Statistical Analysis of Cancer Data

Jessica Smith

Pre-processing:

```
# Read in the data
data.csv <- read.csv("data.csv")
labels.csv <- read.csv("labels.csv", row.names = 1)
```

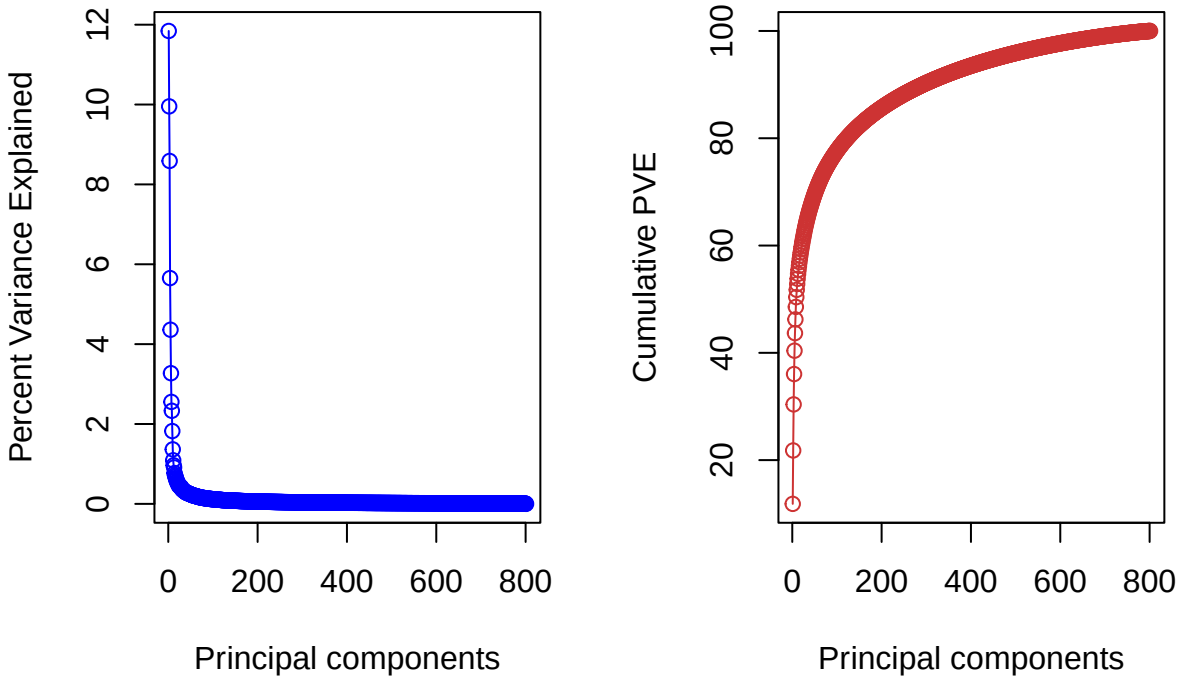
```
# filter out genes from data.csv whose expressions are zero
# for at least 300 subjects
library(dplyr)
library(tibble)
zero_count <- colSums(data.csv == 0)
z_count <- data.frame(zero_count)
keepers = z_count %>%
    rownames_to_column("gene") %>%
    filter(zero_count < 300) %>%
    column_to_rownames("gene")
data1.csv <- data.csv[, which((names(data.csv) %in% rownames(keepers)) ==
    TRUE)]
data1.csv <- column_to_rownames(data1.csv, "X")

# Standardize the gene expressions for each gene in
# 'data1.csv' so that they have sample standard deviation 1
# and sample mean 0.
data2.csv = scale(data1.csv)
```

Identify patterns and low-dimensional structures

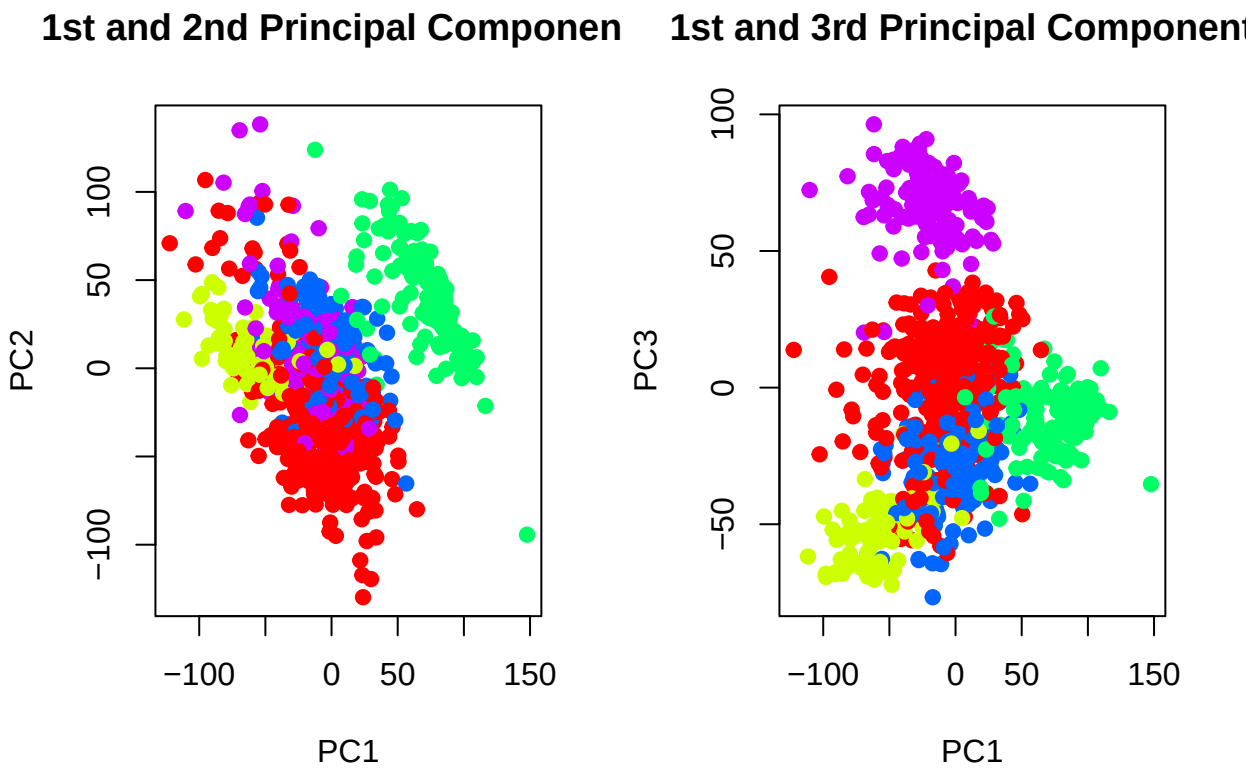
Objectives: Apply PCA, determine the number of principal components, provide visualizations of low-dimensional structures.

```
# Apply PCA
pr.out = prcomp(data1.csv, scale = T)
pve = 100 * pr.out$sdev^2/sum(pr.out$sdev^2)

# determine the number of principal components
par(mfrow = c(1, 2))
plot(pve, type = "o", ylab = "Percent Variance Explained", xlab = "Principal components",
```

```
    col = "blue")
plot(cumsum(pve), type = "o", ylab = "Cumulative PVE", xlab = "Principal components",
    col = "brown3")
```



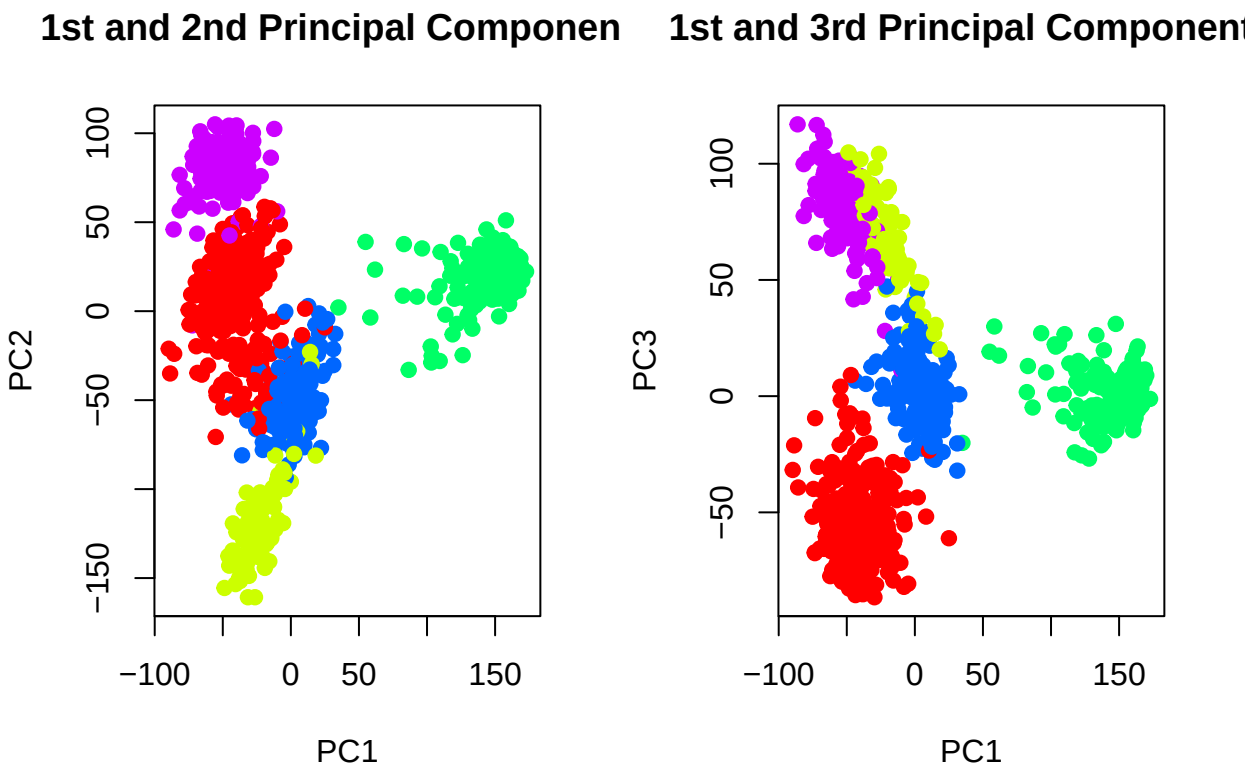

```
# provide visualizations of low-dimensional structures
Cols = function(vec) {
    cols = rainbow(length(unique(vec)))
    return(cols[as.numeric(as.factor(vec))])
}
par(mfrow = c(1, 2))
# Plot the first principle component against the second
plot(pr.out$x[, 1:2], col = Cols(labels.csv[, 1]), pch = 19,
    xlab = "PC1", ylab = "PC2", main = "1st and 2nd Principal Components")
# plot scores for 1st principal component against those for
# the third
plot(pr.out$x[, c(1, 3)], col = Cols(labels.csv[, 1]), pch = 19,
    xlab = "PC1", ylab = "PC3", main = "1st and 3rd Principal Components")
```



Findings: The first 9 principle components account for more than 50% of the total variance. Overall, the cell lines corresponding to a single cancer type tend to have similar values for the first few principal components. This indicates that cell lines from the same cancer type tend to have similar gene expressions.

Objectives: Apply SPCA, provide visualizations of low-dimensional structures.

```
# Apply SPCA
library(sparsepca)
resB = spca(data1.csv, k = 3, alpha = 1e-04, beta = 1e-04, center = TRUE,
    scale = FALSE, max_iter = 200, tol = 1e-05, verbose = FALSE)

# provide visualizations of low-dimensional structures
par(mfrow = c(1, 2))
plot(resB$scores[, 1:2], col = Cols(labels.csv[, 1]), pch = 19,
    xlab = "PC1", ylab = "PC2", main = "1st and 2nd Principal Components")
plot(resB$scores[, c(1, 3)], col = Cols(labels.csv[, 1]), pch = 19,
    xlab = "PC1", ylab = "PC3", main = "1st and 3rd Principal Components")
```



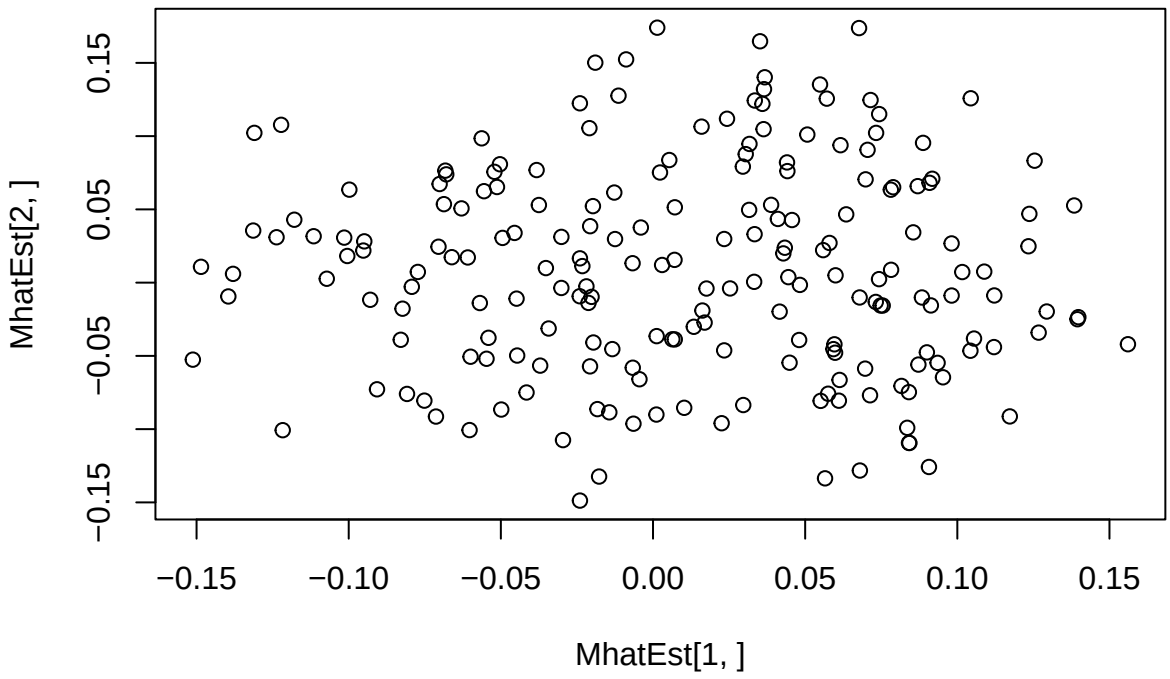
Findings: Applying a SPCA shows that cancer types tend to have similar gene expressions for the first few principle components. The results are similar to those from the PCA.

Objectives: Estimate the latent space.

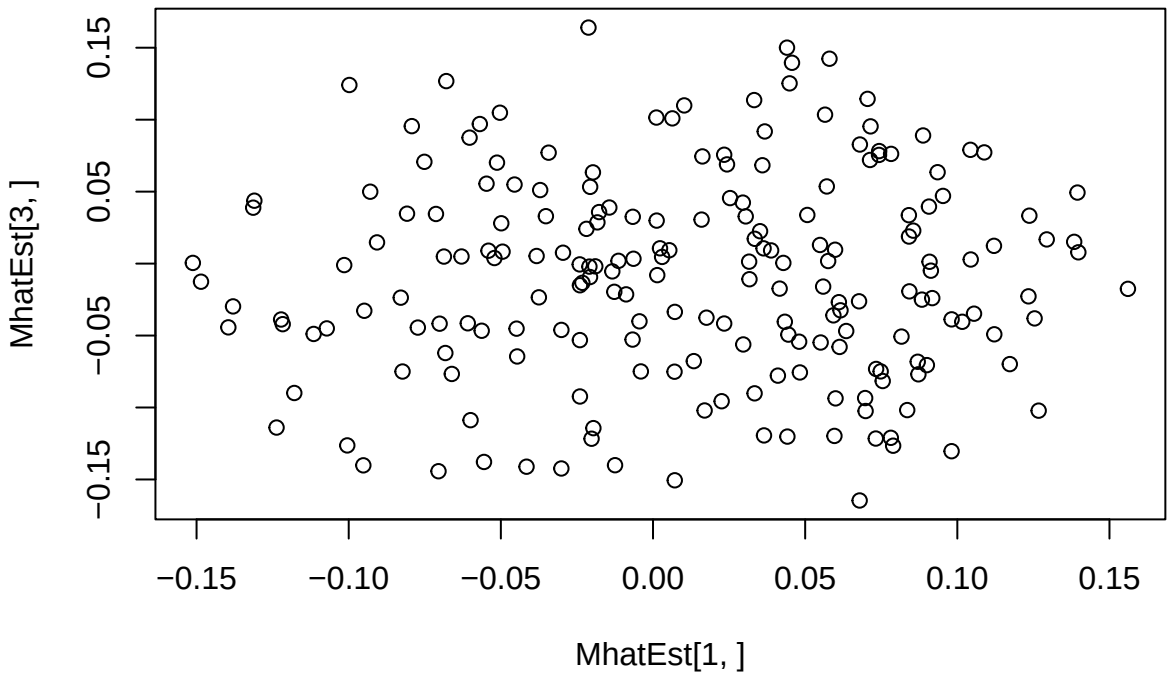
```
# Apply the 'methodology of estimating the latent space'
EstVarsFunc = function(Y = NULL, v = NULL, Dist = NULL) {

    if (Dist != "Normal" & Dist != "Poisson") {
        if (is.null(v)) {
            stop("---Please specify the second (fixed) parameter of the distribution.")
        }
        if (v == 0) {
            stop("---The second (fixed) parameter of the distribution cannot be zero.")
        }
    }

    if (Dist == "Normal") {
        Y0 = matrix(1, nrow(Y), ncol(Y))
    }

    if (Dist == "Poisson") {
        Y0 = Y
    }
```

```
    if (Dist == "Binomial") {
        Y0 = (v * Y - Y^2)/(v - 1)
    }

    if (Dist == "NegativeBinomial") {
        Y0 = (v * Y + Y^2)/(v + 1)
    }

    if (Dist == "Gamma") {
        Y0 = Y^2/(v + 1)
    }

    return(Y0)
}  # end

GetMainMatrixFunc = function(Y = NULL, v = NULL, Dist = NULL) {

    k = dim(Y)[1]
    Y0 = EstVarsFunc(Y, v, Dist)  # get variances

    # compute Dhat and Rhat matrix
    Dhat = diag(colMeans(Y0))
    Rhat = k^(-1) * t(Y) %*% Y - Dhat

    # return R matrix
    return(list(Rhat = Rhat, Dhat = Dhat))
}


EstSpanMFunc = function(Y = NULL, v = NULL, Dist = NULL, r = NULL,
    rEstMeth = "Ratio") {
    k = dim(Y)[1]
    n = dim(Y)[2]

    RhatDhat = GetMainMatrixFunc(Y, v, Dist)
    Rhat = RhatDhat$Rhat

    if (is.null(r)) {
        rEst = EstRankFunc(Y, v, Dist, rEstMeth)
    } else {
        rEst = r
    }

    Rsvd = svd(Rhat)
    RsvdV = Rsvd$v
    Mh = t(RsvdV[, 1:rEst])
    # return
```

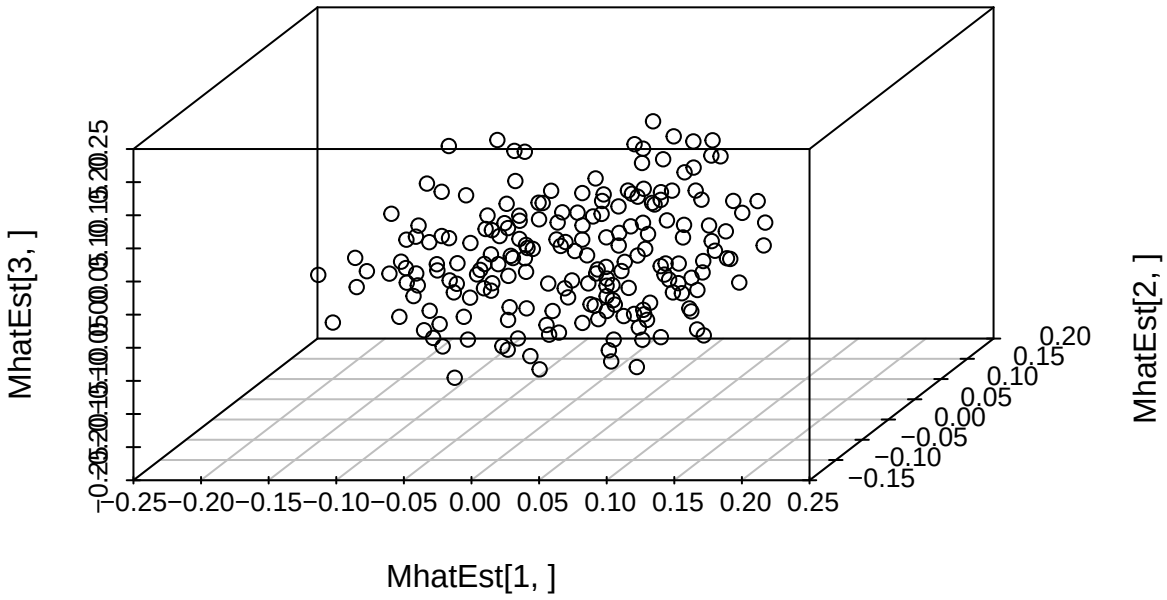
```
    return(list(Mhat = Mh, rEst = rEst))

}  # end of function

# reduce the size of the data set
set.seed(123)
data3.csv = scale(data1.csv[sample(nrow(data1.csv), 200)], center = TRUE,
    scale = TRUE)


# estimate a latent space of dimension 5 by specifying
# 'r=5'
EstRes = EstSpanMFunc(Y = data3.csv, v = 1, Dist = "Normal",
    r = 5, rEstMeth = "Ratio")

MhatEst = EstRes$Mhat
plot(MhatEst[1, ], MhatEst[2, ])
```



```r
plot(MhatEst[1, ], MhatEst[3, ])
```




```r
library("scatterplot3d")
scatterplot3d(MhatEst[1, ], MhatEst[2, ], MhatEst[3, ], angle = 55)
```



Multiple hypothesis testing

Objectives: Test if the mean gene expressions are different between two cancer types.

```
# Pick 2 cancer types that have the largest number of
# observations
table(labels.csv)  #BRCA amd KIRC have the largest number of observations
```

```
## Class
## BRCA COAD KIRC LUAD PRAD
##  300   78  146  141  136
```

```
BRCA = rownames(labels.csv)[labels.csv$Class == "BRCA"]
KIRC = rownames(labels.csv)[labels.csv$Class == "KIRC"]

# Pick the same set of 1000 genes from 'data2.csv' for the
# 2 cancer types.
set.seed(123)
data4.csv = data2.csv[, sample(colnames(data2.csv), 1000)]
BRCA.csv = data4.csv[BRCA, ]
KIRC.csv = data4.csv[KIRC, ]
```

```r
# For each gene, test if the mean gene expressions are
# different between the 2 cancer types
m = 1000
pVec = double(m)
for (i in 1:m) {
    # two-sample t-test with pooled variance; two-sided
    # p-value
    pVec[i] = t.test(BRCA.csv[, i], KIRC.csv[, i], var.equal = TRUE,
        alternative = "two.sided")$p.value
}

# Apply the Benjamini-Hochberg procedure at FDR=0.05
BH <- function(Pvals, FDRlevel) {
    if (any(is.na(Pvals)) | any(is.nan(Pvals))) {
        cat("^^NA or NAN in pvalues... ", "\n")
        print(Pvals[is.na(Pvals)])
    }

    if (any(is.na(FDRlevel)) | any(is.nan(FDRlevel)))
        cat("^^NA or NAN FDRlevel... ", "\n")

    if (is.vector(Pvals))
        lgh3 <- length(Pvals)
    if (is.matrix(Pvals) | is.data.frame(Pvals))
        lgh3 <- dim(Pvals)[1]
    PvalAndIdx <- cbind(Pvals, seq(1:lgh3))
    PvalAndIdxOrd <- PvalAndIdx[order(PvalAndIdx[, 1]), ]

    # cat('^^Dims of PvalAndIdxOrd
    # is:',as.vector(dim(PvalAndIdxOrd)),'\n')

    BHstepups <- seq(1:lgh3) * (FDRlevel/lgh3)
    cmp3 <- PvalAndIdxOrd[, 1] <= BHstepups
    scmp3 <- sum(cmp3)

    # collect rejections if any
    if (scmp3 == 0) {
        print("No rejections made by BH procedure")  # when there are no rejections
        rejAndTresh <- list(matrix(numeric(0), ncol = 2, nrow = 0),
            0)
    } else {
        r <- max(which(cmp3))
        # cat('^^^ Minimax index in BH is:',r,'\n')
        # cat('^^^ Minimax threshold in BH
        # is:',BHstepups[r],'\n')
        if (r == 1) {
            # when r =1, a row is chosen, so should change
```

```
            # it into a matrix of two columns
            BHrej = as.matrix(t(PvalAndIdxOrd[1:r, ]))
            # print(BHrej)
        } else {
            BHrej <- PvalAndIdxOrd[1:r, ]
        }
        rejAndTresh <- list(BHrej, BHstepups[r])
    }
    return(rejAndTresh)
}

# any(is.nan(pVec))

Res = BH(pVec, 0.05)
# Res is a list that has 2 components; Res[[1]] contains
# p-values (in column 1) and their indices (in column 2),
# corresponding to rejected null hypotheses; Res[[2]]
# contains the rejection threshold Report the list of genes
# that are differentially expressed between the 2 cancer
# types.
dim(Res[[1]])
```

```
## [1] 845   2
```

Findings: The cancer types BRCA and KIRC have the largest number of observations. There are 830 genes for which the null hypothesis was rejected, indicating the mean gene expressions are different between the 2 cancer types.

Conclusion: The Principal Components Analysis plots exhibit clustering, indicating that the genes for each cancer type have similar expressions. Multiple hypothesis testing showed that 830 out of 1000 genes expressions were significantly different between the two most frequently observed cancer types.