], centers = 2, iter.max = 10, nstart = 25
snv_sg_2d_clusters <- as.factor(snv_sg_2d_kmeans$cluster) # Extract clusters
```

```r
# Add the cluster assignments to the data frame
Pcs_hsi_snv_sg_2d$Cluster <- snv_sg_2d_clusters

# Visualize the clusters
ggplot(Pcs_hsi_snv_sg_2d, aes(x = PC1, y = PC2, color = Cluster, shape = as.factor(binary_clas
  geom_point(size = 3) +
  labs(x = "PC1", y = "PC3", title = "K-Means Clustering on SNV+SG+2Der. Spectra Data", shape
 scale_color_manual(values = c("blue", "#e31a1c",  "#33a02c",  "#6a3d9a")) +
  theme_minimal() +
  theme(panel.border = element_rect(color = 'black', fill = NA),
        panel.grid = element_blank(),
        axis.text.x = element_text(color = 'black', size = 10),
        axis.text.y = element_text(color = 'black', size = 10),
        aspect.ratio = 1,
        axis.title.x = element_text(size = 9),
        axis.title.y = element_text(size = 9),
        legend.title = element_text(size = 9),
        plot.title = element_text(size = 10, hjust = 0.5))
```



Confusion Matrix - SNV+SG+2D Spectra

```
# Create a confusion matrix
cfmatrix_snv_sg_2d<-confusionMatrix(as.factor(snv_sg_2d_clusters),as.factor(class_k))

cfmatrix_snv_sg_2d #display the confusion matrix
```

```
Confusion Matrix and Statistics

          Reference
Prediction   1   2
         1 535  60
         2 859  15

               Accuracy : 0.3744
                 95% CI : (0.3496, 0.3997)
    No Information Rate : 0.9489
    P-Value [Acc > NIR] : 1

                  Kappa : -0.0689

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.38379
            Specificity : 0.20000
         Pos Pred Value : 0.89916
```

```
          Neg Pred Value : 0.01716
              Prevalence : 0.94894
          Detection Rate : 0.36419
    Detection Prevalence : 0.40504
       Balanced Accuracy : 0.29189

        'Positive' Class : 1
```

```
kable(cfmatrix_snv_sg_2d$byClass)
```

	x
Sensitivity	0.3837877
Specificity	0.2000000
Pos Pred Value	0.8991597
Neg Pred Value	0.0171625
Precision	0.8991597
Recall	0.3837877
F1	0.5379588
Prevalence	0.9489449
Detection Rate	0.3641933
Detection Prevalence	0.4050374
Balanced Accuracy	0.2918938

Multiplicative Scatter Correction (MSC) Spectral Data

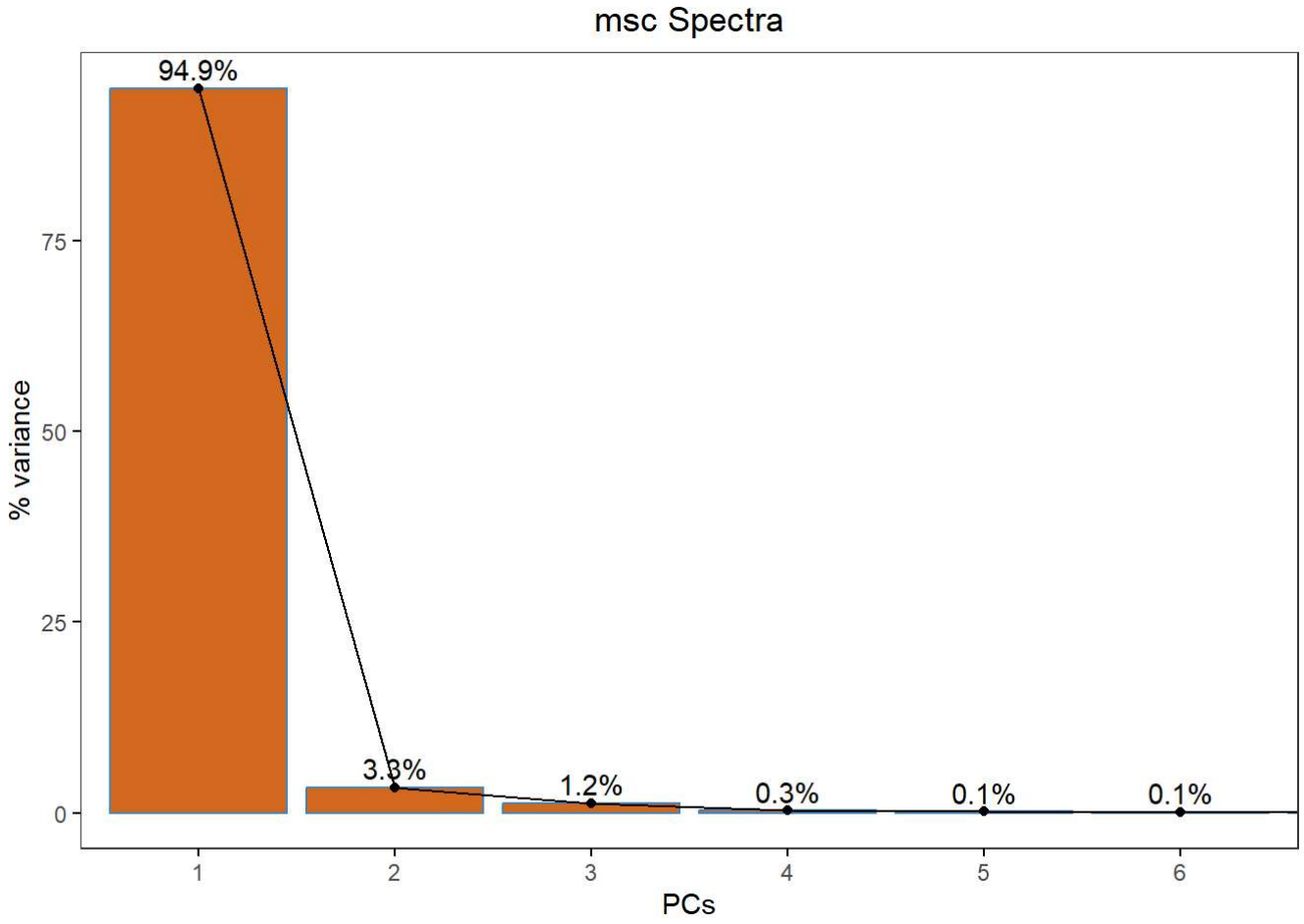
PCA Analysis: MSC pre-treated data

```
pca_msc <- PCA(hsi_msc[,c(6:229)],ncp = 10, graph = FALSE)

pca_msc$eig[1:10,] # Extract the first 5 component eigenvalues
```

```
         eigenvalue percentage of variance cumulative percentage of variance
comp 1  2.126479e+02           94.932097216                           94.93210
comp 2  7.374393e+00            3.292139717                           98.22424
comp 3  2.730568e+00            1.219003475                           99.44324
comp 4  6.332214e-01            0.282688122                           99.72593
comp 5  3.277157e-01            0.146301671                           99.87223
comp 6  1.430937e-01            0.063881102                           99.93611
comp 7  6.218628e-02            0.027761734                           99.96387
comp 8  2.713774e-02            0.012115062                           99.97599
comp 9  1.673640e-02            0.007471605                           99.98346
comp 10 6.475152e-03            0.002890693                           99.98635
```

```
# Scree Plot: Check the number of components to keep
```

```
fviz_eig(pca_msc, addlabels = TRUE, ylim = c(0, 95),
         xlim=c(1,6), main = 'msc Spectra',barfill = "chocolate",
         hjust = 0.5,
         ggtheme = theme_bw(),xlab = "PCs", ylab = "% variance")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(panel.grid = element_blank())
```




```
# Save the PCs in a new data frame
Pcs_hsi_msc <- as.data.frame(pca_msc$ind$coord[,c(1:10)])
colnames(Pcs_hsi_msc) <- c("PC1","PC2","PC3","PC4","PC5",
                           "PC6","PC7","PC8","PC10")
Pcs_hsi_msc <- cbind(columns, Pcs_hsi_msc) # Bind with the columns from initial data--> with s
kable(head(Pcs_hsi_msc))
```

sample_id	binary_class	three_class	adult_percent	cal_val	PC1	PC2	PC3	PC4
Pure_Arabica_10	pure_arabica	pure_arabica	0	1	-11.22488	-2.882522	-1.5531461	1.3020552
Pure_Arabica_3	pure_arabica	pure_arabica	0	1	-21.16959	-3.479679	-0.4609370	0.7100908
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-13.59825	-2.266777	-1.5606186	1.7231101
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-13.26080	-2.430731	-0.2737347	1.9459854
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-14.44702	-2.496798	-1.0735596	1.5607353
Pure_Arabica_25	pure_arabica	pure_arabica	0	2	-15.52722	-2.872859	-1.1624711	1.6180356

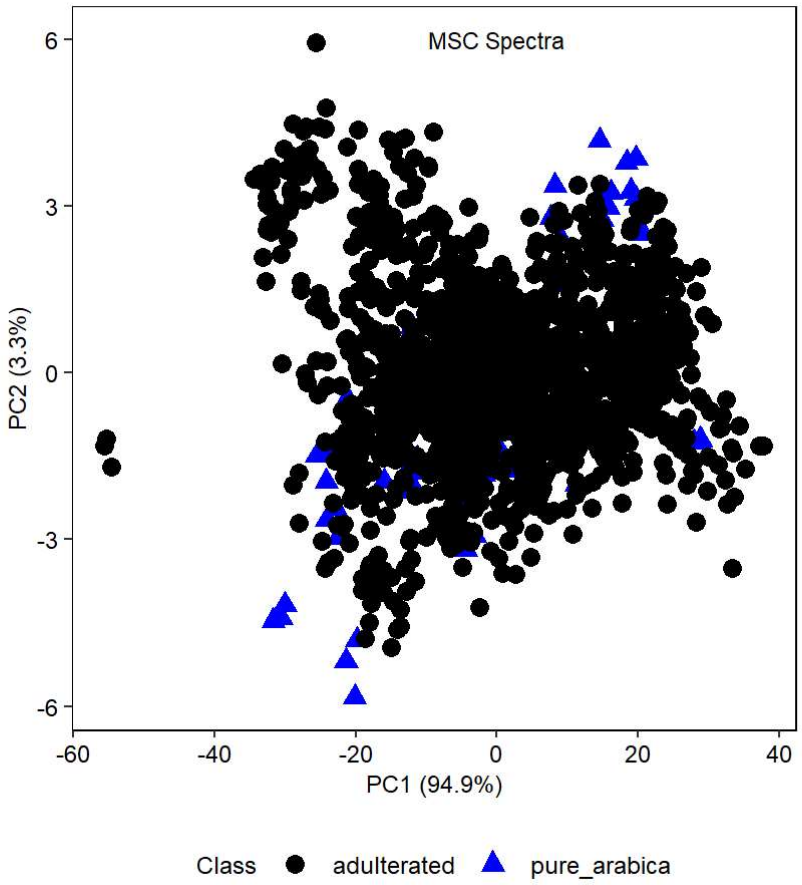
```
#write.csv(Pcs_hsi_msc, file = "pcs_hsi_msc.csv", row.names = FALSE) #save PC data
```

Insights: PCA-MSC spectral treatment

- The first two principal components, **PC1 and PC2**, account for the most the variance (**98.2%**).

PCA Plot for MSC Spectra

```
# msc spectra PCA Plot

Pcs_hsi_msc %>%
  ggplot(mapping = aes(x = PC1, y = PC3,
                       shape = as.factor(binary_class), color = binary_class)) +
  geom_point(size= 3) +
  labs(x = "PC1 (94.9%)", y = "PC2 (3.3%)",
       title = "",shape = "Class", color = "Class") +
  theme_bw() +
  theme(
    panel.border = element_rect(color = 'black', fill = NA),
    panel.grid = element_blank(),
    axis.text.x = element_text(color = 'black', size = 9),
    axis.text.y = element_text(color = 'black', size = 9),
    aspect.ratio = 1,
    axis.title.x = element_text(size = 9),
    axis.title.y = element_text(size = 9),
    plot.title = element_text(size = 9, hjust = 0.5),
    legend.title = element_text(size = 9),
    legend.text = element_text(size = 9),
    legend.position = "bottom"
  ) +
  scale_color_manual(values = c("pure_arabica" = "blue", "adulterated" = "black")) +
   annotate("text", x = 0, y = 6, label = "MSC Spectra", size = 3, color = "black")
```



Check Important variables contributing to PC1 and PC2 for HSI msc Spectra

```
# Let us extract the loadings from the PCA
loadings <- pca_msc$var$coord

# Sort the loadings for PC1 and PC2
top_PC1 <- head(sort(abs(loadings[, 1]), decreasing = TRUE), 10) # Top 10 for PC1
top_PC2 <- head(sort(abs(loadings[, 2]), decreasing = TRUE), 10)  # Top 10 for PC2

# Combine into a data frame for easy viewing
top_PC1 <- as.data.frame(top_PC1)
top_PC2 <- as.data.frame(top_PC2)

# Display the top contributing variables
#kable(top_PC1)
kable(top_PC2)
```

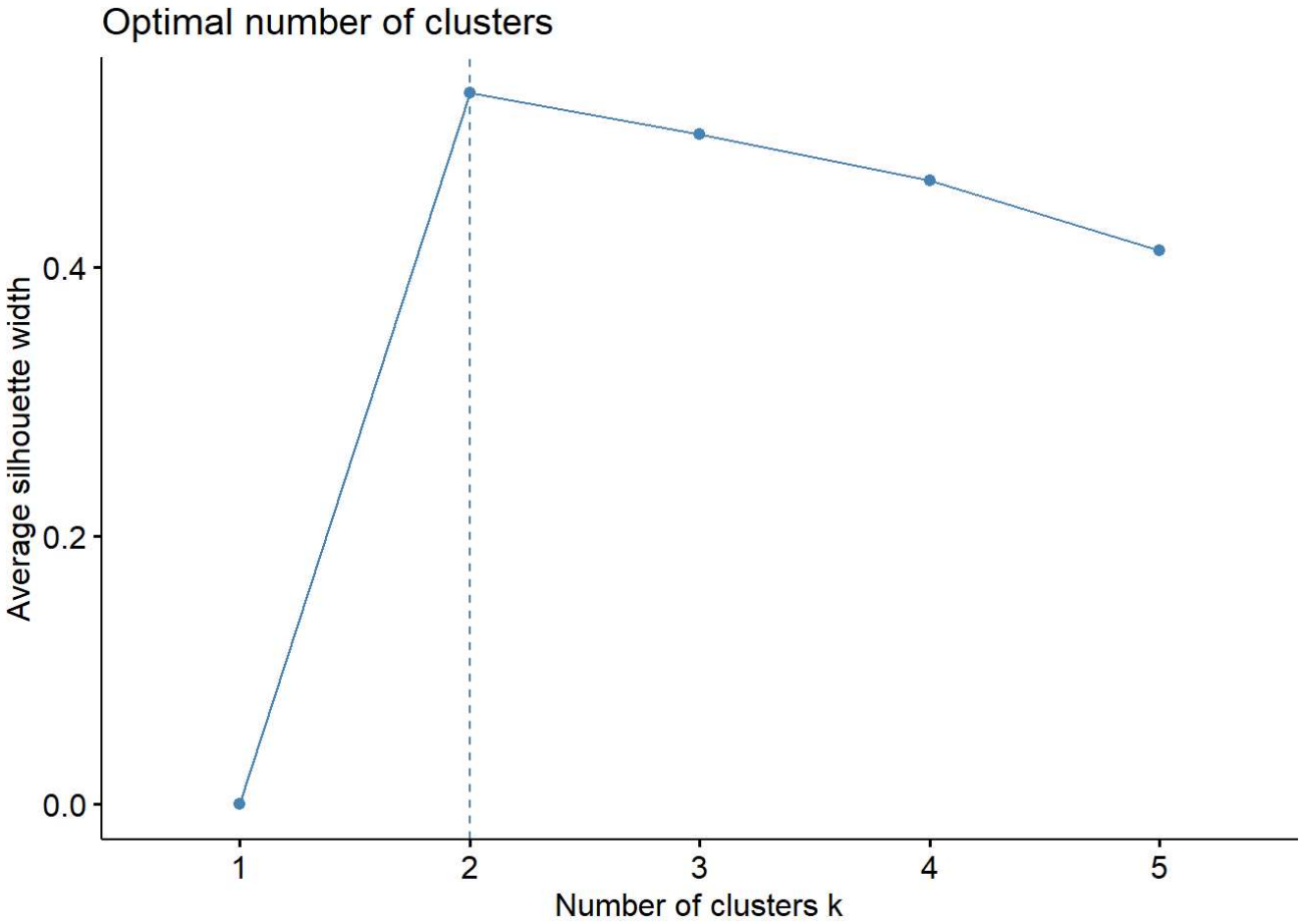
	top_PC2
X1343.780029	0.6534117
X1720.22998	0.5244273
X1716.650024	0.5124793
X1407.26001	0.4991740
X1713.069946	0.4987888

	top_PC2
X1410.790039	0.4955730
X1709.48999	0.4833175
X1403.72998	0.4821471
X1340.26001	0.4776650
X1414.319946	0.4737191

- Top contributor variable regions: **949 to 952 nm**, 1168 to 1178 nm, and **1548 to 1552 nm** for **PC1**. **PC2** is contributed by **1340-1343 nm**, **1403-1414 nm**, and **1709-1720 nm**.

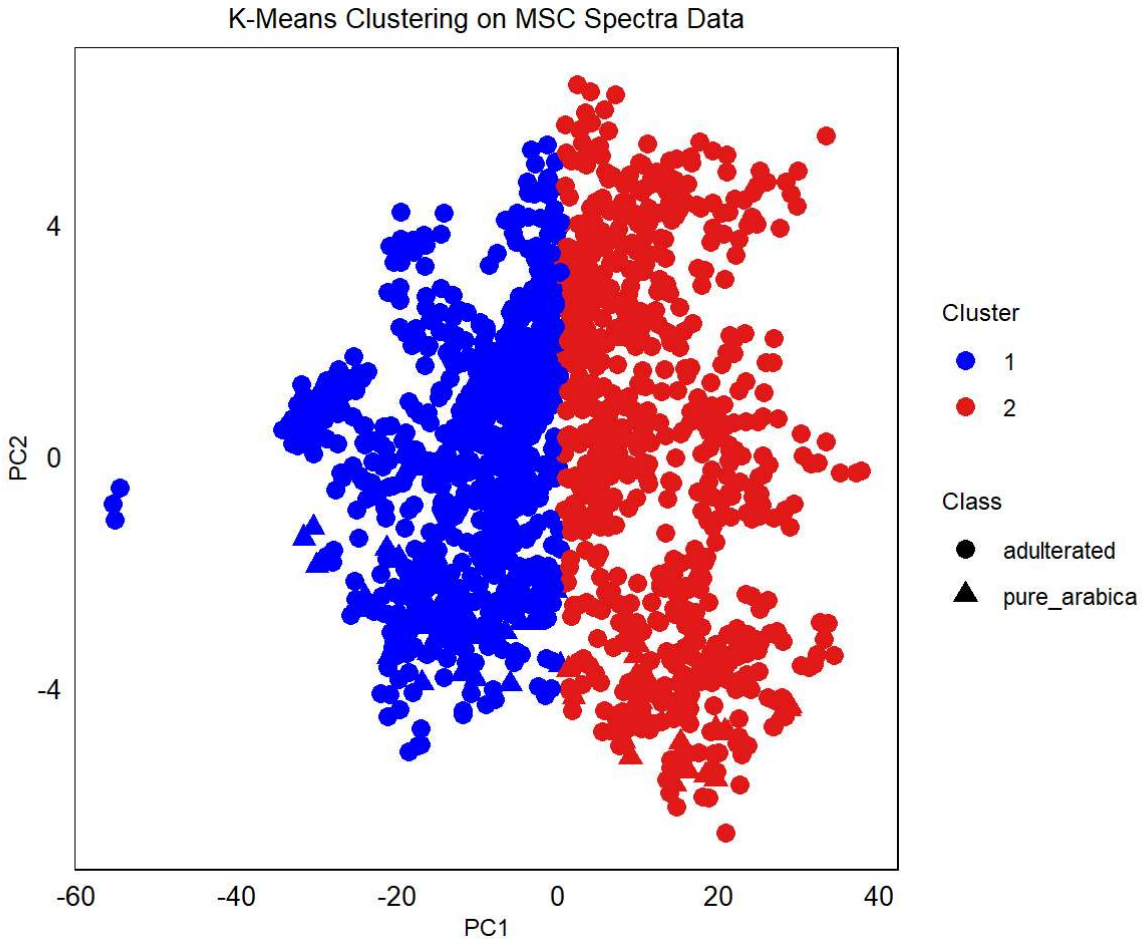
K-Means Clustering: MSC Spectra

```
# Use PCA reduced data to first determine the number of clusters  by silhouette method

# optimal number of clusters for HSI
msc_clust<-fviz_nbclust(Pcs_hsi_msc[,c(6,7)],
                        kmeans, method = "silhouette", k.max=5)

print(msc_clust) #determine the number of clusters
```




```
# Perform K-Means Classification using the optimal number of clusters
msc_kmeans <- kmeans(Pcs_hsi_msc[,c(6,7)],centers = 2,iter.max = 10,nstart = 25)
msc_clusters <- as.factor(msc_kmeans$cluster) # Extract clusters
```

```
# Add the cluster assignments to the data frame
Pcs_hsi_msc$Cluster <- msc_clusters

# Visualize the clusters
ggplot(Pcs_hsi_msc, aes(x = PC1, y = PC2, color = Cluster, shape = as.factor(binary_class))) +
  geom_point(size = 3) +
  labs(x = "PC1", y = "PC2", title = "K-Means Clustering on MSC Spectra Data", shape = 'Class'
 scale_color_manual(values = c("blue", "#e31a1c")) +
  theme_minimal() +
  theme(panel.border = element_rect(color = 'black', fill = NA),
        panel.grid = element_blank(),
        axis.text.x = element_text(color = 'black', size = 10),
        axis.text.y = element_text(color = 'black', size = 10),
        aspect.ratio = 1,
        axis.title.x = element_text(size = 9),
        axis.title.y = element_text(size = 9),
        legend.title = element_text(size = 9),
        plot.title = element_text(size = 10, hjust = 0.5))
```



Confusion Matrix - MSC Spectra

```
# Create a confusion matrix
cfmatrix_msc<-confusionMatrix(as.factor(msc_clusters),as.factor(class_k))

cfmatrix_msc #display the confusion matrix
```

```
Confusion Matrix and Statistics

          Reference
Prediction   1   2
         1 720  46
         2 674  29

               Accuracy : 0.5099
                 95% CI : (0.484, 0.5357)
    No Information Rate : 0.9489
    P-Value [Acc > NIR] : 1

                  Kappa : -0.0195

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.51650
            Specificity : 0.38667
         Pos Pred Value : 0.93995
         Neg Pred Value : 0.04125
             Prevalence : 0.94894
         Detection Rate : 0.49013
   Detection Prevalence : 0.52144
      Balanced Accuracy : 0.45158

       'Positive' Class : 1
```

```
kable(cfmatrix_msc$byClass)
```

	x
Sensitivity	0.5164993
Specificity	0.3866667
Pos Pred Value	0.9399478
Neg Pred Value	0.0412518
Precision	0.9399478
Recall	0.5164993
F1	0.6666667
Prevalence	0.9489449
Detection Rate	0.4901293
Detection Prevalence	0.5214432
Balanced Accuracy	0.4515830

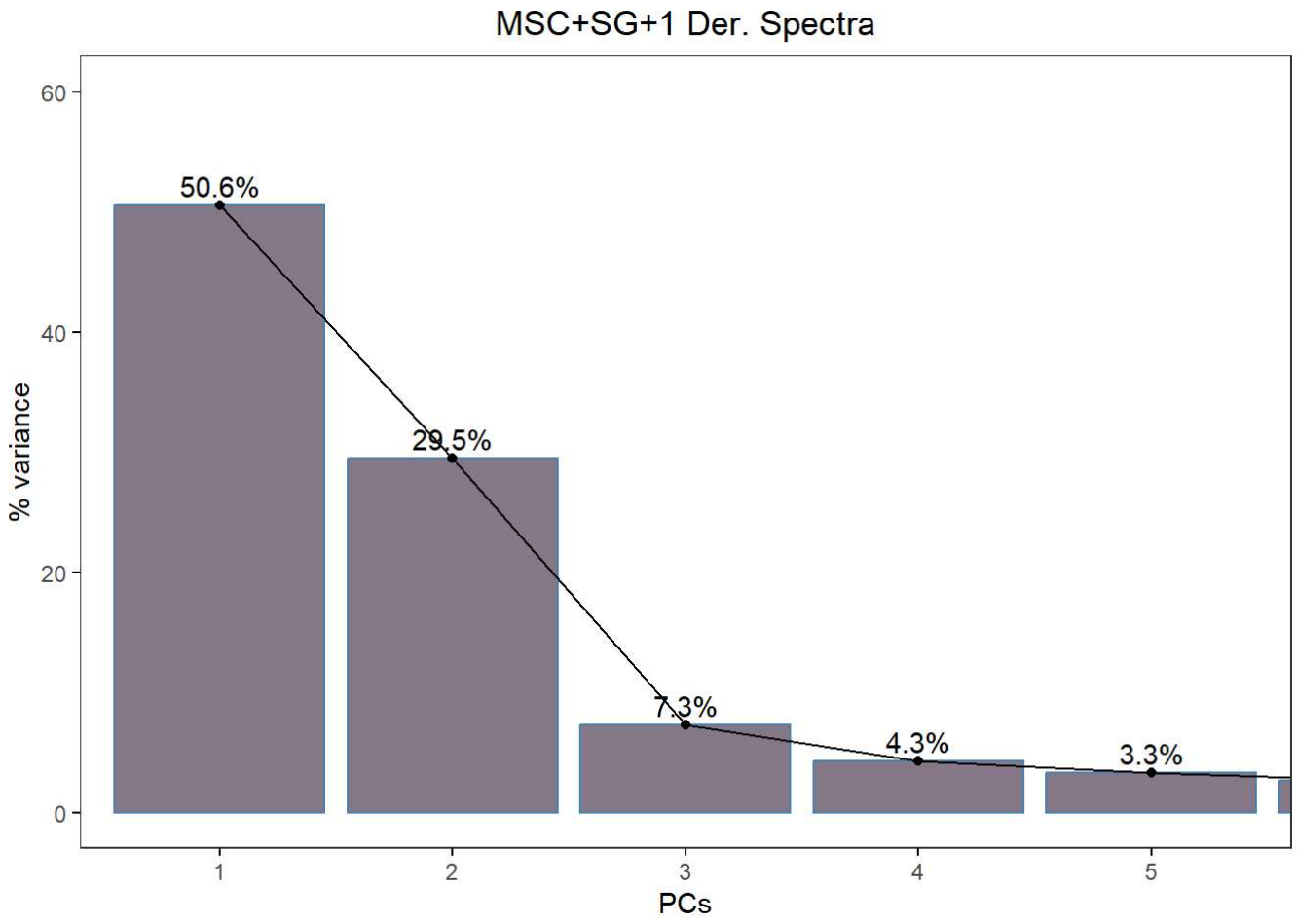
Multiplicative Scatter Correction, Savitzky_Golay and 1st Derivative Spectral Data

PCA Analysis: MSC+SG+1D pre-treated data

```
pca_msc_sg_1d <- PCA(hsi_msc_1d[,c(6:229)],ncp = 15, graph = FALSE)

pca_msc_sg_1d$eig[1:15,] # Extract the first 5 component eigenvalues
```

	eigenvalue	percentage of variance	cumulative percentage of variance
comp 1	113.23745897	50.55243704	50.55244
comp 2	66.03912396	29.48175177	80.03419
comp 3	16.38064826	7.31278940	87.34698
comp 4	9.54826909	4.26262013	91.60960
comp 5	7.37898827	3.29419119	94.90379
comp 6	5.83895934	2.60667828	97.51047
comp 7	2.40931519	1.07558714	98.58605
comp 8	0.94207295	0.42056828	99.00662
comp 9	0.88006189	0.39288477	99.39951
comp 10	0.28062795	0.12528034	99.52479
comp 11	0.23669423	0.10566707	99.63046
comp 12	0.15959145	0.07124618	99.70170
comp 13	0.12805712	0.05716836	99.75887
comp 14	0.07727134	0.03449613	99.79337
comp 15	0.06335689	0.02828433	99.82165

```
# Scree Plot: Check the number of components to keep

fviz_eig(pca_msc_sg_1d, addlabels = TRUE, ylim = c(0, 60),
         xlim=c(1,5), main = 'MSC+SG+1 Der. Spectra',barfill = "thistle4",
         hjust = 0.5,
         ggtheme = theme_bw(),xlab = "PCs", ylab = "% variance")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(panel.grid = element_blank())
```




```
# Save the PCs in a new data frame
Pcs_hsi_msc_sg_1d <- as.data.frame(pca_msc_sg_1d$ind$coord[,c(1:15)])
colnames(Pcs_hsi_msc_sg_1d) <- c("PC1","PC2","PC3","PC4","PC5",
                                 "PC6","PC7","PC8","PC9","PC10",
                                 "PC11","PC12","PC13","PC14","PC15")
Pcs_hsi_msc_sg_1d <- cbind(columns, Pcs_hsi_msc_sg_1d) # Bind with the columns from initial da
kable(head(Pcs_hsi_msc_sg_1d))
```

sample_id	binary_class	three_class	adult_percent	cal_val	PC1	PC2	PC3	PC4
Pure_Arabica_10	pure_arabica	pure_arabica	0	1	8.458034	-8.445949	9.045044	3.911966
Pure_Arabica_3	pure_arabica	pure_arabica	0	1	17.144920	-10.099217	8.065836	5.548340
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	10.283693	-7.672246	12.136694	3.961296
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	10.275313	-6.561373	12.015279	5.246912
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	11.166464	-7.654687	10.446477	3.414268
Pure_Arabica_25	pure_arabica	pure_arabica	0	2	12.023117	-8.334828	10.974346	5.513674

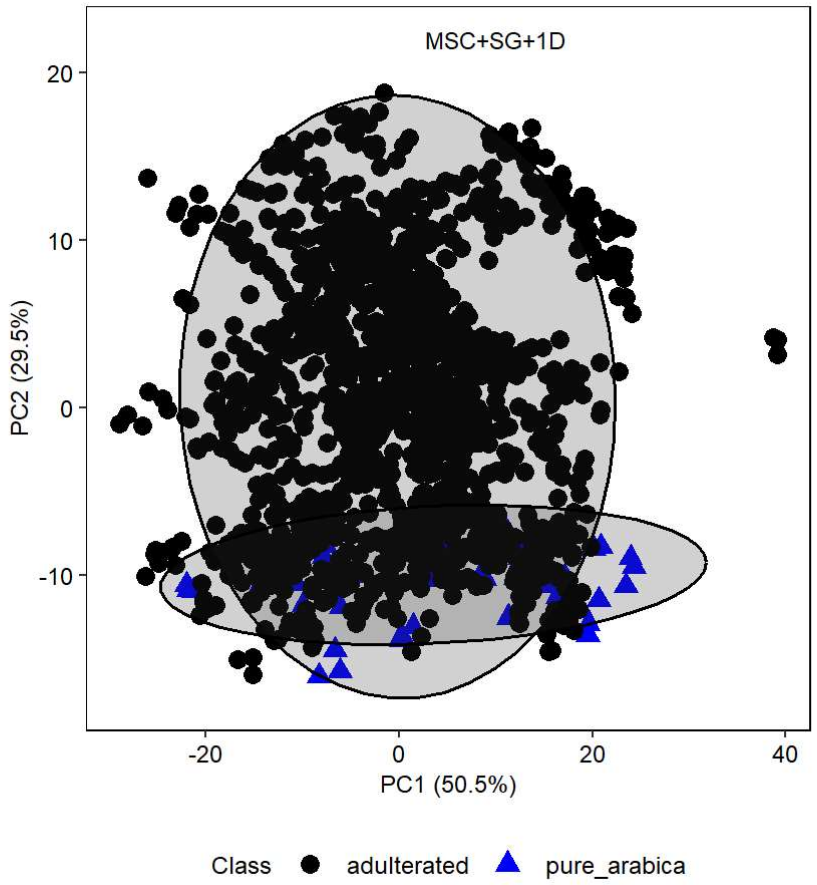
```
#write.csv(Pcs_hsi_msc_sg_1d, file = "pcs_hsi_msc_sg_1d.csv", row.names = FALSE) #save PC data
```

Insights: PCA-MSC+SG+2D spectral treatment

- The first four principal components account for 80% of the variation in the data.

PCA Plot for msc+SG+1D Spectra

```
# MSC+SG+1D spectra PCA Plot

Pcs_hsi_msc_sg_1d %>%
  ggplot(mapping = aes(x = PC1, y = PC2,
                       shape = as.factor(binary_class), color = binary_class)) +
  geom_point(size=3) +
  labs(x = "PC1 (50.5%)", y = "PC2 (29.5%)",
       title = "",shape = "Class", color = "Class") +
  theme_bw() +
  theme(
    panel.border = element_rect(color = 'black', fill = NA),
    panel.grid = element_blank(),
    axis.text.x = element_text(color = 'black', size = 9),
    axis.text.y = element_text(color = 'black', size = 9),
    aspect.ratio = 1,
    axis.title.x = element_text(size = 9),
    axis.title.y = element_text(size = 9),
    plot.title = element_text(size = 9, hjust = 0.5),
    legend.title = element_text(size = 9),
    legend.text = element_text(size = 9),
    legend.position = "bottom"
  ) +
  stat_ellipse(aes(group = binary_class),
               level = 0.95,
               geom = "polygon", alpha = 0.2,
               color = 'black', linewidth = 0.6)+
  scale_color_manual(values = c("pure_arabica" = "blue", "adulterated" = "black")) +
   annotate("text", x = 10, y = 22, label = "MSC+SG+1D", size = 3, color = "black")
```



Check Important variables contributing to PC1 and PC2 for HSI msc+SG+1st Derivative Spectra

```
# Let us extract the loadings from the PCA
loadings <- pca_msc_sg_1d$var$coord

# Sort the loadings for PC1 and PC2
top_PC1 <- head(sort(abs(loadings[, 1]), decreasing = TRUE), 10) # Top 10 for PC1
top_PC2 <- head(sort(abs(loadings[, 2]), decreasing = TRUE), 10)  # Top 10 for PC2

# Combine into a data frame for easy viewing
top_PC1 <- as.data.frame(top_PC1)
top_PC2 <- as.data.frame(top_PC2)

# Display the top contributing variables
#kable(top_PC1)
kable(top_PC2)
```

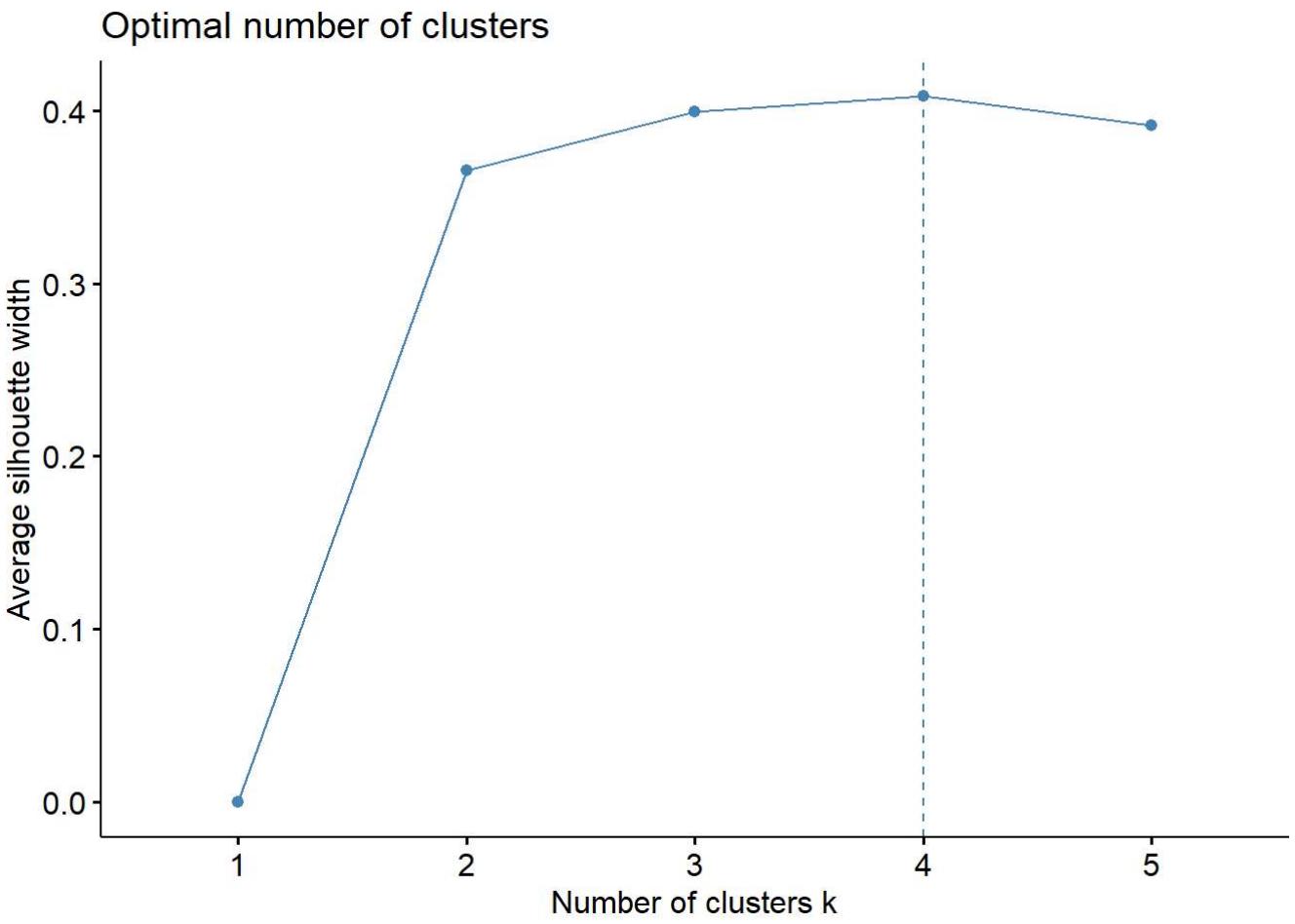
	top_PC2
X1570.310059	0.9759730
X1573.869995	0.9735652
X1566.75	0.9715574
X1563.199951	0.9647711

	top_PC2
X1559.640015	0.9574164
X1577.430054	0.9541198
X1556.089966	0.9494329
X1552.530029	0.9425514
X1548.97998	0.9381158
X1545.420044	0.9332511

- According to the x-loadings, Spectral wavelengths in the regions **1060-1088 nm**, and **1414 nm** contribute to the variance in PC1. The majority of the variation in PC2 is contributed by regions **1548 nm to 1580 nm**.

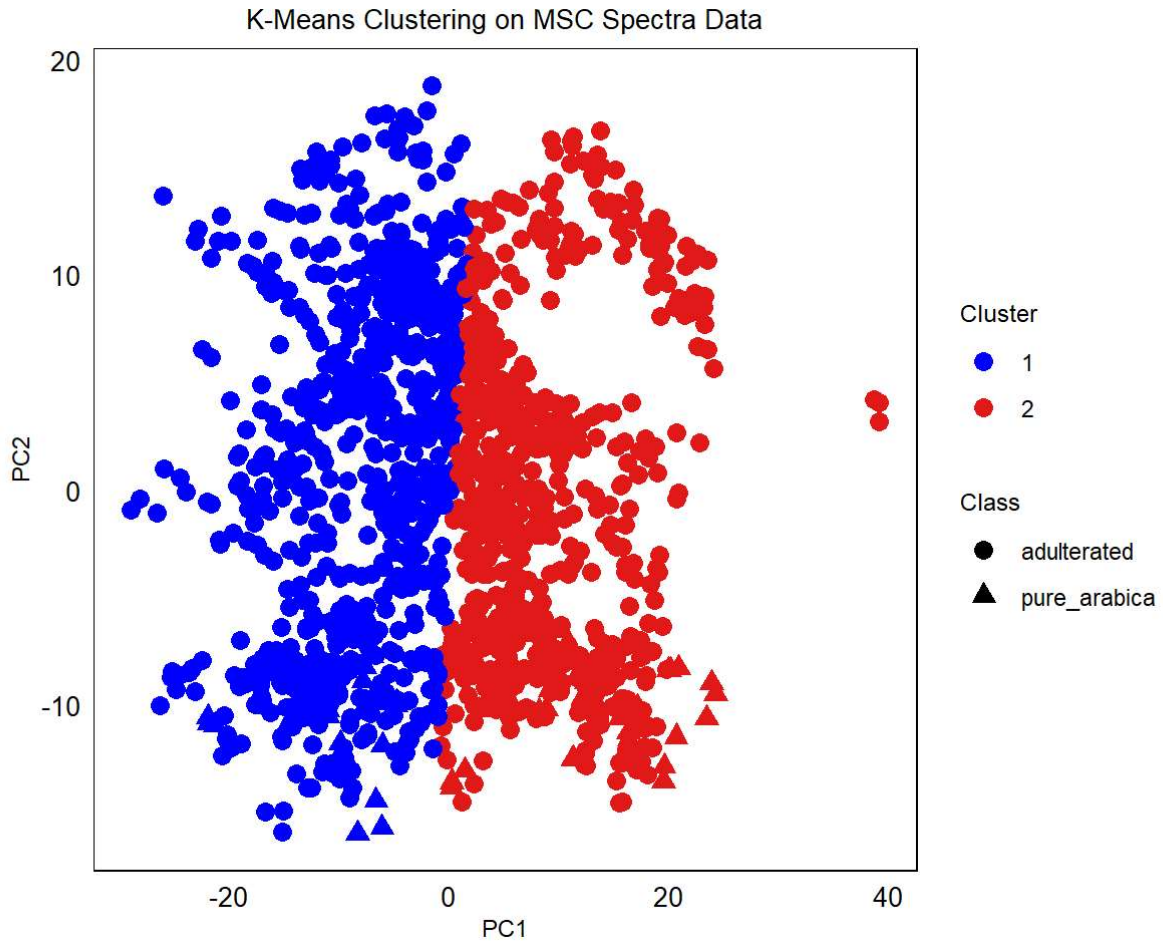
K-Means Clustering: MSC+SG+1st Derivative Spectra

```
# Use PCA reduced data to first determine the number of clusters  by silhouette method

# optimal number of clusters for HSI
msc_sg_1d_clust<-fviz_nbclust(Pcs_hsi_msc_sg_1d[,c(6,7)],
                        kmeans, method = "silhouette", k.max=5)

print(msc_sg_1d_clust) #determine the number of clusters
```



```
# Perform K-Means Classification using the optimal number of clusters
msc_sg_1d_kmeans <- kmeans(Pcs_hsi_msc_sg_1d[,c(6,7)],centers = 2,iter.max = 10,nstart = 25)
msc_sg_1d_clusters <- as.factor(msc_sg_1d_kmeans$cluster) # Extract clusters
```

```
# Add the cluster assignments to the data frame
Pcs_hsi_msc_sg_1d$Cluster <- msc_sg_1d_clusters

# Visualize the clusters
ggplot(Pcs_hsi_msc_sg_1d, aes(x = PC1, y = PC2, color = Cluster, shape = as.factor(binary_clas
  geom_point(size = 3) +
  labs(x = "PC1", y = "PC2", title = "K-Means Clustering on MSC Spectra Data", shape = 'Class'
 scale_color_manual(values = c("blue", "#e31a1c",  "#33a02c",  "#6a3d9a")) +
  theme_minimal() +
  theme(panel.border = element_rect(color = 'black', fill = NA),
        panel.grid = element_blank(),
        axis.text.x = element_text(color = 'black', size = 10),
        axis.text.y = element_text(color = 'black', size = 10),
        aspect.ratio = 1,
        axis.title.x = element_text(size = 9),
        axis.title.y = element_text(size = 9),
        legend.title = element_text(size = 9),
        plot.title = element_text(size = 10, hjust = 0.5))
```



Confusion Matrix - MSC+SG+1D Spectra

```
# Create a confusion matrix
cfmatrix_msc_sg_1d<-confusionMatrix(as.factor(msc_sg_1d_clusters),as.factor(class_k))

cfmatrix_msc_sg_1d #display the confusion matrix
```

```
Confusion Matrix and Statistics

          Reference
Prediction   1   2
         1 742  27
         2 652  48

               Accuracy : 0.5378
                 95% CI : (0.5119, 0.5635)
    No Information Rate : 0.9489
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0349

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.53228
            Specificity : 0.64000
         Pos Pred Value : 0.96489
         Neg Pred Value : 0.06857
             Prevalence : 0.94894
         Detection Rate : 0.50511
   Detection Prevalence : 0.52349
      Balanced Accuracy : 0.58614

       'Positive' Class : 1
```

```
kable(cfmatrix_msc_sg_1d$byClass)
```

	x
Sensitivity	0.5322812
Specificity	0.6400000
Pos Pred Value	0.9648895
Neg Pred Value	0.0685714
Precision	0.9648895
Recall	0.5322812
F1	0.6860841
Prevalence	0.9489449
Detection Rate	0.5051055
Detection Prevalence	0.5234854
Balanced Accuracy	0.5861406

Multiplicative Scatter Correction (MSC), Savitzky_Golay and 2nd Derivative Spectral Data

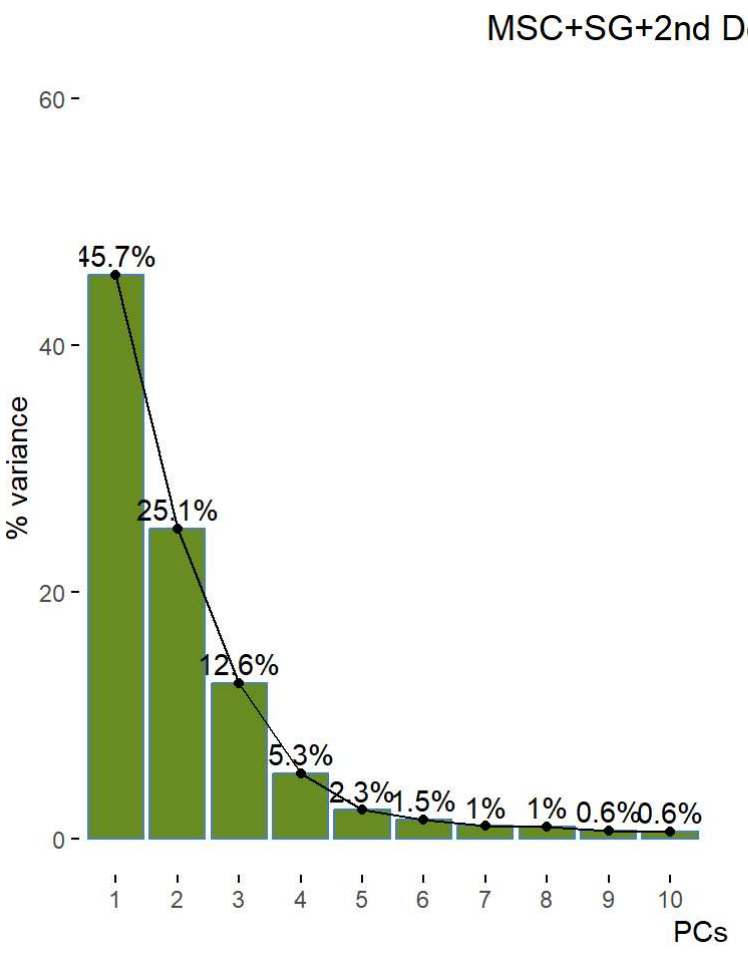
PCA Analysis: MSC+SG+2D pre-treated data

```
pca_msc_sg_2d <- PCA(hsi_msc_2d[,c(6:229)],ncp = 60, graph = FALSE)

pca_msc_sg_2d$eig[1:60,] # Extract the first 5 component eigenvalues
```

	eigenvalue	percentage of variance	cumulative percentage of variance
comp 1	102.34164725	45.688235380	45.68824
comp 2	56.23917739	25.106775622	70.79501
comp 3	28.12332498	12.555055794	83.35007
comp 4	11.83926123	5.285384479	88.63545
comp 5	5.14033861	2.294794024	90.93025
comp 6	3.40225587	1.518864227	92.44911
comp 7	2.19433449	0.979613613	93.42872
comp 8	2.14452721	0.957378217	94.38610
comp 9	1.34007667	0.598248515	94.98435
comp 10	1.27257958	0.568115885	95.55247
comp 11	0.98363874	0.439124437	95.99159
comp 12	0.97337223	0.434541175	96.42613
comp 13	0.78102753	0.348673006	96.77480
comp 14	0.63964557	0.285556056	97.06036
comp 15	0.57317648	0.255882355	97.31624
comp 16	0.51764738	0.231092579	97.54734
comp 17	0.49477825	0.220883147	97.76822
comp 18	0.41465542	0.185114026	97.95333
comp 19	0.37182463	0.165993137	98.11933
comp 20	0.32957483	0.147131619	98.26646
comp 21	0.31998529	0.142850576	98.40931
comp 22	0.31247275	0.139496764	98.54880
comp 23	0.26850735	0.119869351	98.66867
comp 24	0.24520947	0.109468513	98.77814
comp 25	0.21083118	0.094121063	98.87226
comp 26	0.20579936	0.091874715	98.96414
comp 27	0.17705178	0.079040975	99.04318
comp 28	0.14560205	0.065000913	99.10818
comp 29	0.14373664	0.064168143	99.17235
comp 30	0.13069190	0.058344596	99.23069
comp 31	0.11819929	0.052767542	99.28346
comp 32	0.10709073	0.047808361	99.33127
comp 33	0.09533642	0.042560902	99.37383
comp 34	0.09000784	0.040182072	99.41401
comp 35	0.08437674	0.037668188	99.45168
comp 36	0.07534527	0.033636279	99.48532
comp 37	0.07066937	0.031548827	99.51687
comp 38	0.06370062	0.028437778	99.54530
comp 39	0.05768236	0.025751055	99.57105
comp 40	0.05592923	0.024968407	99.59602
comp 41	0.05084992	0.022700859	99.61872
comp 42	0.04930622	0.022011704	99.64073
comp 43	0.04315988	0.019267803	99.66000

```
comp 44   0.04309528            0.019238962                    99.67924
comp 45   0.04173677            0.018632487                    99.69787
comp 46   0.03840581            0.017145452                    99.71502
comp 47   0.03490897            0.015584361                    99.73060
comp 48   0.03323527            0.014837173                    99.74544
comp 49   0.03228742            0.014414027                    99.75986
comp 50   0.02790695            0.012458460                    99.77231
comp 51   0.02710207            0.012099137                    99.78441
comp 52   0.02592902            0.011575454                    99.79599
comp 53   0.02434203            0.010866977                    99.80686
comp 54   0.02314193            0.010331220                    99.81719
comp 55   0.02230230            0.009956383                    99.82714
comp 56   0.02078873            0.009280682                    99.83642
comp 57   0.01871388            0.008354412                    99.84478
comp 58   0.01759798            0.007856241                    99.85263
comp 59   0.01685969            0.007526647                    99.86016
comp 60   0.01644952            0.007343537                    99.86750
```

```r
# Scree Plot: Check the number of components to keep

fviz_eig(pca_msc_sg_2d, addlabels = TRUE, ylim = c(0, 60),
         xlim=c(1,20), main = 'MSC+SG+2nd Der. Spectra',barfill = "olivedrab",
         hjust = 0.5,xlab = "PCs", ylab = "% variance")+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(panel.grid = element_blank())
```



```
#Save the PCs in a new data frame
Pcs_hsi_msc_sg_2d <- as.data.frame(pca_msc_sg_2d$ind$coord[,c(1:60)])
colnames(Pcs_hsi_msc_sg_2d) <- c("PC1","PC2","PC3","PC4","PC5",
                                 "PC6","PC7","PC8","PC9","PC10",
                                 "PC11","PC12","PC13","PC14","PC15",
                                 "PC16","PC17","PC18","PC19","PC20",
                                 "P21","P22","P23","P24","PC25",
                                 "PC26","PC27","PC28","PC29","PC30",
                                 "PC31","PC32","PC33","PC34","PC35",
                                 "PC36","PC37","PC38","PC39","PC40",
                                 "PC41","PC42","PC43","PC44","PC45",
                                 "PC46","PC47","PC48","PC49","PC50",
                                 "PC51","PC52","PC53","PC54","PC55",
                                 "PC56","PC57","PC58","PC59","PC60")
Pcs_hsi_msc_sg_2d <- cbind(columns, Pcs_hsi_msc_sg_2d) # Bind with the columns from initial da
kable(head(Pcs_hsi_msc_sg_2d))
```

sample_id	binary_class	three_class	adult_percent	cal_val	PC1	PC2	PC3	PC4
Pure_Arabica_10	pure_arabica	pure_arabica	0	1	-3.4014005	12.37823	12.71737	-0.2013704
Pure_Arabica_3	pure_arabica	pure_arabica	0	1	2.0679176	15.73442	14.70386	-3.8089402
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-0.5808616	13.52767	17.64203	-0.4298545
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	-0.2053454	13.84743	18.65312	-0.5203377
Pure_Arabica_2	pure_arabica	pure_arabica	0	1	0.1140191	13.54787	14.52595	0.4152834
Pure_Arabica_25	pure_arabica	pure_arabica	0	2	-0.3183256	15.18330	17.89881	-3.0213725

```
#write.csv(Pcs_hsi_msc_sg_2d, file = "pcs_hsi_msc_sg_2d.csv", row.names = FALSE) #save PC data
```

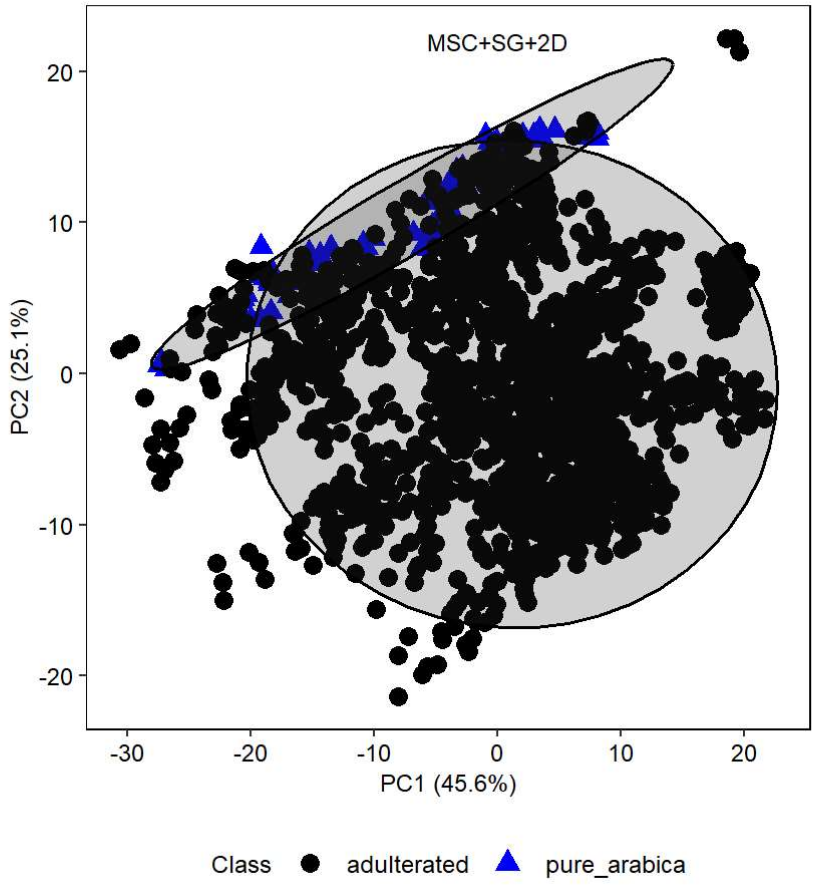
Insights: PCA-MSC+SG+2D spectral treatment

- The first five principal components account for about 90% of the variation in the data.

PCA Plot for MSC+SG+2D Spectra

```
# MSC+SG+2D spectra PCA Plot

Pcs_hsi_msc_sg_2d %>%
  ggplot(mapping = aes(x = PC1, y = PC2,
                       shape = as.factor(binary_class), color = binary_class)) +
  geom_point(size=3) +
  labs(x = "PC1 (45.6%)", y = "PC2 (25.1%)",
       title = "",shape = "Class", color = "Class") +
  theme_bw() +
  theme(
    panel.border = element_rect(color = 'black', fill = NA),
    panel.grid = element_blank(),
    axis.text.x = element_text(color = 'black', size = 9),
    axis.text.y = element_text(color = 'black', size = 9),
    aspect.ratio = 1,
```

```
    axis.title.x = element_text(size = 9),
    axis.title.y = element_text(size = 9),
    plot.title = element_text(size = 9, hjust = 0.5),
    legend.title = element_text(size = 9),
    legend.text = element_text(size = 9),
    legend.position = "bottom"
  ) +
  stat_ellipse(aes(group = binary_class),
               level = 0.95,
               geom = "polygon", alpha = 0.2,
               color = 'black', linewidth = 0.6)+
  scale_color_manual(values = c("pure_arabica" = "blue", "adulterated" = "black")) +
   annotate("text", x = 0, y = 22, label = "MSC+SG+2D", size = 3, color = "black")
```



Check Important variables contributing to PC1 and PC2 for HSI msc+SG+2nd Derivative Spectra

```
# Let us extract the loadings from the PCA
loadings <- pca_msc_sg_2d$var$coord

# Sort the loadings for PC1 and PC2
top_PC1 <- head(sort(abs(loadings[, 1]), decreasing = TRUE), 10) # Top 10 for PC1
top_PC2 <- head(sort(abs(loadings[, 2]), decreasing = TRUE), 10)  # Top 10 for PC2

# Combine into a data frame for easy viewing
top_PC1 <- as.data.frame(top_PC1)
```

```
top_PC2 <- as.data.frame(top_PC2)

# Display the top contributing variables
kable(top_PC1)
```

	top_PC1
X1333.219971	0.9672573
X1224.369995	0.9607636
X1252.410034	0.9558480
X1255.920044	0.9552578
X1336.73999	0.9521321
X1220.869995	0.9497682
X1248.900024	0.9442438
X973.650024	0.9434675
X1199.869995	0.9425191
X1259.420044	0.9419573

```
kable(top_PC2)
```

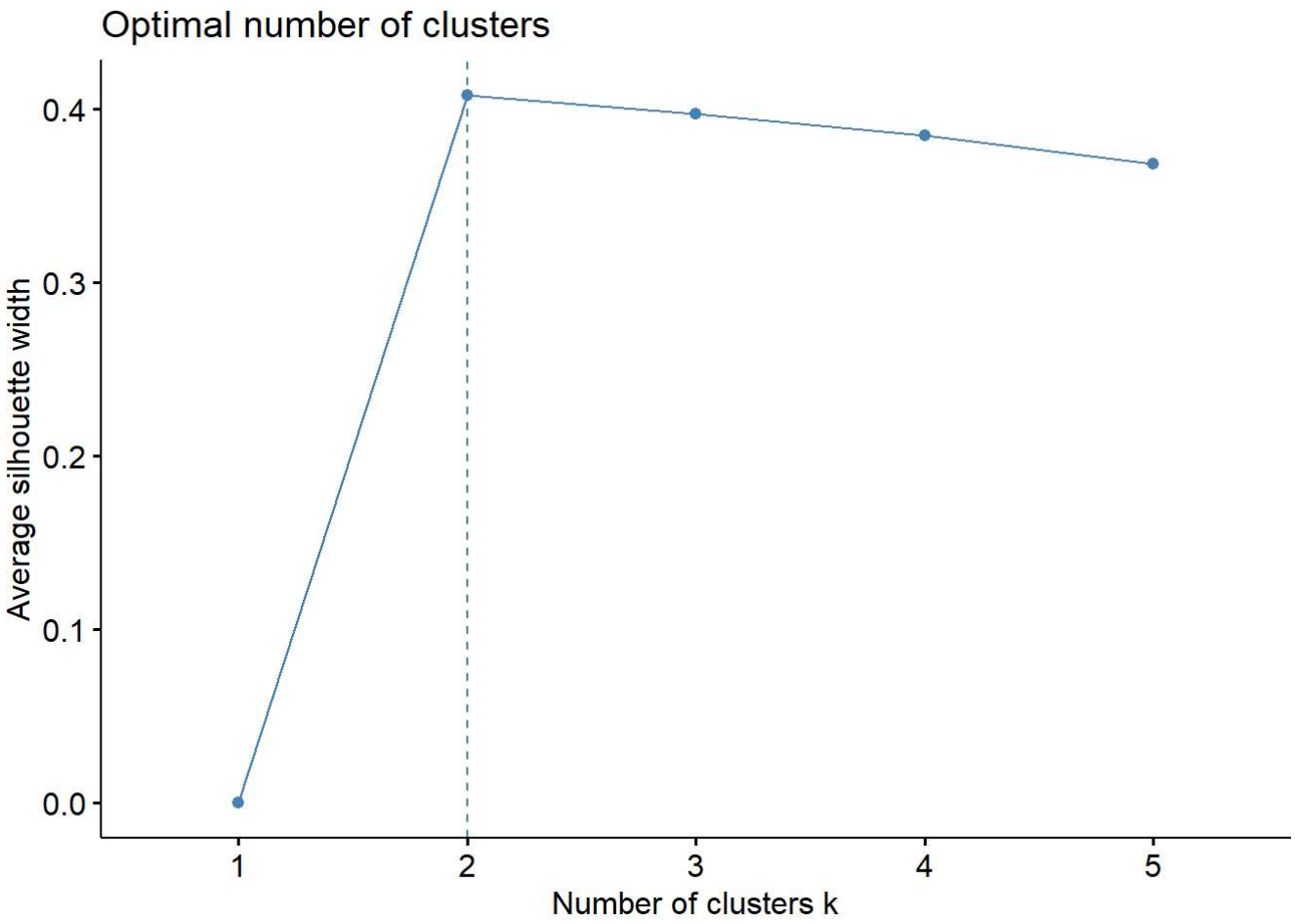
	top_PC2
X1424.920044	0.9700669
X1421.390015	0.9503190
X1287.51001	0.9392701
X1283.98999	0.9349863
X1389.609985	0.9279831
X1393.140015	0.9203150
X1291.02002	0.9181747
X1305.079956	0.9069544
X1308.589966	0.9051351
X1294.530029	0.9023163

- The variance in **PC1** is largely contributed by wavelengths regions around **1220-1259 nm** and **1333-1336 nm**, while regions **1287-1308 nm** and **1389 - 1424 nm** contribute to PC2.

K-Means Clustering: MSC+SG+2nd Derivative Spectra

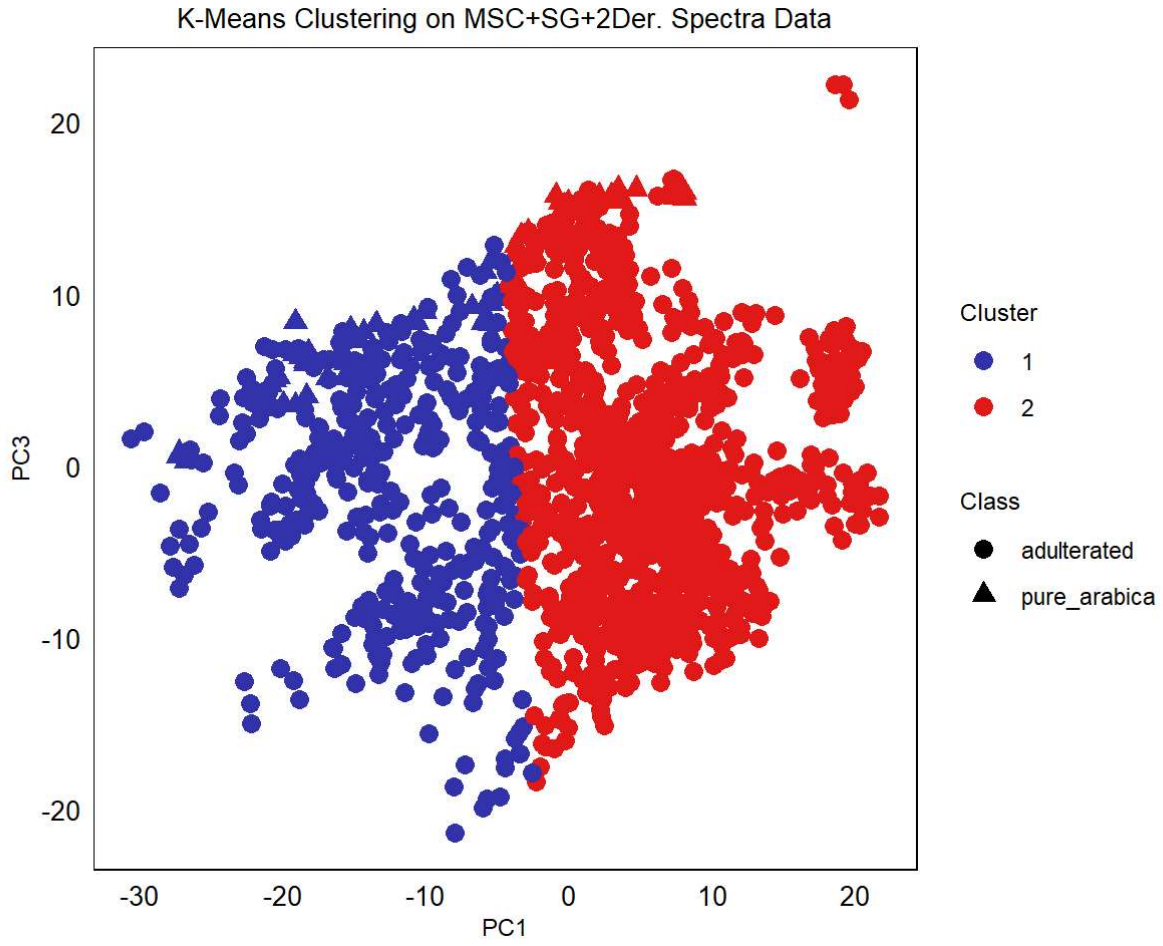
```
# Use PCA reduced data to first determine the number of clusters  by silhouette method

# optimal number of clusters for HSI
msc_sg_2d_clust<-fviz_nbclust(Pcs_hsi_msc_sg_2d[,c(6,7)],
                        kmeans, method = "silhouette", k.max=5)
```

```
print(msc_sg_2d_clust) #determine the number of clusters
```




```
# Perform K-Means Classification using the optimal number of clusters
msc_sg_2d_kmeans <- kmeans(Pcs_hsi_msc_sg_2d[,c(6,7)],centers = 2, iter.max = 10,nstart = 25)
msc_sg_2d_clusters <- as.factor(msc_sg_2d_kmeans$cluster) # Extract clusters
```

```
# Add the cluster assignments to the data frame
Pcs_hsi_msc_sg_2d$Cluster <- msc_sg_2d_clusters

# Visualize the clusters
ggplot(Pcs_hsi_msc_sg_2d, aes(x = PC1, y = PC2, color = Cluster, shape = as.factor(binary_clas
  geom_point(size = 3) +
  labs(x = "PC1", y = "PC3", title = "K-Means Clustering on MSC+SG+2Der. Spectra Data", shape =
 scale_color_manual(values = c("#33a",  "#e31a1c")) +
  theme_minimal() +
  theme(panel.border = element_rect(color = 'black', fill = NA),
        panel.grid = element_blank(),
        axis.text.x = element_text(color = 'black', size = 10),
        axis.text.y = element_text(color = 'black', size = 10),
        aspect.ratio = 1,
        axis.title.x = element_text(size = 9),
        axis.title.y = element_text(size = 9),
        legend.title = element_text(size = 9),
        plot.title = element_text(size = 10, hjust = 0.5))
```



Confusion Matrix - MSC+SG+1D Spectra

```
# Create a confusion matrix
cfmatrix_msc_sg_2d<-confusionMatrix(as.factor(msc_sg_2d_clusters),as.factor(class_k))

cfmatrix_msc_sg_2d #display the confusion matrix
```

```
Confusion Matrix and Statistics

          Reference
Prediction   1   2
         1 397  40
         2 997  35

               Accuracy : 0.2941
                 95% CI : (0.2709, 0.3181)
    No Information Rate : 0.9489
    P-Value [Acc > NIR] : 1

                  Kappa : -0.0353

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.28479
            Specificity : 0.46667
         Pos Pred Value : 0.90847
```

```
          Neg Pred Value : 0.03391
              Prevalence : 0.94894
          Detection Rate : 0.27025
    Detection Prevalence : 0.29748
       Balanced Accuracy : 0.37573

        'Positive' Class : 1
```

```
kable(cfmatrix_msc_sg_2d$byClass)
```

	x
Sensitivity	0.2847920
Specificity	0.4666667
Pos Pred Value	0.9084668
Neg Pred Value	0.0339147
Precision	0.9084668
Recall	0.2847920
F1	0.4336428
Prevalence	0.9489449
Detection Rate	0.2702519
Detection Prevalence	0.2974813
Balanced Accuracy	0.3757293

Conclusion

PCA and K-Means were successfully applied to both raw and preprocessed spectral data as part of the exploratory analysis. These unsupervised techniques provided initial insights into the underlying structure of the dataset by reducing dimensionality (PCA) and grouping samples based on similarity (K-Means). However, the results demonstrated that neither method could distinctly separate or enable effective visualization of the two expected clusters: pure and adulterated coffee samples.

This outcome suggests that the spectral differences between pure and adulterated coffee, even after preprocessing, might be subtle, complex, or nonlinear, making them challenging to capture using traditional unsupervised methods. The preprocessing techniques applied, including Standard Normal Variate (SNV), Savitzky-Golay smoothing, and derivatives], were aimed at enhancing signal quality and reducing noise but may require further optimization or combination with advanced feature extraction methods.

To overcome these challenges, supervised learning approaches such as Support Vector Machines (SVM), Partial Least Squares Discriminant Analysis (PLS-DA), or Random Forest will be explored. These methods utilize labeled data to maximize class separability, making them more suitable for distinguishing between pure and adulterated samples.

The findings underscore the limitations of relying solely on unsupervised methods and highlight the importance of leveraging supervised techniques to better exploit the data. The next steps will focus on applying advanced supervised methods to evaluate the ability of hyperspectral imaging techniques in detecting adulteration of Arabica coffee.