Clustering
& Dimension Reduction & Classification

Performed the Clustering Analysis

For the initial question, we first invoked the ***GEOquery*** library, which we had previously installed. Subsequently, we downloaded the **GSE13276** dataset using the ***getGEO*** function. Next, we defined the expression data of the dataset. As the **samples should be in columns**, we **transposed** the data using the ***t*** function and stored it in a variable. Following this, we applied **Pearson** correlation to the expression data using the ***as.dist*** function, saving the obtained results for later use. Moving on, we utilized the hclust function with the **centroid** method and assigned it to the h_samples variable. Finally, to obtain the **dendrogram**, we used the **plot(as.dendrogram(h_samples))** script and visualized the dendrogram associated with the GSE13276 dataset (Figure 1).

```r
#BSB513-Homework V: Clustering & Dimension Reduction & Classification

#Question a: Performed the Clustering Analysis

library(GEOquery)
gse13276 <- getGEO("GSE13276", AnnotGPL = TRUE) #using the getGEO function, downloaded the data
exprs_gse13276 <- exprs(gse13276[[1]]) #we defined the expression values on our dataset

#we manipulated the rows and cols. We defined the columns as the samples(genes)
t_gse13276 = t(exprs_gse13276) #samples on the columns in t_gse13276 arg

# We used to pearson correlation in as.dist function.
dist_samples <- as.dist(1-cor(exprs_gse13276, method = "pearson"))

# After, used to hclust function and in dist_samples and centroid method
h_samples <- hclust(dist_samples, method = "centroid")

#we created the dendrogram using the as.dendrogram fucntion
#after, visualize the h_samples dendrogram
plot(as.dendrogram(h_samples))
```

Figure 1. Performed the Clustering Analysis for GSE13276 dataset in R.

Samples from the same group, as seen in Figure 2, are grouped more closely together, resulting in shorter dendrograms. For instance, Core Samples GSM335185, GSM335186, GSM335187, GSM335188, and GSM335189 exhibit significant proximity, clustering on the right side, while being distanced from other groups.

Similarly, within the White Matter Control group, samples are closely connected with shorter dendrograms. Additionally, they show proximity to GBM surrounding tissue samples. Upon a

comprehensive assessment of the entire dendrogram, a clear interpretation can be made that tumor samples are predominantly positioned on the right side.

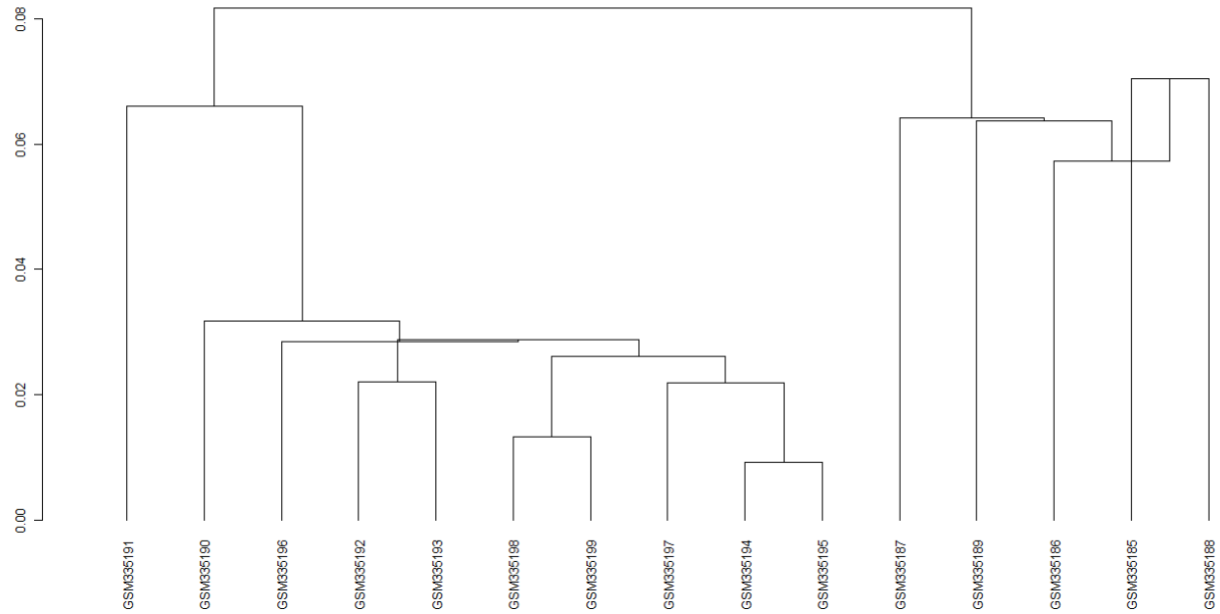

Figure 2. GSE13276 Expression Data Dendrogram Display

Performed the Principal Component Analysis (PCA)

Fort he question b, firstly, we installed the ***ggplot2*** package and called the library, as it is necessary for creating the plot we aim to obtain. Subsequently, we applied the ***prcomp*** function to the dataset by using the transposed data, where samples are in columns. The input for the prcomp function should contain observations in rows and variables in columns. To visualize the variation explained by each principal component, we used **summary(pca_result)** (Figure 3).

```
Importance of components:
                          PC1     PC2     PC3     PC4     PC5     PC6     PC7     PC8     PC9    PC10    PC11
Standard deviation     3.5784 0.73779 0.42929 0.33524 0.28605 0.24613 0.23615 0.20586 0.18415 0.17763 0.14034
Proportion of Variance 0.9146 0.03888 0.01316 0.00803 0.00584 0.00433 0.00398 0.00303 0.00242 0.00225 0.00141
Cumulative Proportion  0.9146 0.95352 0.96669 0.97471 0.98056 0.98489 0.98887 0.99190 0.99432 0.99657 0.99798
                          PC12    PC13    PC14
Standard deviation     0.11474 0.09164 0.08211
Proportion of Variance 0.00094 0.00060 0.00048
Cumulative Proportion  0.99892 0.99952 1.00000
```

Figure 3. Output of Summary(pca_result). Importance of Components.

Following that, we utilized the plot function to draw a two-dimensional PCA plot of the scores. To add labels to the plot, we employed the text function. In this context, PC1 and PC2 were utilized. We defined t_gse13276 as the Row names variable and specified pos=3 to ensure that the names are positioned on the right side (Figure 4).

```
22  #Question b: Performed the Principal Component Analysis (PCA)
23
24  install.packages("ggplot2")  #for created the plots
25  library(ggplot2)
26
27  pca_result <- prcomp(t_gse13276, center = TRUE, scale = TRUE)
28  summary(pca_result)
29  plot(pca_result$x[, 1], pca_result$x[, 2], xlab = "PC1", ylab = "PC2")
30  text(pca_result$x[, 1], pca_result$x[, 2], rownames(t_gse13276), pos = 3)
```

Figure 4. Performed teh Principal Componetn Analysis (PCA) in R.

In the PCA plot we obtained, Core Samples GSM335185, GSM335186, GSM335187, GSM335188, and GSM335189 are once again positioned in a distinct manner compared to other samples. The proximity of other samples to each other, as depicted in the PCA plot, is indicative of the logical consistency of the results. In other words, the clustering results align with the findings in the PCA plot, mutually reinforcing the validity of both outcomes (Figure 5).

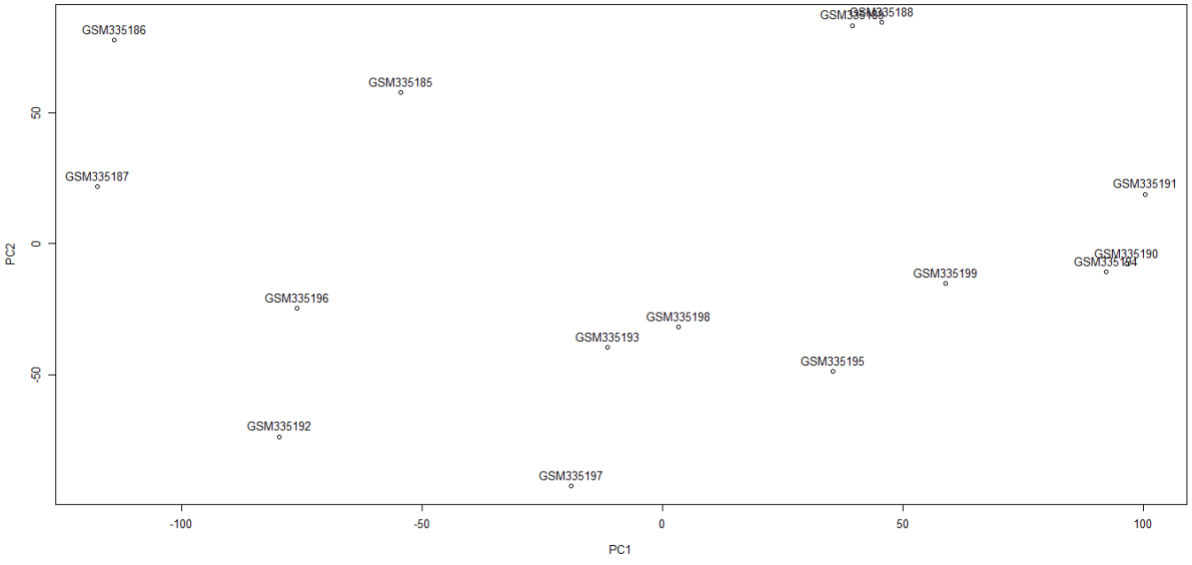

Figure 5. PCA Plot for GSE13276 Dataset.

Classification with Support Vector Machine (SVM)

For this stage, we initially installed the **e1071** package and subsequently called the library. Then, we defined **train_data**, where the first 5 samples are tumor samples, and the next 5 samples are surrounding tissue samples. We then assigned **0** for **Tumor Samples** and **1** for **Surrounding Tissue** as labels. As for **test_data**, we assigned the samples that we already know belong to the surrounding tissue (in the 11th and 12th rows).

Within the **svm** function, using train_data and label information, we created an **SVM model** and designated it as **svm_model**. To visualize the model, we used the print and summary functions. For the testing phase, we employed the predict function with svm_model and train data to validate the model. Subsequently, we applied our model to test_data (Figure 6).

```
#Classification with SVM
install.packages("e1071")
library(e1071)

#tumor sample 0, sur_tissue 1
train_data <- rbind(t_gse13276[1:5,], t_gse13276[6:10,])
label <- c(rep(0, 5), rep(1 ,5))

test_data <- rbind(t_gse13276[11,], t_gse13276[12,])

svm_model <- svm(train_data, label)
print(svm_model)
summary(svm_model)

predictions <- predict(svm_model, train_data)  #test with train data
table(predictions, label) #check accuracy

check <- predict(svm_model, test_data)

cm <- table( check)
cm
```

Figure 6. Classification with SVM for GSE13276 Dataset.

Finally, our 11th and 12th samples, GSM335195 and GSM335196, have been classified as 1 according to the SVM model. We had designated Surrounding Tissue samples as 1. Therefore, based on this result, we can conclude that our SVM model has led us to the correct outcome (Figure 7).

```
check
0.448067162041307 0.578503401331383
                1                 1
```

Figure 7. svm_model Test Result.