MA321-7-SP (2023-2024) APPLIED STATISTICS ASSIGNMENT

Assignment template

Melek Kuru (mk23930@essex.ac.uk)

Task 1: Data reading and simple exploration (25%)

```r
# Read the data into RStudio (or R) using the read.csv R command
InitialData<-read.csv(file="gene-expression-invasive-vs-noninvasive-cancer.csv")

# Check using the str, dim and dimnames command it worked - parts of the
#generated output are added as comments; lines starting
#with "#" comments and not R code.



#
# 'data.frame': 78 obs. of  4949 variables:
# $ J00129        : num  -0.448 -0.48 -0.568 -0.819 -0.112 -0.391 -0.624 -0.528 -0.811 -0.839 ...
# $ Contig29982_RC: num  -0.296 -0.512 -0.411 -0.267 -0.67 -0.31 -0.12 -0.447 -0.536 2 ...
# ...
#

dim(InitialData)
```

```
## [1]   78 4949
```

```r
#
# [1]   78 4949
#

dimnames(InitialData)[[2]][4947:4949]
```

```
## [1] "NM_000898" "AF067420"  "Class"
```

```r
#
# [1] "NM_000898" "AF067420"  "Class"
#
#
# The data set has
#    78 rows (patients)
#4949 columns with 4948 gene expression measurement of cancer tissue,
#each column representing a 'gene'  with column 4949 having the information
#of a class variable with two values: 1 and 2.

table(InitialData[4949])
```

```
## Class
##  1  2
## 34 44
```

```
#
# Class
# 1  2
# 34 44
#

# To gurantee that your data narrative is individual we ask you to set
# the seed of the random  number generator with R function set.seed using
# your Registration Number; for example "2244222"
# - replace by your registration number

#
#  [1] 4136 4878  175 1409 3208   69 1571 4844  211 3325
#
```

```
set.seed(2315873)
#
#Note: R function set.seed generates no R output.
#Make sure that you run your final data analysis with the submitted R code and
#that all tables and figures are generated by the final code.
#You may use R Mark Down or similar R tools to support this.

# Selects your random individual subset of 10 genes
index <- sample(4948, size=10)
mynewdata <- InitialData[, index]
dim(mynewdata)
```

```
## [1] 78 10
```

```
#1-(a) Compute the variance, co-variance and correlation matrix of your random
#subset of 10 genes. Add an appropriate table to your report
#install.packages("xtable")
#library(xtable)

#Variance matrix for the 10 genes
variances <- apply(mynewdata, 2, var)
variances
```

```
##      NM_006517 Contig39153_RC      NM_018002 Contig53098_RC Contig57825_RC
##     0.02344914     0.10869386     0.02689419     0.05424863     0.11229481
## Contig47810_RC      NM_002125         X66945      NM_015364      NM_014873
##     0.03196439     0.08620006     0.06081737     0.04240068     0.03564606
```

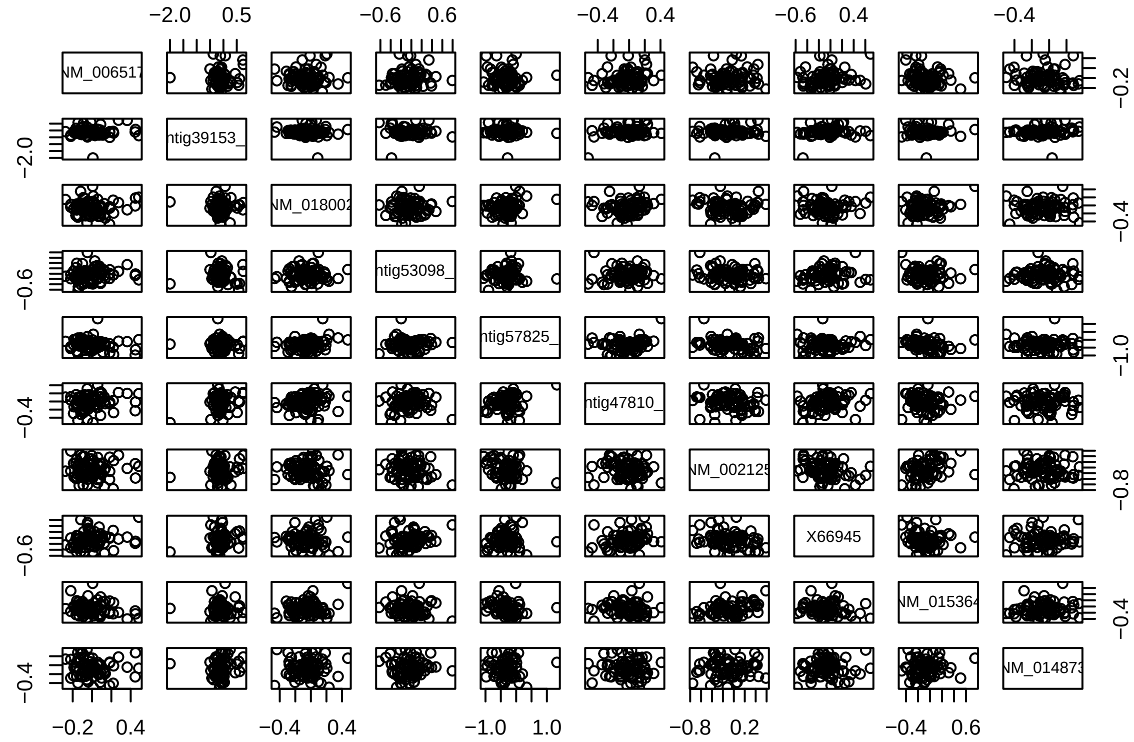
```
#xtable(variances)

# Covariance matrix for the 10 genes
covarience <- round(cov(mynewdata),4)
covarience
```

```
##                NM_006517 Contig39153_RC NM_018002 Contig53098_RC Contig57825_RC
## NM_006517         0.0234         0.0034    0.0005         0.0041        -0.0050
## Contig39153_RC    0.0034         0.1087   -0.0022        -0.0027        -0.0075
## NM_018002         0.0005        -0.0022    0.0269         0.0021         0.0137
## Contig53098_RC    0.0041        -0.0027    0.0021         0.0542         0.0027
## Contig57825_RC   -0.0050        -0.0075    0.0137         0.0027         0.1123
## Contig47810_RC    0.0041         0.0173    0.0045         0.0021         0.0156
## NM_002125         0.0006         0.0099   -0.0033        -0.0110        -0.0274
## X66945            0.0085         0.0140   -0.0006         0.0100         0.0060
## NM_015364        -0.0065        -0.0022    0.0063        -0.0044        -0.0157
## NM_014873        -0.0027         0.0079    0.0005         0.0017        -0.0013
##                Contig47810_RC NM_002125  X66945 NM_015364 NM_014873
## NM_006517              0.0041    0.0006  0.0085   -0.0065   -0.0027
## Contig39153_RC         0.0173    0.0099  0.0140   -0.0022    0.0079
## NM_018002              0.0045   -0.0033 -0.0006    0.0063    0.0005
## Contig53098_RC         0.0021   -0.0110  0.0100   -0.0044    0.0017
## Contig57825_RC         0.0156   -0.0274  0.0060   -0.0157   -0.0013
## Contig47810_RC         0.0320   -0.0076  0.0110   -0.0024   -0.0005
## NM_002125             -0.0076    0.0862 -0.0192    0.0195   -0.0007
## X66945                 0.0110   -0.0192  0.0608   -0.0087    0.0031
## NM_015364             -0.0024    0.0195 -0.0087    0.0424    0.0045
## NM_014873             -0.0005   -0.0007  0.0031    0.0045    0.0356
```

```
#xtable(covarience)

# Correlation matrix for the 10 genes
correlation<- round(cor(mynewdata),4)
correlation
```

```
##                NM_006517 Contig39153_RC NM_018002 Contig53098_RC Contig57825_RC
## NM_006517         1.0000         0.0674    0.0201         0.1141        -0.0975
## Contig39153_RC    0.0674         1.0000   -0.0406        -0.0354        -0.0678
## NM_018002         0.0201        -0.0406    1.0000         0.0562         0.2499
## Contig53098_RC    0.1141        -0.0354    0.0562         1.0000         0.0347
## Contig57825_RC   -0.0975        -0.0678    0.2499         0.0347         1.0000
## Contig47810_RC    0.1481         0.2927    0.1534         0.0514         0.2601
## NM_002125         0.0144         0.1023   -0.0690        -0.1608        -0.2783
## X66945            0.2251         0.1719   -0.0158         0.1736         0.0721
## NM_015364        -0.2053        -0.0317    0.1869        -0.0911        -0.2279
## NM_014873        -0.0929         0.1276    0.0174         0.0397        -0.0198
##                Contig47810_RC NM_002125  X66945 NM_015364 NM_014873
## NM_006517              0.1481    0.0144  0.2251   -0.2053   -0.0929
## Contig39153_RC         0.2927    0.1023  0.1719   -0.0317    0.1276
## NM_018002              0.1534   -0.0690 -0.0158    0.1869    0.0174
## Contig53098_RC         0.0514   -0.1608  0.1736   -0.0911    0.0397
## Contig57825_RC         0.2601   -0.2783  0.0721   -0.2279   -0.0198
## Contig47810_RC         1.0000   -0.1442  0.2490   -0.0645   -0.0143
## NM_002125             -0.1442    1.0000 -0.2651    0.3233   -0.0131
## X66945                 0.2490   -0.2651  1.0000   -0.1707    0.0657
## NM_015364             -0.0645    0.3233 -0.1707    1.0000    0.1159
## NM_014873             -0.0143   -0.0131  0.0657    0.1159    1.0000
```

```r
#xtable(correlation_matrix)

#Use the scatter plot
pairs(mynewdata[,c(1:10)])
```



THE EXPLANATION FOR QUESTION 1 - A:

VARIANCE: Varaince quantifies how widely distributed or variable a set of data points are with respect to their mean, or average, value. Genes with low variance such as NM_001921: 0.01858699 have consistent expression levels across samples, indicating stable expression that may be less influenced by the conditions being studied. Genes with medium variance such as AJ270996: 0.04112909 show moderate variability in expression across samples, suggesting some influence of experimental conditions on gene expression, but not as pronounced as high variance genes. Genes with high variance such as NM_003359: 0.08727705 exhibit significant variability in expression, indicating a strong response to study conditions or involvement in diverse biological processes. These genes could be of interest for further study due to their potential link to phenotypic differences, disease states, or treatment responses.

COVARIANCE: Covariance measures the directional relationship between the returns on two assets. A positive covariance means asset returns move together, while a negative covariance means they move inversely. The relationships between the expression levels of ten different genes are displayed in this covariance matrix. Covariance quantifies the combined change in two variables. When two genes have positive values, their expression levels typically rise or fall together; when two genes have negative values, one gene's expression tends to rise while the other's tends to fall. Because covariance depends on the scale of the variables, its magnitude cannot be used to determine the strength of the relationship.As an example from the output, if there is a positive covariance value, like 0.0209, between Contig46289_RC and NM_003359, it indicates that the expression levels of these two genes tend to increase together.A negative covariance value, such as -0.0228 between AJ270996 and NM_003359, indicates that the expression levels of these two genes change in opposite directions.A very low covariance value, like -0.0001 between Contig36530_RC and AJ270996, suggests there is no significant relationship between the expression levels of these two genes.

CORRELATION: Correlation is a statistical method that measures the direction and strength of the relationship between two or more variables. Values close to +1 indicate a strong positive relationship between the variables. As the value of one variable increases, the value of the other variable also increases. Values close to -1 indicate a strong negative relationship between the variables. As the value of one variable increases, the value of the other variable decreases. Values close to 0 indicate that there is no significant linear relationship between the variables. For example NM_001921 and Contig46289_RC have a correlation of 0.4376, indicating a strong positive relationship. AJ270996 and NM_003359 have one of the highest negative correlations (-0.3798), indicating a significant inverse relationship that may point to competing regulatory roles in specific biological processes.Genes such as Contig36530_RC and AJ270996, whose correlations are close to 0, probably function independently in the cellular environment and show no discernible linear relationship in the levels of expression between the samples.

```
#1-(b) Using R to calculate the distance matrix of your random subset
#of 10 genes. Add an appropriate table to your report.

#Identify and present the scaled distance matrix.
distance<- dist(scale(mynewdata, center = FALSE))
distance_matrix<- as.dist(round(as.matrix(distance), 2)[1:13, 1:13])
distance_matrix
```

```
##       1    2    3    4    5    6    7    8    9    10   11   12
## 2  2.39
## 3  3.61 4.16
## 4  3.07 2.90 4.72
## 5  3.89 4.23 6.53 3.22
## 6  3.07 2.96 3.90 4.37 5.08
## 7  2.79 3.04 2.76 3.96 5.26 3.47
## 8  3.32 3.76 5.58 2.17 3.89 5.13 4.87
## 9  1.77 2.20 2.80 3.45 4.79 2.87 2.82 4.10
## 10 2.63 3.27 3.57 3.88 4.12 2.13 3.08 4.68 2.91
## 11 1.71 2.27 4.17 3.22 3.76 3.13 3.83 3.73 2.08 2.86
## 12 2.82 4.22 4.79 3.86 4.72 4.95 4.06 4.20 3.80 4.61 3.28
## 13 3.76 4.05 3.81 5.34 6.48 2.56 3.54 6.02 3.24 3.96 4.37 4.89
```

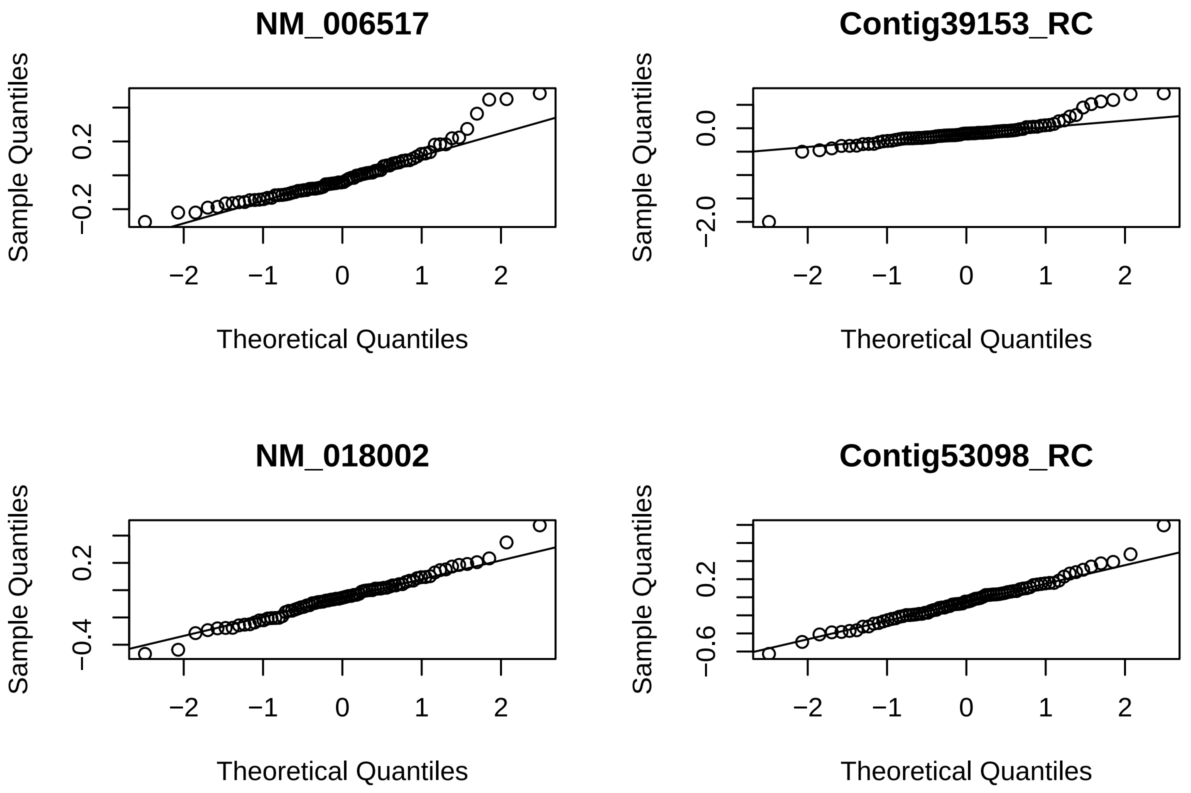
```
# xtable(distancesDf, include.rownames = TRUE, include.colnames = TRUE)
```

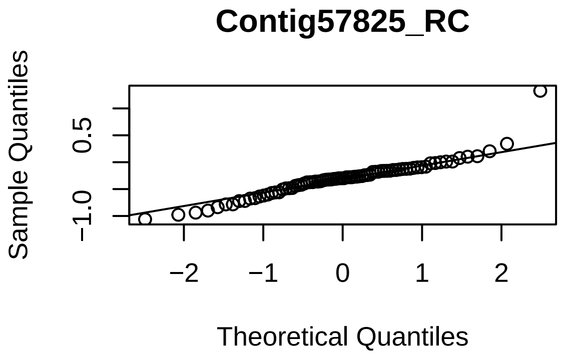
THE EXPLANATION FOR QUESTION 1 - B:

The concept of distance measures the closeness of observations.We are interested to measure how similar or how close are the multivariate observations. The distance matrix reveals the relationships between the expression levels of a subset of 10 genes. Closer distances suggest similar expression patterns, while larger distances indicate greater dissimilarity. For instance, genes 1 and 7 show a remarkably small distance of 2.15, which may imply a close relationship in their expression across samples. On the other hand, genes 1 and 8 display a distance of 6.04, suggesting significant differences in their expression profiles. These distance measurements can provide insights into potential gene expression regulation mechanisms or pathways shared among similar genes. Further investigation could be warranted for gene pairs with notably small or large distances to understand their biological connections or functional similarities.

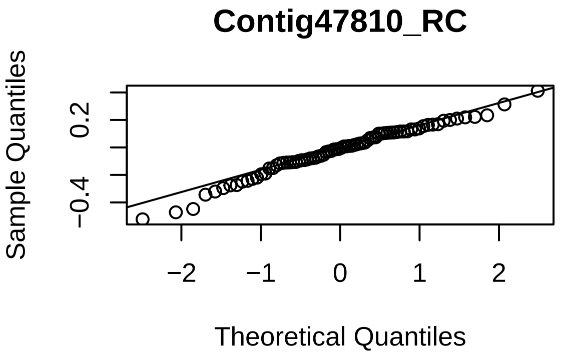
```
#1-(c) Using R to calculate univariate Q-Q-plots and a Q-Q-plot based on
#the generalised distance for the observations of your random subset of 10 genes
#Add appropriate figures to your report.
```

```
#Q-Q Plots
columns_mean <- colMeans(mynewdata)
generalized_distance <- apply(mynewdata, MARGIN = 1, function(x)
  t(x - columns_mean) %*% solve(covarience) %*% (x - columns_mean))
par(mfrow=c(2,2))
for(gene in names(mynewdata)) {
  qqnorm(mynewdata[[gene]], main = gene)
  qqline(mynewdata[[gene]])
}
```

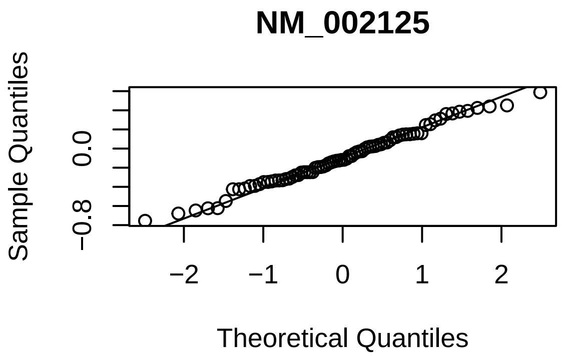




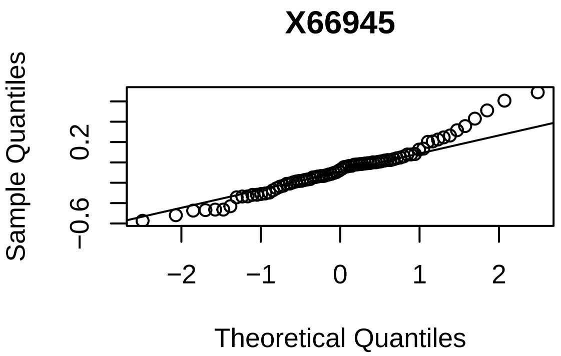
Contig57825_RC



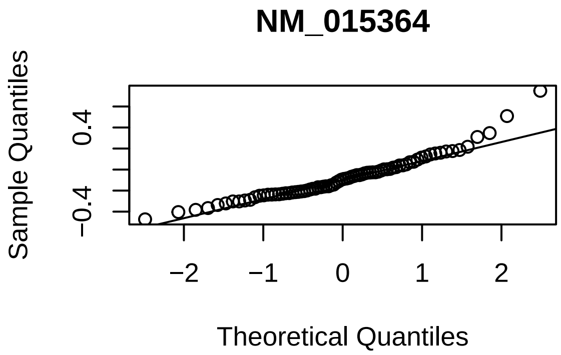
Contig47810_RC

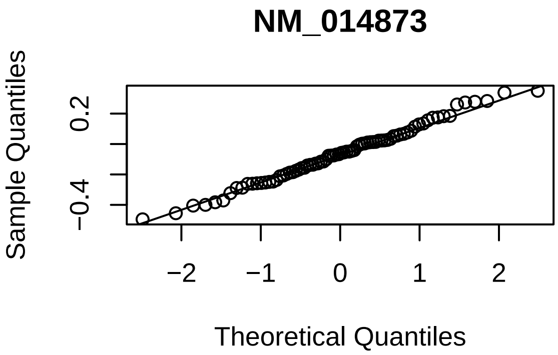


NM_002125



X66945





THE EXPLANATION FOR QUESTION 1 - C:

The quantile-quantile plot, also known as the QQ plot, is a graphical tool that we can use to determine whether a set of data is likely to have come from a normal or exponential theoretical distribution. If data points line up closely with the reference line (the expected values from a normal distribution), it suggests that the gene expression data are approximately normally distributed. This is important for situations where statistical tests and analyses assume normality. NM_003561: The plot shows data points that follow the reference line closely except for slight deviations at the ends. This indicates that the gene expression for NM_003561 is mostly normally distributed with some outliers.

Contig2022_RC: The plot reveals a good alignment with the reference line across the distribution, suggesting that the gene expression for Contig2022_RC closely follows a normal distribution.

Contig56434_RC: This plot also displays data points that adhere well to the reference line, indicating that gene expression for Contig56434_RC is approximately normally distributed.

NM_013301: Similar to the others, this gene's expression data largely follow the expected normal line, with minor deviations at the tails, which is typical in biological data.

AF052159: The data points are well-aligned with the theoretical line, which suggests a normal distribution of expression levels for AF052159.

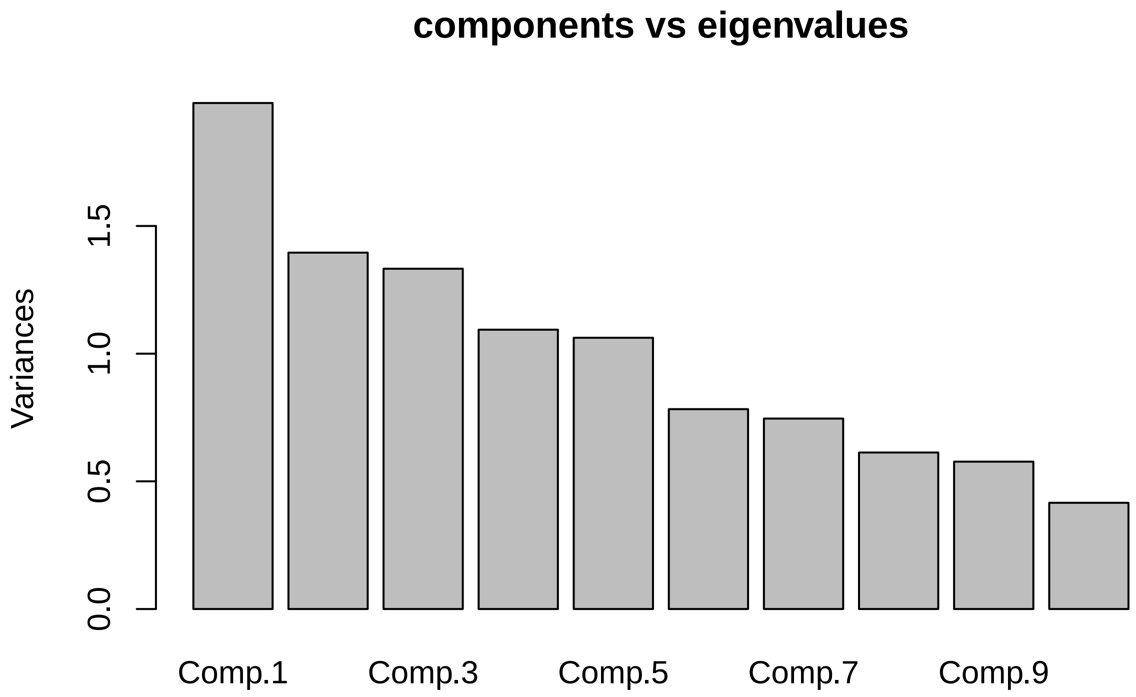
AB040900: The Q-Q plot shows the data points hugging the reference line tightly, indicating that the gene expression for AB040900 is well-modeled by a normal distribution, with minimal extreme values.

```
#2-Use R for a principal component analysis of your random subset of 10 genes.
#Add appropriate tables and figures to your report.

# Principal Component Analysis
```

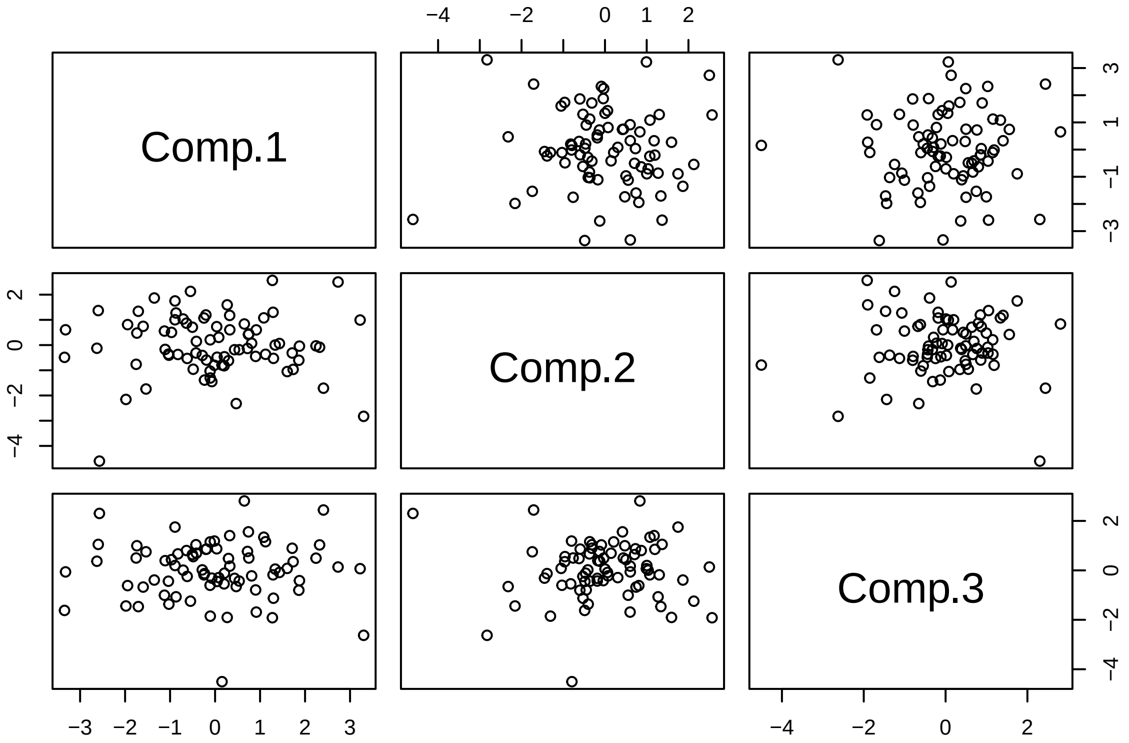
```r
pca <- princomp(mynewdata, cor = TRUE)
summary(pca, loadings = TRUE)
```

```
## Importance of components:
##                           Comp.1    Comp.2    Comp.3    Comp.4    Comp.5
## Standard deviation     1.4076478 1.1813229 1.1543519 1.0458922 1.0306445
## Proportion of Variance 0.1981472 0.1395524 0.1332528 0.1093890 0.1062228
## Cumulative Proportion  0.1981472 0.3376996 0.4709525 0.5803415 0.6865643
##                            Comp.6     Comp.7     Comp.8     Comp.9    Comp.10
## Standard deviation     0.88469134 0.86356741 0.78288035 0.75966030 0.64493712
## Proportion of Variance 0.07826788 0.07457487 0.06129016 0.05770838 0.04159439
## Cumulative Proportion  0.76483220 0.83940707 0.90069723 0.95840561 1.00000000
##
## Loadings:
##                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9
## NM_006517       0.245  0.381  0.215  0.315  0.442  0.480  0.257         0.139
## Contig39153_RC  0.142  0.548 -0.329        -0.319 -0.271  0.181 -0.514 -0.111
## NM_018002       0.119 -0.319 -0.523  0.182  0.433  0.231        -0.482
## Contig53098_RC  0.251         0.111 -0.387  0.573 -0.610  0.267
## Contig57825_RC  0.363 -0.475 -0.210  0.126 -0.227         0.282  0.314 -0.410
## Contig47810_RC  0.410  0.166 -0.397  0.278 -0.101 -0.207         0.435  0.472
## NM_002125      -0.442  0.315 -0.153  0.264  0.120         0.315  0.349 -0.515
## X66945          0.441  0.283        -0.201         0.153 -0.587        -0.531
## NM_015364      -0.392        -0.476         0.303        -0.428  0.207
## NM_014873              0.136 -0.327 -0.713 -0.110  0.435  0.339  0.176  0.107
##                Comp.10
## NM_006517       0.357
## Contig39153_RC  0.291
## NM_018002      -0.304
## Contig53098_RC
## Contig57825_RC  0.430
## Contig47810_RC -0.314
## NM_002125      -0.320
## X66945         -0.137
## NM_015364       0.530
## NM_014873
```

```r
# Scree plot
plot(pca, main='components vs eigenvalues')
```



```r
# Extract the scores of the observations in each principal component (1,2,3)
# to check the variance of the components
scores <- pca$scores
# scatter plot for the first three principal components
pairs(pca$scores[,1:3])
```



THE EXPLANATION FOR QUESTION 2:

A statistical method called Principal Component Analysis (PCA) finds the most important features that capture the most variance, reducing the dimensionality of the data and making analysis and visualization simpler. The first component (Comp.1) is the most significant source of variance among the ten genes analyzed, accounting for 19.8% of the total variance in our dataset. With the second (Comp.2) and third (Comp.3) components accounting for 13.95% and 13.33% of the variance, respectively, the remaining components explain progressively less variance. Approximately 47.1% of the total variance can be attributed to the first three components combined, which highlights most of the data in the dataset. The directionality of the loadings in either direction—positive or negative—speaks to patterns of gene expression related to components. Negative values indicate an inverse relationship between gene expression and the score of the component.For example, NM_002125 contributes negatively to Comp.1, suggesting that higher values of Comp.1 correspond to lower expression levels of this gene. Certain genes, such as X66945, and Contig53098_RC, play important roles in some components, suggesting that they have a major impact on the variance of the dataset in those dimensions.

```
#3-(a)Fit a multivariate analysis of variance model (MANOVA) to your random
#subset of 10 genes. Investigate if there is a difference between invasive
#(label 1) and noninvasive (label 2) cancer. Note: You need to add
#column 4949 containing the information invasive and noninvasive cancer to your
#random subset of 10 genes. Add appropriate tables and figures to your report.
# Ensure 'Class' column is included in your data frame for MANOVA analysis

set.seed(2315873)
my_index <- sample(1:4948, size = 10)
my_class_subset <- InitialData[, c(my_index,4949 )]
```

```
# check the subset's dimensions
dim(my_class_subset)
```

```
## [1] 78 11
```

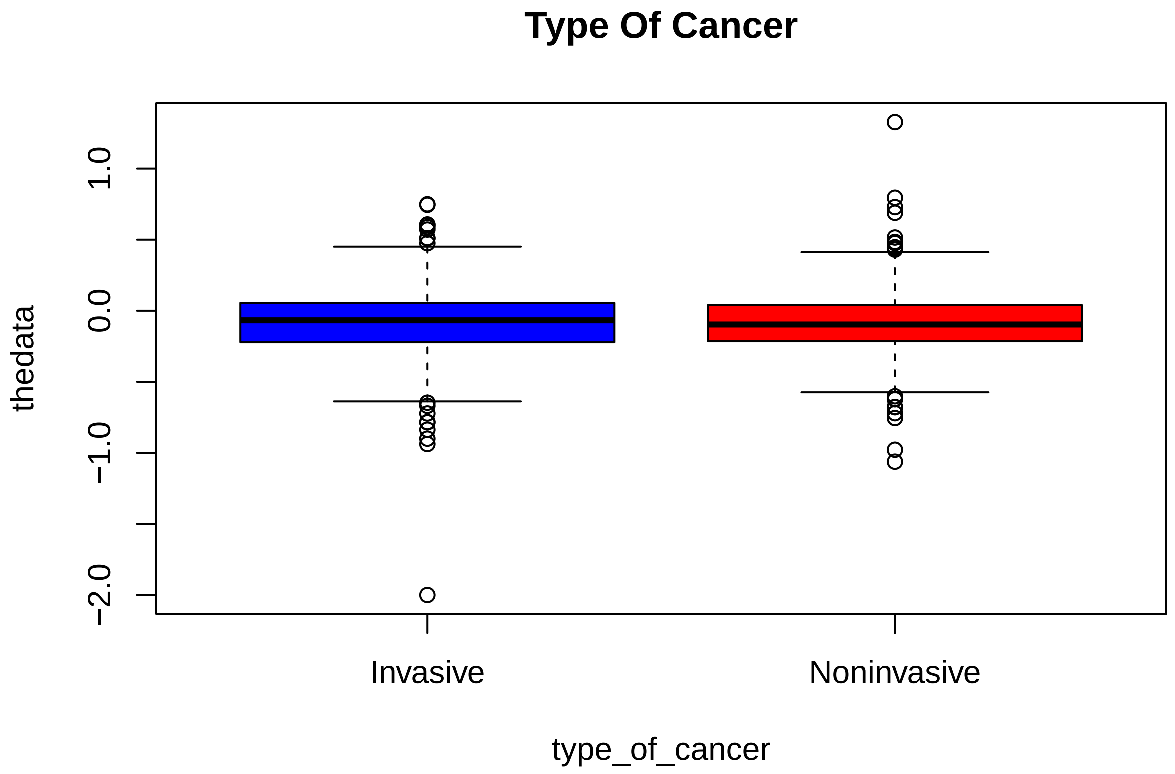
```
# Assign each label to the class that it belongs to (label1/2).
my_class_subset$type_of_cancer <- ifelse(my_class_subset$Class == 1,
                                         "Invasive", "Noninvasive")
my_class_subset$type_of_cancer <- factor(my_class_subset$type_of_cancer,
                                         label=c("Invasive","Noninvasive"))

# use a contingency table to verify the counts
table(my_class_subset$type_of_cancer)
```

```
##
##    Invasive Noninvasive
##          34          44
```

```
thedata <- as.matrix(my_class_subset[,c(1:10)])
manova_fit <- manova(thedata ~ type_of_cancer, data = my_class_subset)
summary(manova_fit,intercept=TRUE)
```

```
##                Df  Pillai approx F num Df den Df    Pr(>F)
## (Intercept)     1 0.65729  12.8500     10     67 3.646e-12 ***
## type_of_cancer  1 0.15178   1.1989     10     67    0.3078
## Residuals      76
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Give each of the two groups a boxplot.
boxplot(thedata ~ type_of_cancer, data = my_class_subset,
        main = "Type Of Cancer", col = c("blue", "red"))
```



```
names(my_class_subset)
```

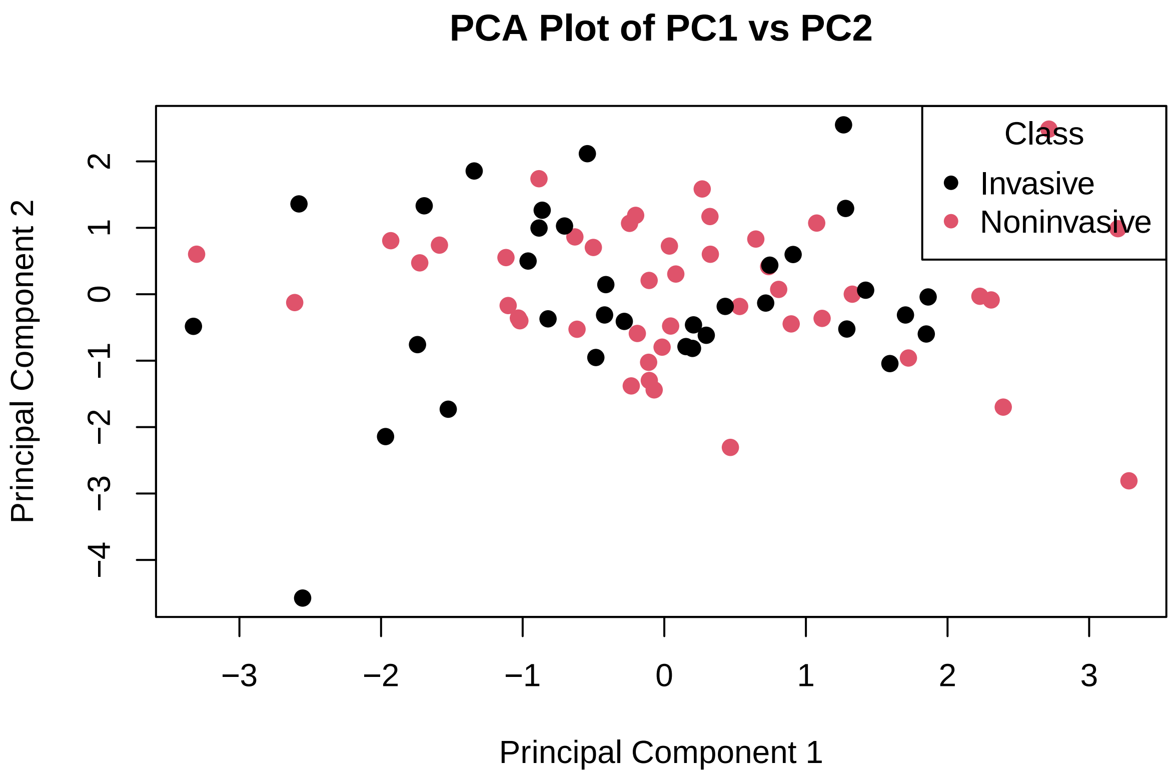
```
##  [1] "NM_006517"      "Contig39153_RC" "NM_018002"      "Contig53098_RC"
##  [5] "Contig57825_RC" "Contig47810_RC" "NM_002125"      "X66945"
##  [9] "NM_015364"      "NM_014873"      "Class"          "type_of_cancer"
```

THE EXPLANATION FOR QUESTION 3-A:

Multivariate analysis of variance (MANOVA) compares multivariate sample means between groups to see if there are any appreciable differences in the mean vectors. The analysis included 78 samples with 34 labeled as invasive and 44 as noninvasive.The Pillai's trace statistic for the cancer type effect is 0.15178, with an associated F-value of approximately 1.1989 and a p-value of 0.3078.The p-value is greater than 0.05, suggesting that there is no statistically significant difference in the expression of the 10 genes between invasive and noninvasive cancer types in this dataset. For the box plot, the blue box represents invasive cancer types, and the red box represents noninvasive types.The median gene expression level in both groups is close to zero, which may be the result of data normalization.The height of the boxes shows that the interquartile range (ICR) of gene expression levels in invasive cancers is marginally tighter than that of noninvasive cancers. The MANOVA analysis found no significant differences in the expression of the 10 genes between invasive and noninvasive cancer types. The boxplot shows that the median expression levels are similar for both cancer types, although noninvasive samples display a bit more variability. This suggests that the gene expression profiles of these 10 genes do not distinctly separate the two cancer types in this dataset.

```
#3-(b) Use the first and second principal component to illustrate,
#if there is a difference between invasive and
#noninvasive cancer. Add appropriate tables and figures to your report.
```

```r
# Assuming your data is in a dataframe `data`

# Perform PCA
pca_result <- prcomp(my_class_subset[, 1:10], center = TRUE, scale = TRUE)

# Extract the first and second principal components
pc1 <- pca_result$x[, 1]
pc2 <- pca_result$x[, 2]

# Create a scatter plot using the first and second principal components
plot(pc1, pc2, col = my_class_subset$type_of_cancer, pch = 16,
     main = "PCA Plot of PC1 vs PC2",
     xlab = "Principal Component 1", ylab = "Principal Component 2", cex = 1.2)

# Add a legend
legend("topright", legend = levels(my_class_subset$type_of_cancer),
       col = 1:2, pch = 16, title = "Class")
```



```r
names(my_class_subset)
```

```
##  [1] "NM_006517"      "Contig39153_RC" "NM_018002"      "Contig53098_RC"
##  [5] "Contig57825_RC" "Contig47810_RC" "NM_002125"      "X66945"
##  [9] "NM_015364"      "NM_014873"      "Class"          "type_of_cancer"
```

THE EXPLANATION FOR QUESTION 3-B:

The distribution of dots representing invasive and noninvasive cancer types on the PCA scatter plot does not demonstrate distinct clustering, suggesting that the first and second principal components do not offer a clear differentiation between the two cancer categories.

```
#4-(a) Apply LDA to your random subset of 10 genes and the class variable
#(invasive (label 1) and noninvasive
#(label 2) cancer). Calculate a confusion matrix, sensitivity, specificity and
#misclassification error. Add appropriate tables and figures to your report.

library(MASS)
lda1 <- lda(type_of_cancer ~ . - Class, data = my_class_subset)
lda1
```

```
## Call:
## lda(type_of_cancer ~ . - Class, data = my_class_subset)
##
## Prior probabilities of groups:
##    Invasive Noninvasive
##   0.4358974   0.5641026
##
## Group means:
##                NM_006517 Contig39153_RC    NM_018002 Contig53098_RC
## Invasive    -0.015235294    -0.10255882 -0.009235294    -0.03779412
## Noninvasive  0.006704545    -0.09977273 -0.087272727    -0.04531818
##             Contig57825_RC Contig47810_RC   NM_002125      X66945   NM_015364
## Invasive        -0.3638824   -0.043176471 -0.09026471 -0.09188235 -0.03823529
## Noninvasive     -0.2669318   -0.005295455 -0.09297727 -0.05322727 -0.10763636
##               NM_014873
## Invasive    -0.02623529
## Noninvasive -0.09661364
##
## Coefficients of linear discriminants:
##                       LD1
## NM_006517       0.8497299
## Contig39153_RC -0.1748986
## NM_018002      -4.7770301
## Contig53098_RC -0.1632517
## Contig57825_RC  1.6234566
## Contig47810_RC  1.3803482
## NM_002125       0.6745285
## X66945          0.6030916
## NM_015364      -0.6981193
## NM_014873      -2.4192150
```

```
#using training data, determine each observation's class label.
lda_predict <- predict(lda1)$class

#the results' confusion matrix
conf_matrix <- table(lda_predict,my_class_subset$type_of_cancer)
conf_matrix
```

```
##
## lda_predict   Invasive Noninvasive
##   Invasive          16          10
##   Noninvasive       18          34
```

```
#the entire set of observations used for training
total <- sum(conf_matrix) ; cat('Total observations: ', total, '\n')
```

```
## Total observations:  78
```

```
#total of the confusion matrix's rows and columns
row1 <- sum(conf_matrix[1, ])
cat('sum of the first row: ', row1, '\n')
```

```
## sum of the first row:  26
```

```
row2 <- sum(conf_matrix[2, ])
cat('sum of the second row: ',row2, '\n')
```

```
## sum of the second row:  52
```

```
column1 <- sum(conf_matrix[,1])
cat('sum of the first column: ',column1, '\n')
```

```
## sum of the first column:  34
```

```
column2 <- sum(conf_matrix[,2])
cat('sum of the second column: ',column2, '\n')
```

```
## sum of the second column:  44
```

```
#total number of classification errors seen
error <- conf_matrix[2,1]+ conf_matrix[1,2]
cat('Misclassified observations: ', error, '\n')
```

```
## Misclassified observations:  28
```

```
# probability of misclassification
prob_of_misclass = error/total
cat('probability of misclassification: ', prob_of_misclass, '\n')
```

```
## probability of misclassification:  0.3589744
```

```
# Sensitivity
sensitivity <- conf_matrix[1,1]/ row1
cat('Sensitivity is: ' , sensitivity, '\n')
```

```
## Sensitivity is:  0.6153846
```

```
# specificity
specificity <- conf_matrix[2,2]/ row2
cat('Specificity is:',specificity)
```

```
## Specificity is: 0.6538462
```

THE EXPLANATION FOR QUESTION 4-A:

We started by looking at the prior probabilities of the groups in order to evaluate the Linear Discriminant Analysis (LDA) model. This showed us that the dataset contains roughly 43.6% “Invasive” and 56.4% “Non-invasive” cancer types. Group means provided information about inter-class differences by clarifying the average values of predictors within each class. The construction of the linear combination of predictors for classification was found to depend critically on the coefficients of linear discriminants (LD1). With 78 observations taken into consideration, the confusion matrix allowed for a thorough evaluation that distinguished between true positives, false positives, true negatives, and false negatives. A concrete measure for assessing model performance was provided by the 28 misclassified observations, which were further quantified by a probability of misclassification of 35.9%.

```
#4-(b)Apply Quadratic discriminant analysis (QDA) to your random subset of
#10 genes and the class variable (invasive (label 1) and noninvasive
#(label 2) cancer). Calculate a confusion matrix, sensitivity, specificity
#and misclassification error. Add appropriate tables and figures to your report.
library(MASS)
qda <- qda(type_of_cancer ~ . - Class, data = my_class_subset)
qda
```

```
## Call:
## qda(type_of_cancer ~ . - Class, data = my_class_subset)
##
## Prior probabilities of groups:
##    Invasive Noninvasive
##   0.4358974   0.5641026
##
## Group means:
##                NM_006517 Contig39153_RC    NM_018002 Contig53098_RC
## Invasive    -0.015235294    -0.10255882 -0.009235294    -0.03779412
## Noninvasive  0.006704545    -0.09977273 -0.087272727    -0.04531818
##             Contig57825_RC Contig47810_RC   NM_002125      X66945   NM_015364
## Invasive        -0.3638824   -0.043176471 -0.09026471 -0.09188235 -0.03823529
## Noninvasive     -0.2669318   -0.005295455 -0.09297727 -0.05322727 -0.10763636
##               NM_014873
## Invasive    -0.02623529
## Noninvasive -0.09661364
```

```
#using training data, determine each observation's class label.
qda_predict <- predict(qda)$class

#the results' confusion matrix
conf_matrix <- table(qda_predict,my_class_subset$type_of_cancer)
conf_matrix
```

```
##
```

```
## qda_predict   Invasive Noninvasive
##   Invasive          25           4
##   Noninvasive        9          40
```

```
#the entire set of observations used for training
total <- sum(conf_matrix) ; cat('Total observations: ', total, '\n')
```

```
## Total observations:  78
```

```
#total of the confusion matrix's rows and columns
row1 <- sum(conf_matrix[1, ])
cat('sum of the first row: ', row1, '\n')
```

```
## sum of the first row:  29
```

```
row2 <- sum(conf_matrix[2, ])
cat('sum of the second row: ',row2, '\n')
```

```
## sum of the second row:  49
```

```
column1 <- sum(conf_matrix[,1])
cat('sum of the first column: ',column1, '\n')
```

```
## sum of the first column:  34
```

```
column2 <- sum(conf_matrix[,2])
cat('sum of the second column: ',column2, '\n')
```

```
## sum of the second column:  44
```

```
#total number of classification errors seen
error <- conf_matrix[2,1]+ conf_matrix[1,2]
cat('Misclassified observations: ', error, '\n')
```

```
## Misclassified observations:  13
```

```
# probability of misclassification
prob_of_misclass = error/total
cat('probability of misclassification: ', prob_of_misclass, '\n')
```

```
## probability of misclassification:  0.1666667
```

```
# Sensitivity
sensitivity <- conf_matrix[1,1]/ row1
cat('Sensitivity is: ' , sensitivity, '\n')
```

```
## Sensitivity is:  0.862069
```

```
# specificity
specificity <- conf_matrix[2,2]/ row2
cat('Specificity is:',specificity)
```

```
## Specificity is: 0.8163265
```

THE EXPLANATION FOR QUESTION 4-B:

In the QDA model evaluation, we applied the QDA algorithm to our random subset of 10 genes and the class variable representing invasive and noninvasive cancer types. The model outputs indicate the prior probabilities of groups, with invasive cancer types constituting approximately 43.6% and noninvasive types around 56.4%. Group means provide insights into the average values of predictors within each class, aiding in understanding inter-class variations. The confusion matrix displays classification results, revealing 25 instances correctly classified as invasive and 40 instances correctly classified as noninvasive, with 13 misclassified observations. This translates to a misclassification error of approximately 16.7%. Sensitivity, measuring the proportion of true positives correctly identified, is approximately 86.2%, indicating strong performance in identifying invasive cancer types. Specificity, measuring the model's ability to correctly identify true negatives, is around 81.6%, suggesting a good ability to discern noninvasive cancer types.

```
#4-(c) Discuss the difference between LDA and QDA using the results on your
#random subset of 10 genes andthe class variable
#(invasive (label 1) and noninvasive (label 2) cancer).
```

THE EXPLANATION FOR QUESTION 4-C:

When comparing the outputs of Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) on the random subset of 10 genes and the class variable (invasive and noninvasive cancer), it's notable that both methods yield similar confusion matrices, with some variations in classification errors and sensitivities. Both LDA and QDA have the same prior probabilities of groups and group means, indicating consistency in their classification approaches.

```
#5. (25 marks). Use the median of the first principal component of your random
#subset of 10 genes to predict the class variable (invasive (label 1) and
#noninvasive (label 2) cancer).
#Use Fisher's Exact test and sensitivity,
#specificity and Youden index. Add appropriate tables and figures to your report.
# The first principal component P.C.1
pc1 <- pca$scores[, 1]

# The median of the first principal component
pc1_median <- median(pc1) ; cat('Median of pc1 is: ', pc1_median)
```

```
## Median of pc1 is:  -0.04378912
```

```
decision <- ifelse(pc1 > pc1_median, "Invasive", "Noninvasive")
decision <- factor(decision, label=c("Invasive","Noninvasive"))

# build a confusion matrix
conf_matrix <- table(decision,my_class_subset$type_of_cancer) ; conf_matrix
```

```
##
## decision      Invasive Noninvasive
##   Invasive          16          23
##   Noninvasive       18          21
```

```r
# Fisher's Exact test
fisher.test(conf_matrix)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  conf_matrix
## p-value = 0.8196
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.3010991 2.1825224
## sample estimates:
## odds ratio
##  0.8137785
```

```r
# Sensitivity
sensitivity <- conf_matrix[1,1]/ row1 ;  sensitivity
```

```
## [1] 0.5517241
```

```r
# specificity
specificity <- conf_matrix[2,2]/ row2 ; specificity
```

```
## [1] 0.4285714
```

```r
# Youden index
youden_index <- sensitivity+specificity-1;youden_index
```

```
## [1] -0.01970443
```

THE EXPLANATION FOR QUESTION 5:

The median of the first principal component from our random subset of 10 genes, we predicted the class variable (invasive and noninvasive cancer). The median of the first principal component (P.C.1) was found to be -0.04378912. We then classified observations as "Invasive" or "Noninvasive" based on whether their P.C.1 values were above or below the median, respectively. The resulting confusion matrix indicates that out of 78 observations, 16 were classified as Invasive and 23 as Noninvasive, while 18 were misclassified as Noninvasive and 21 as Invasive. We conducted Fisher's Exact test to assess the association between the predicted classes and the actual cancer types. The test yielded a p-value of 0.8196, suggesting no significant association between the predicted and actual classes. Furthermore, we calculated the sensitivity and specificity of the classification. The sensitivity, representing the proportion of correctly identified invasive cases, was 55.2%, while the specificity, indicating the proportion of correctly identified noninvasive cases, was 42.9%. Lastly, the Youden index, a measure of the classifier's overall performance, was found to be -0.0197, indicating limited discriminative ability between the predicted and actual classes. These findings suggest that using the median of the first principal component alone may not be sufficient for accurate prediction of cancer types, as evidenced by the relatively low sensitivity and specificity and the nonsignificant association revealed by Fisher's Exact test.