Capstone project

Chi Nguyen

22/12/2023

Title

Prostate Cancer

Abstract

Prostate cancer (PC) is the second most frequently diagnosed type of cancer. In this project, we investigated the transcriptome differences between untreated prostate cancer and locally recurrent castration-resistant prostate cancer CRPC.

Introduction

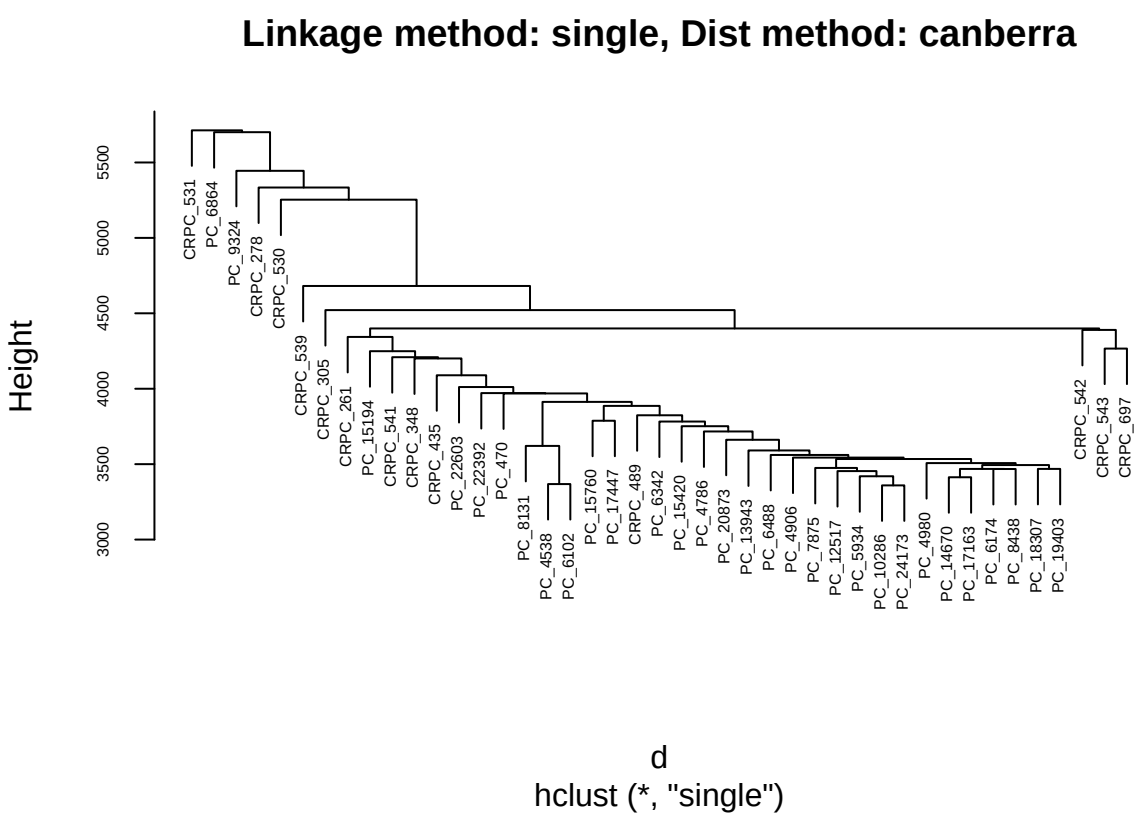
The two groups of prostate cancer samples were used to study the transcriptome differences. One group of the sample is untreated sample, while the other group of samples is radical prostatectomy and locally recurrent. The transcripts data were obtained from an RNA-seq experiment to analyse the expression, especially the expressed transcripts. Therefore, identifying these specific prostate cancer CRPC in terms of their functional association in prostate cancer progression might help to discover or explain the differences in gene regulation of the two groups.

Results

After looking at the article and the data set given, it is noted that filtering step is needed to perform to improve and avoid bias before doing further analysis. Normalization was done also in the project to confirm that this step is not needed as the quality of boxplot is not yet improved. Next, all the genes with zero counts of transcripts across all the samples were removed.

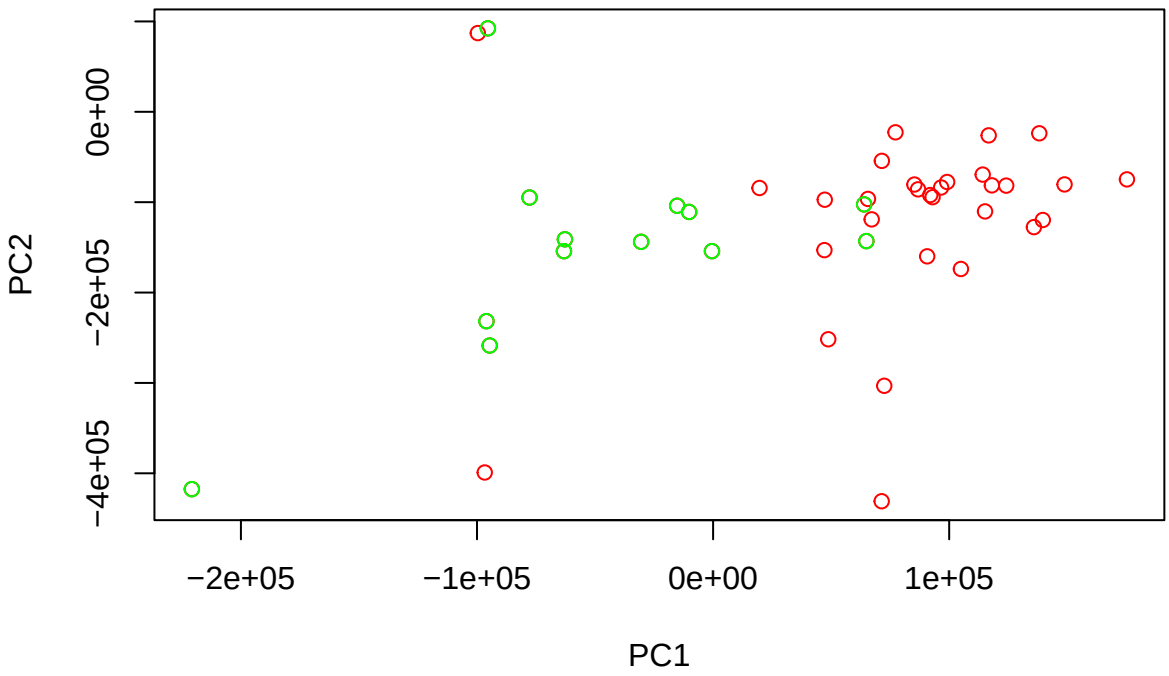
Clustering samples were performed, which resulted in almost homogeneous for most of the combination of linkage and distance function used for the clustering. Clustering using Canberra distance method and single linkage method highlights the outliers from PC sample group.

```
title = sprintf('Linkage method: %s, Dist method: %s', 'single', 'canberra')
par(cex.axis=0.5)
plot(result, main = title, cex = 0.5)
```



PCA analysis was performed to visualize the variance per component. Different level of espression for the two group was clearly obsevered, in which green dots represented for CRPC and red dots represented for PC. The result confirmed the clustering map that we got from analysis 2 as they are grouped as the way I expected after reviewing different clustering maps.

```
plot(points, col = 'red')
points(points[1:13,], col = 'green')
```



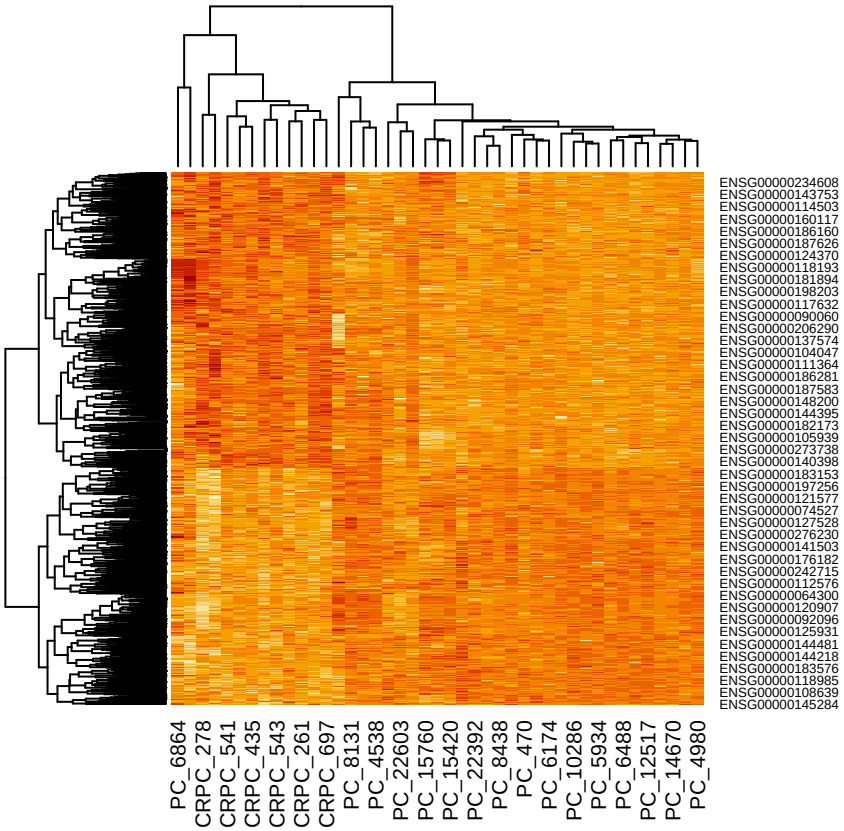
Differential espression analysis was carried out using DESqe. Student t'test was chosen for the statistical test. Genes satisfying both two criteria which are: p-value<0.0001 and effect size>1 are chosen to plot the volcano plot and be saved as the gene list variable. The chosen p-value is because of lagre amount of genes have really small p-value, thus using 0.0001 help to signify the differential espression. Volcano plot shows genes that are upregulated or downregulated, and signficantly expressed in different colors. By ordering the gene list with asccending p-value-adjusted, genes that are significantly expressed are recorded in the top of the list. The gene list below show the genes that are most differentially expressed between the two sample groups. There are in total 506 differentially expressed genes (with selected criteria).

```
head(ordered_gene_list, 5)
```

```
##                 statistic        dm      p.value  p.value.adj
## ENSG00000147647 -9.213596 -5.265276 1.534393e-11 2.530522e-07
## ENSG00000124205 -8.729041 -5.762183 6.767890e-11 2.872607e-07
## ENSG00000122877 -8.803296 -2.794893 5.381532e-11 2.872607e-07
## ENSG00000184588 -8.647583 -1.946615 8.709092e-11 2.872607e-07
## ENSG00000007908 -8.648253 -3.524544 8.691019e-11 2.872607e-07
```

Heatmap was plotted using the list of differentially expressed genes from analysis 4. However, the plot seems to be quiet homogeneous. On the other hand, sample PC6864 is present accordingly with those previous analysis.

```
heatmap(as.matrix(count_matrix_de_ttest), distfun = correlation_dist)
```



Next, KEGG pathway enrichment is performed using the list of differentially expressed genes identified in Analysis 4. The results revealed several enriched KEGG pathways, implying potential biological processes associated with the observed gene expression change.

```
head(enriched_pathways, 10)
```

```
##            p-value
## 05146 0.0001131809
## 04512 0.0001575726
## 00140 0.0001627656
## 04110 0.0002390672
## 05150 0.0002752227
## 04974 0.0003037818
## 04621 0.0055106550
## 04270 0.0060008124
## 00500 0.0067983039
## 00590 0.0156606348
```

Discussion

Examining the results from our analyses, it is notable that two samples, PC 6864 and PC 9324, standing out as outliers was consistent across both hierarchical clustering and PCA analyses. It can display certain molecular characteristics that are in line with castration-resistant prostate cancer (CRPC). From the differential expression analysis, those genes that are most differential expressed are downregulated due to negative log gold change. It indicates a significant decrease in expression levels in one group compared to the other one. Therefore, the enriched pathways analysis offer insights into potential genetic mechanisms contributing to treatment resistance. From my perspective, many of the enriched pathways are associated with stress,

infection, cell division regulation, hormone regulation and blood regulation.

Material and methods

The data are acquired from an RNA-seq experiment of clinical samples from prostate cancer patients during surgery. There are two groups of samples: untreated prostate cancer (PC) and castration-resistant prostate cancer (CRPC). There are 13 and 30 samples in each groups PC and CRPC respectively. The lowly expressed genes are defined as those having the count across all sample smaller than the 25th median. Different distance method were used, however according to the clustering results, Canberra was chosen to obtain reasonable results, ans single linkage method was used to emphasize the outlier from the data. For the differential expression analysis, Student t'test for row was applied to study. The threshold in which the selected genes have p-values < 0.0001 and effect size > 1. And, the threshold p-value chosen for enriched pathway is 0.05.

R scripts

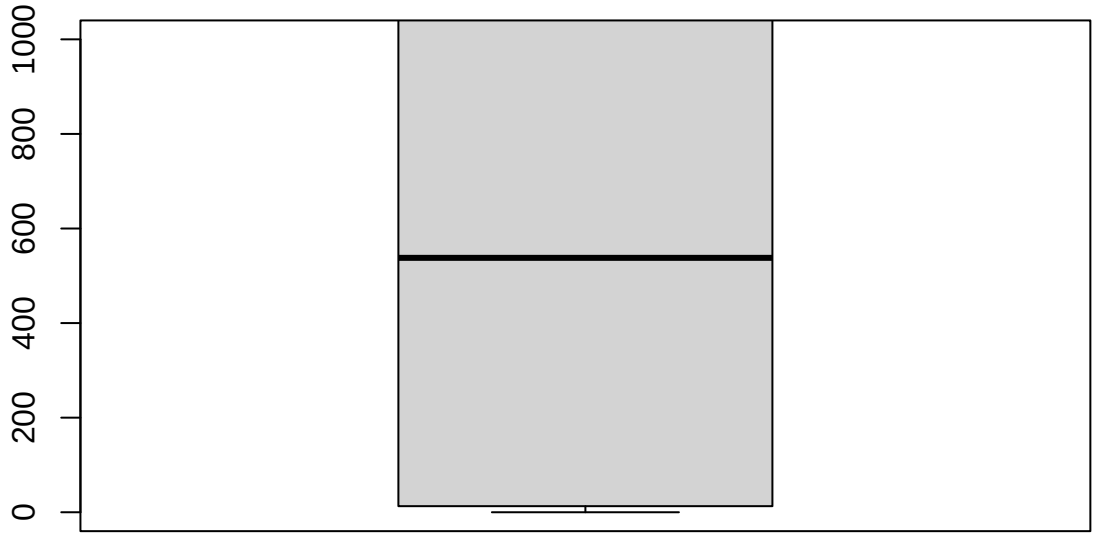
Analysis 1

```
count_matrix <- readRDS('/student_data/BBT.BI.202_2023/data/capstone_project/RNA_expressions.RDS')
filter = rowSums(count_matrix) == 0
with_zero <- count_matrix[filter,]
```

```
row_medians <-apply(count_matrix, MARGIN = 1, FUN = median)
max(row_medians)
```

```
## [1] 636190
```

```
boxplot(row_medians,
  ylim=c(0,1000))
```




```
threshold <- quantile(row_medians, probs = 0.25)
sum(row_medians<=threshold)
```

```
## [1] 5498
```

```
valid_rows = row_medians > threshold
count_matrix_filtered <- count_matrix[valid_rows,]
```

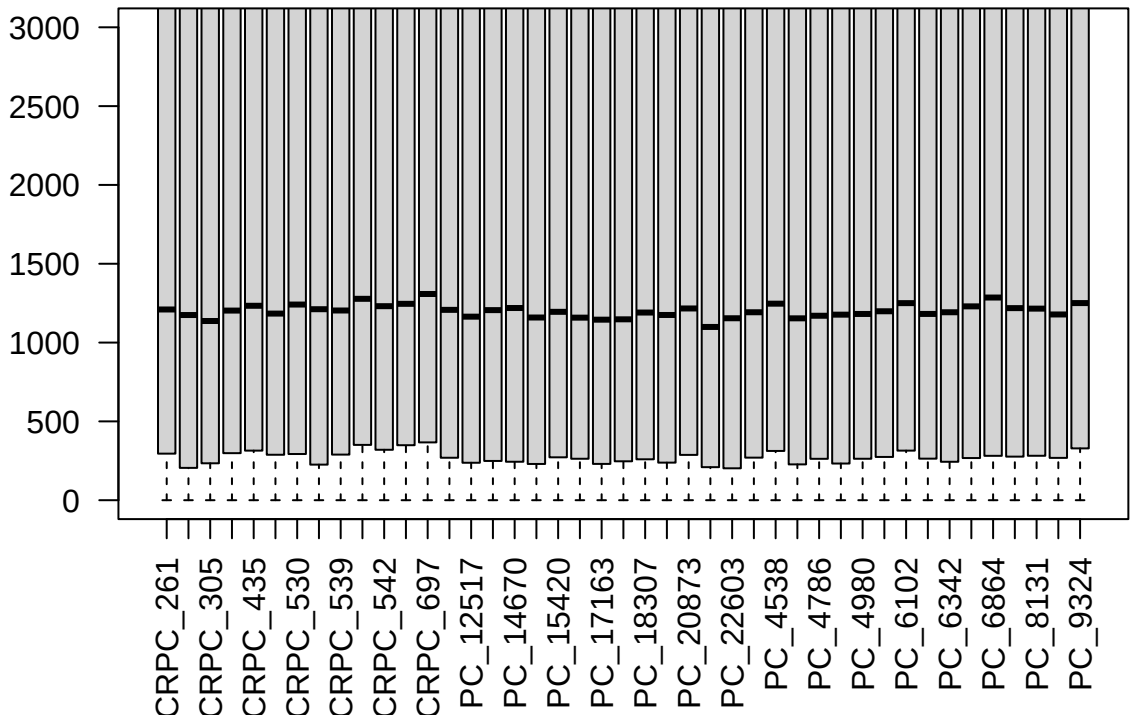
Now, the filtered count matrix should be normalized using median of ratios.

```
library(DESeq2)

## Preparing the sample_types that is going to assigned to colData argument in DESeqDataSetFromMatrix()
column_names = colnames(count_matrix_filtered)

sample_types = matrix(sub(pattern="_.*",
                          replacement="",
                          column_names)
                      )

rownames(sample_types) = column_names
colnames(sample_types) = c("Type")

## creating the data set expected by DESeq2
count_matrix_filtered[] <- lapply(count_matrix_filtered, as.integer)
dataset <- DESeqDataSetFromMatrix(countData=count_matrix_filtered,
                                  colData=sample_types,
                                  design=~1)
```

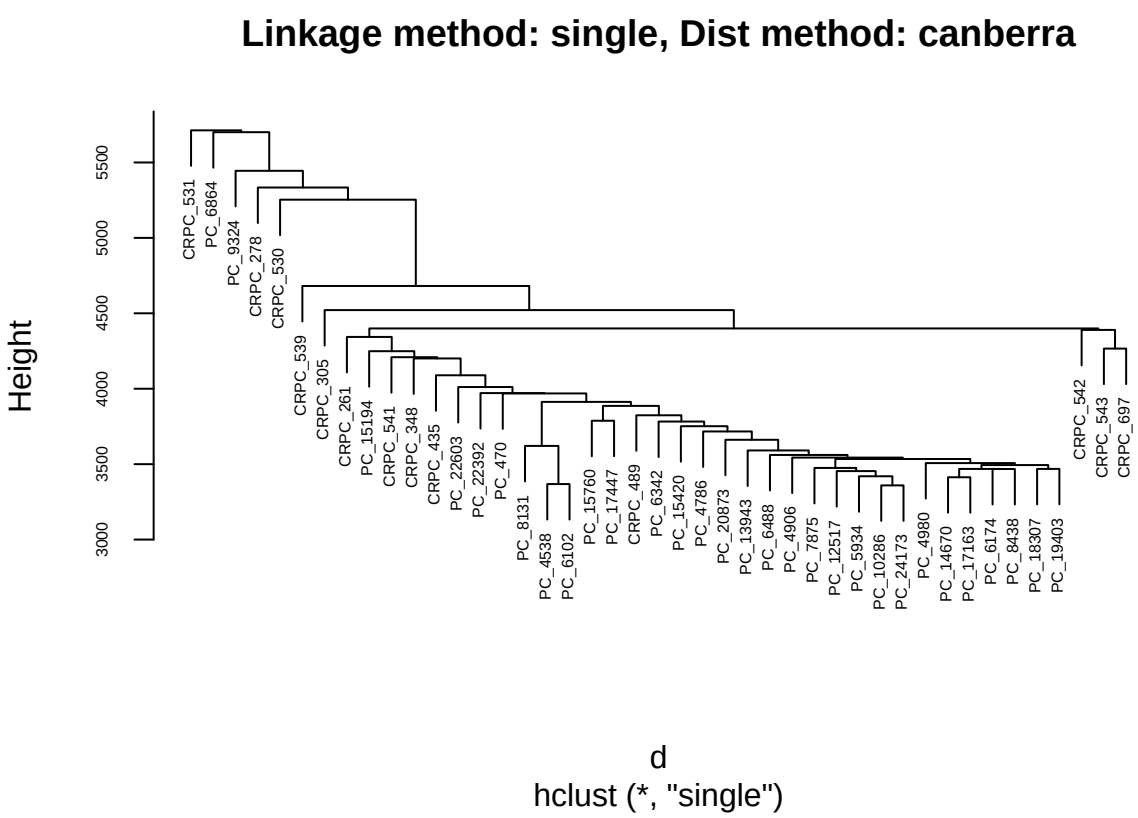
```
dataset <- estimateSizeFactors(dataset)

count_matrix_filtered_mor_normed <- counts(dataset,
                                           normalized=TRUE)
```

Boxplot for the data after filtering and normalization:

```
boxplot(count_matrix_filtered_mor_normed, las = 2, ylim=c(0,3000))
```



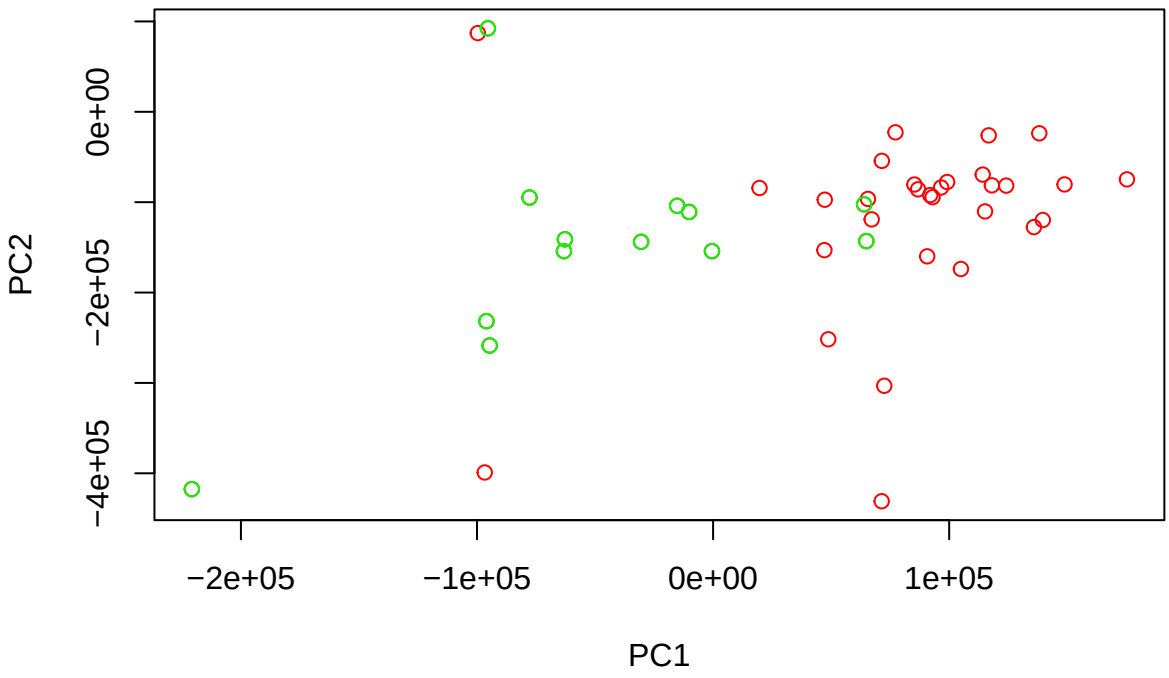

```
d = dist(t(count_matrix_filtered_mor_normed), method = 'canberra')
result = hclust(d, method = 'single')
title = sprintf('Linkage method: %s, Dist method: %s', 'single', 'canberra')
par(cex.axis=0.5)
plot(result, main = title, cex = 0.5)
```



Analysis 2

Analysis 3

```
pca_result = prcomp(t(count_matrix_filtered_mor_normed), center = TRUE, scale. = TRUE)
points = t(t(pca_result$rotation[,1:2])%*% as.matrix(count_matrix_filtered_mor_normed))
plot(points, col = 'red')
points(points[1:13,], col = 'green')
```



Analysis 4

```
## prepare the factor variable for fac argument in rowttests function and perform the statistical test
library(genefilter)
count_matrix_norm_log = log2(count_matrix_filtered+1)
sample = sub("_[0000-9999]*", "", colnames(count_matrix_norm_log))
matrix_fac = factor(sample)
ttest_results = rowttests(as.matrix(count_matrix_norm_log), matrix_fac)
sum(ttest_results$p.value<0.05)
```

```
## [1] 5224
```

Adjust p_value using Benjamini-Hochberg correction method

```
p.value.adj = p.adjust(ttest_results$p.value, 'BH')
ttest_results <- cbind(ttest_results, p.value.adj)
```

Another statistical test is done with a view to comparing the results.

```
my_wilcox = function(v, group1, group2) {

  test_result = wilcox.test(x=v[group1], y=v[group2], exact=FALSE)

  p.value = test_result$p.value
  names(p.value) = "p.value"

  return(c(test_result$statistic, p.value))
}
wilcox_results <- apply(count_matrix_norm_log, 1, my_wilcox, group1=1:13, group2=14:43)
```

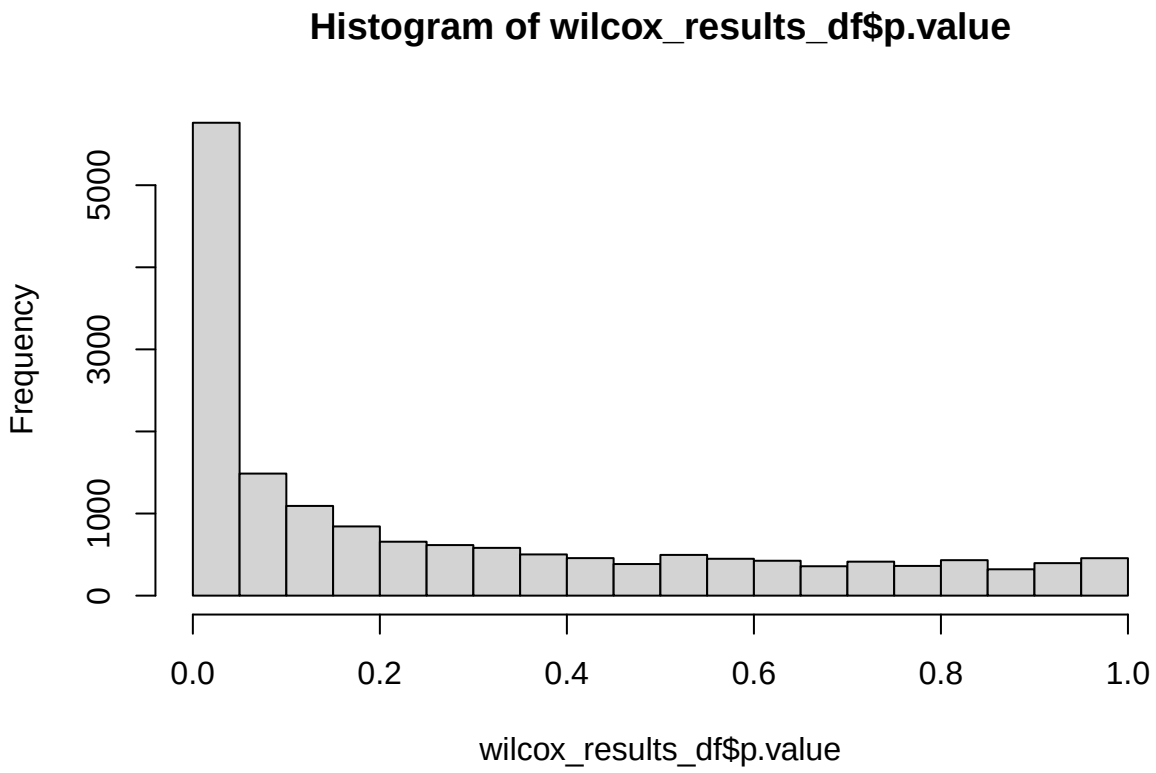
```r
wilcox_results_df = as.data.frame(t(wilcox_results))
# note the t()
count(wilcox_results_df$p.value<0.1)
```

```
## [1] 7248
```

```r
which.min(wilcox_results_df$p.value)
```

```
## [1] 6768
```

```r
hist(wilcox_results_df$p.value)
```



Adjust the p_value of wilcox test using BH

```r
p.value.adj.BH = p.adjust(wilcox_results_df$p.value, 'BH')
wilcox_results_df['p.value.adj.BH'] = p.value.adj.BH
```

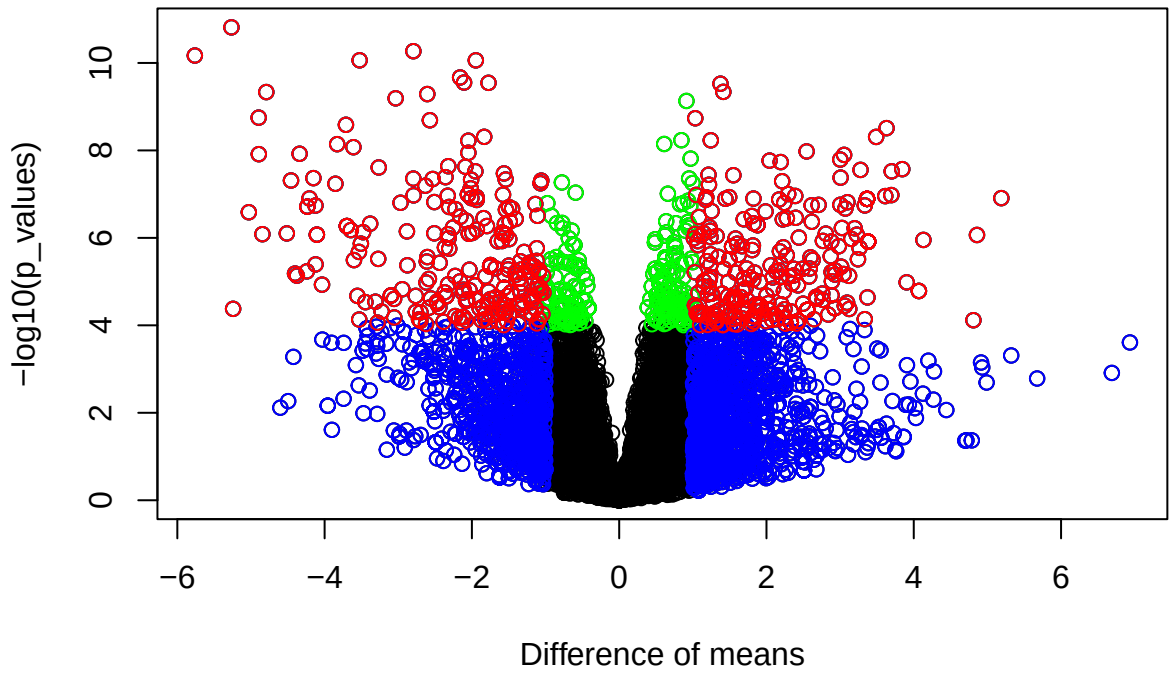
The effect size for each gene and p-values are used to plot the volcano plot: We have 2 criterion to select genes: p-values < 0.0001 and effect size > 1.

```r
statcrit = which(ttest_results[,3]<1e-4)
effcrit = which(abs(ttest_results[,2])>1)
totcrit = which((ttest_results[,3]<1e-4) & (abs(ttest_results[,2])>1))
length(totcrit)
```

```
## [1] 506
```

```r
plot(ttest_results[,2], -log10(ttest_results[,3]), xlab = "Difference of means", ylab = "-log10(p_values
points(ttest_results[statcrit,2],
       -log10(ttest_results[statcrit,3]), col='green')
```

```
points(ttest_results[effcrit,2],
       -log10(ttest_results[effcrit,3]), col='blue')
points(ttest_results[totcrit,2],
       -log10(ttest_results[totcrit,3]), col='red')
```



Green: genes satisfy the p-value criterion (p-value < 1e-4) Blue: genes satisfy the effect size criterion (effect size > 1) Red: genes satisfy both 2 criterion Black: genes do not satisfy both 2 criterion

Tables of genes:

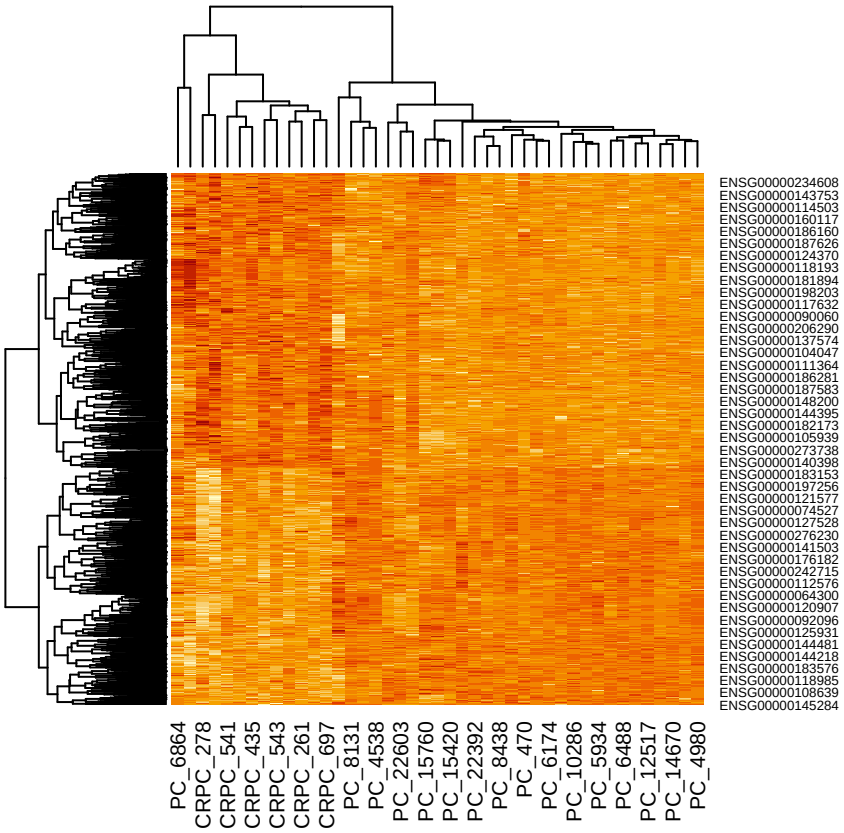
```
gene_list = ttest_results[(ttest_results$p.value<1e-4) & abs(ttest_results$dm)>1,]
# Ordering genes by adjusted p-values
ordered_gene_list <- gene_list[order(gene_list$p.value.adj), ]
head(ordered_gene_list, 5)
```

```
##                 statistic        dm      p.value  p.value.adj
## ENSG00000147647 -9.213596 -5.265276 1.534393e-11 2.530522e-07
## ENSG00000124205 -8.729041 -5.762183 6.767890e-11 2.872607e-07
## ENSG00000122877 -8.803296 -2.794893 5.381532e-11 2.872607e-07
## ENSG00000184588 -8.647583 -1.946615 8.709092e-11 2.872607e-07
## ENSG00000007908 -8.648253 -3.524544 8.691019e-11 2.872607e-07
```

```
write.table(ordered_gene_list,
            file = "Differentially_Expressed_Genes.tsv",
            sep = "\t",
            quote = FALSE)
```

Analysis 5:

```
count_matrix_de_ttest <- count_matrix_norm_log[row.names(ttest_results) %in% row.names(ttest_results)[(
heatmap(as.matrix(count_matrix_de_ttest), distfun = correlation_dist)
```



Analysis 6:

```
library(AnnotationDbi)
library(biomaRt)
mart <- useDataset("hsapiens_gene_ensembl", useMart("ensembl"))
mappings_ensembl_entrez <- getBM(filters = "ensembl_gene_id",
  attributes = c("ensembl_gene_id", "entrezgene_id"),
  values = rownames(count_matrix), mart = mart)

library(org.Hs.eg.db)
map_ke <- as.list(org.Hs.egPATH)
kegg_entrez_pathway = map_ke[!is.na(match(mappings_ensembl_entrez$entrezgene_id, names(map_ke)))]

ensembl2kegg_pathway = list()
for (i in 1:nrow(mappings_ensembl_entrez)) {
  temp = kegg_entrez_pathway[as.character(mappings_ensembl_entrez[i, ]$entrezgene_id)]
  if (!is.na(names(temp))){
    ensembl2kegg_pathway[mappings_ensembl_entrez[i, ]$ensembl_gene_id] = temp
  }
}
```

```
pvalues <- c()
# Calculate N - the total number of genes in all pathways
N <- sum(!is.na(unlist(ensembl2kegg_pathway)))
```

```r
# Calculate n, corresponding to the number of genes in the KEGG pathways
ns <- table(unlist(ensembl2kegg_pathway))
# Calculate M, corresponding to the number of the genes in our gene list (i.e. differentially expressed
ensembl2kegg_pathway_de <- ensembl2kegg_pathway[names(ensembl2kegg_pathway) %in% rownames(gene_list)]
M <- sum(!is.na(unlist(ensembl2kegg_pathway_de)))
# Calculate k, corresponding to the number of the DE genes in the KEGG pathways
ks <- rep(0, length(ns))
res <- table(unlist(ensembl2kegg_pathway[rownames(gene_list)]))
names(ks) <- names(ns)
ks[names(res)] <- res
for (i in 1:length(ns)) {
contingency_table <- data.frame(matrix(NA, nrow = 2, ncol = 2))
contingency_table[1, 1] <- N - ns[i] - (M - ks[i])
contingency_table[1, 2] <- ns[i] - ks[i]
contingency_table[2, 1] <- M - ks[i]
contingency_table[2, 2] <- ks[i]
results <- fisher.test(contingency_table, alternative = "greater")
pvalues[i] <- results$p.value }
names(pvalues) <- names(ns)
significance_threshold <- 0.05
sum(pvalues < significance_threshold)
```

```
## [1] 19
```

Enriched pathways

```r
enriched_pathways <- as.data.frame(sort(pvalues[pvalues < significance_threshold]))
colnames(enriched_pathways) <- c("p-value")
rownames(enriched_pathways)[nrow(enriched_pathways)]
```

```
## [1] "04640"
```

```r
head(enriched_pathways, 10)
```

```
##            p-value
## 05146 0.0001131809
## 04512 0.0001575726
## 00140 0.0001627656
## 04110 0.0002390672
## 05150 0.0002752227
## 04974 0.0003037818
## 04621 0.0055106550
## 04270 0.0060008124
## 00500 0.0067983039
## 00590 0.0156606348
```

```r
write.table(enriched_pathways,
            file = "Enriched pathway.tsv",
            sep = "\t",
            quote = FALSE)
```

Analysis 8:

Let's consider all the results we got so far. The hierarchical clustering turned out interestingly due to the presence of some samples in the opposite group. There are 2 samples (PC 6864 and PC 9324) from PC group that fell far away from their group, which is confirmed by the following PCA analysis. Thus, these 2 samples

are outliers which can represent different in biological sense compared to the PC group. According to the Cancer research, it is likely that these samples share the molecular characteristics with CRPC because of some specific genetic alterations that might lead to different molecular pathways. Together with analysis 6, we should have a clearer picture about the potential biological pathways which not only help to distinguish the two sample group but also imply key player behind the genetic alteration in PC contributing to treatment resistance.