# Project02 - Group01

Eva, Tobi, Kathi, Laura

14 Juni 2019

## data loading

## data scaling

After checking for normalization, we scaled our data in the first place to provide the scaled data for further analysis.

```
list = list(Treated,Untreated)
nlist = lapply(list,scale)
Treated = as.data.frame(nlist[[1]])
Untreated = as.data.frame(nlist[[2]])
Fold_Change = Treated - Untreated
Fold_Change = data.frame(Fold_Change)
rm(NCI_TPW_gep_treated,NCI_TPW_gep_untreated,list,nlist)
```

# 1. broad analysis

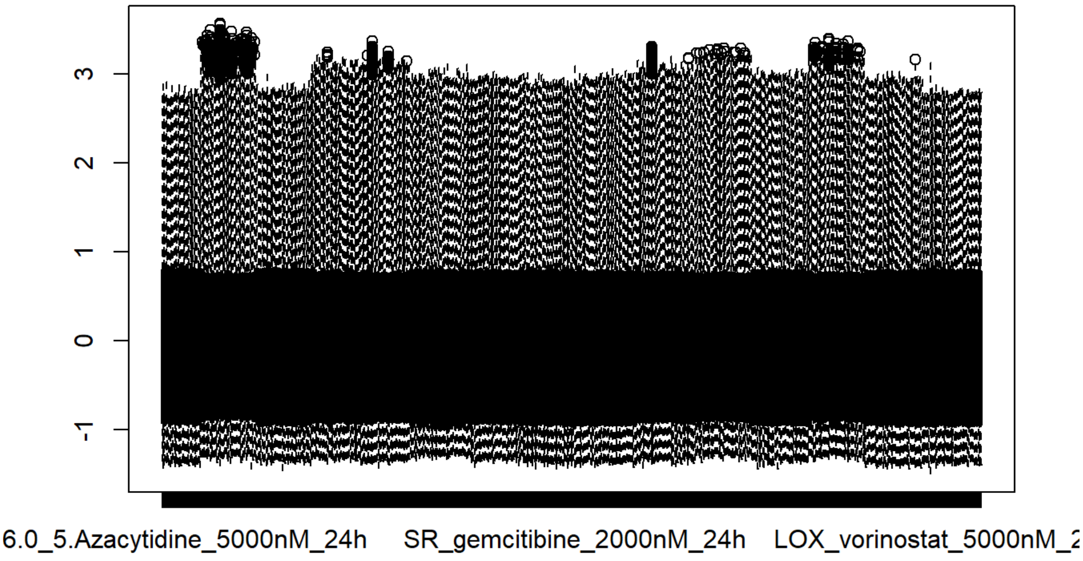
## installing packages

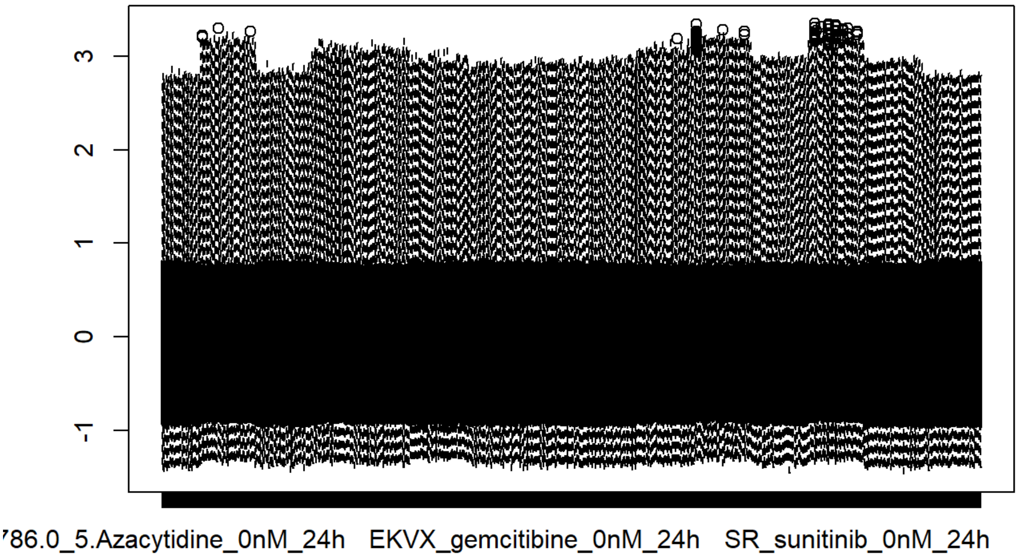
```
library(cluster)
```

```
## Warning: package 'cluster' was built under R version 3.5.3
```

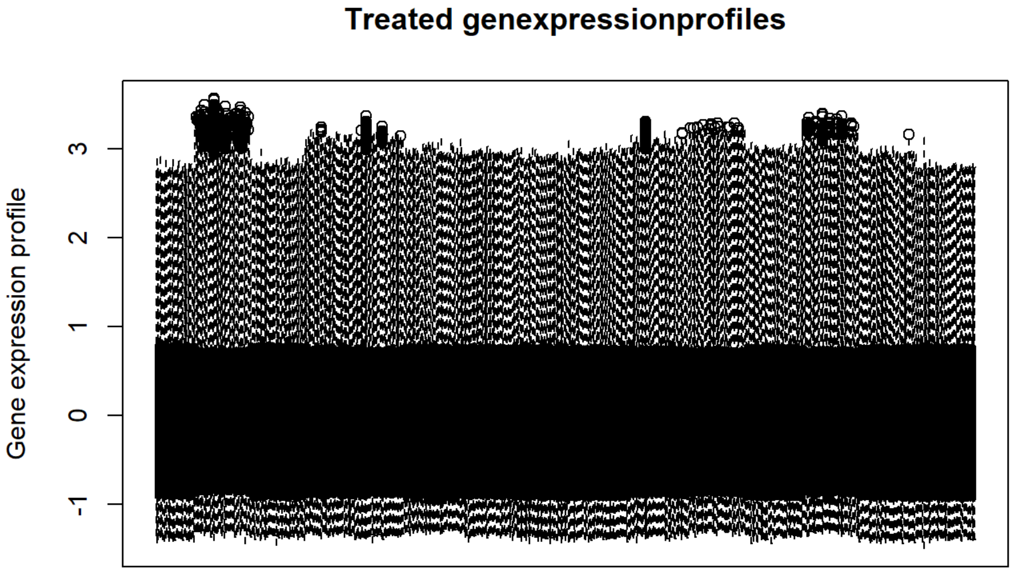
## Boxplots (already normalized)

This step was done before scaleing the data. The boxplots showed a deviation which is the reason for scaling the data.

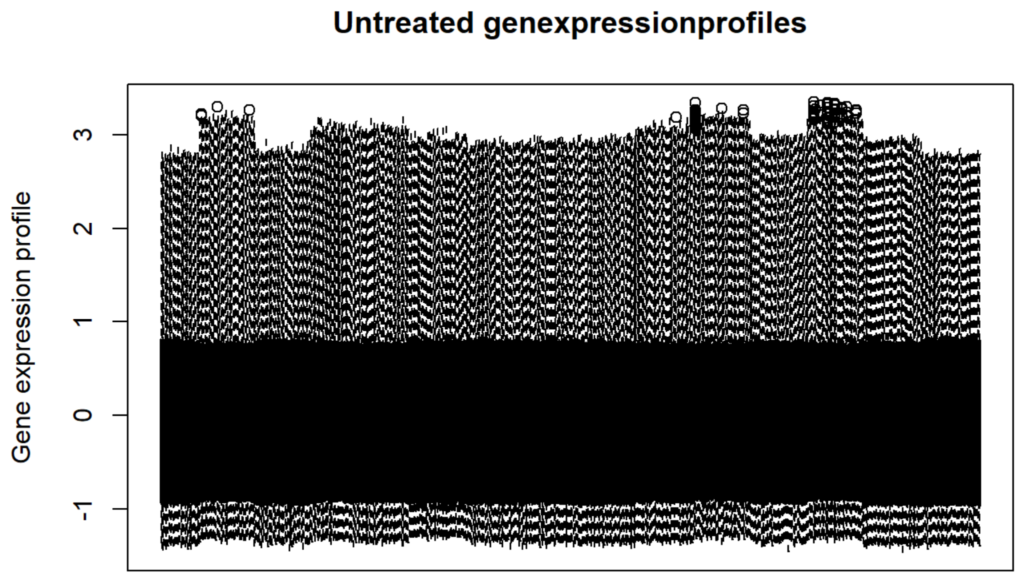




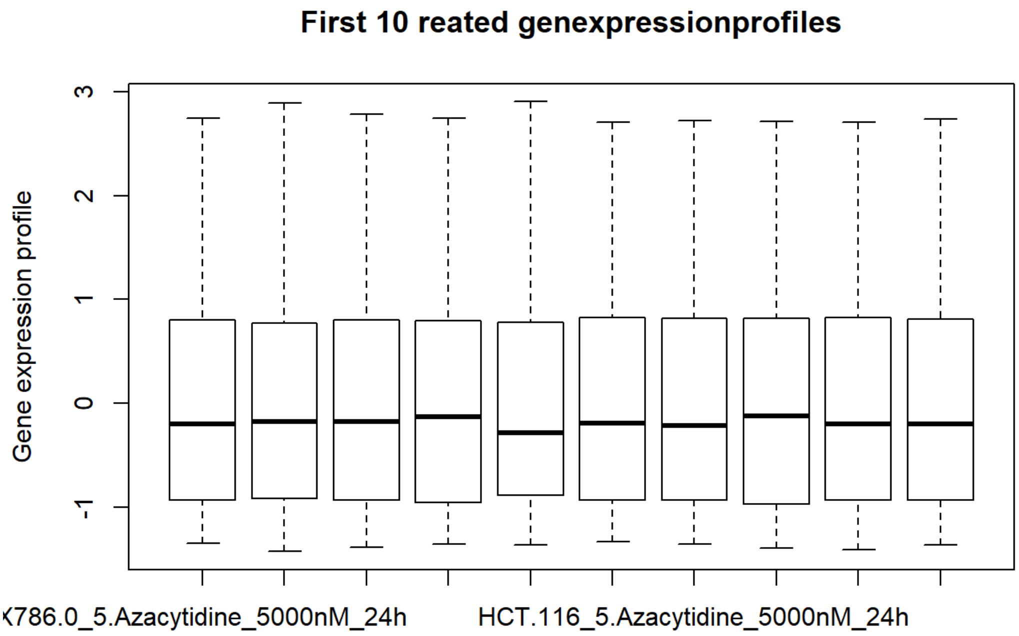
```r
boxplot(Treated,  ylab = "Gene expression profile", main = "Treated genexpressionprofiles",xaxt = "n")
```

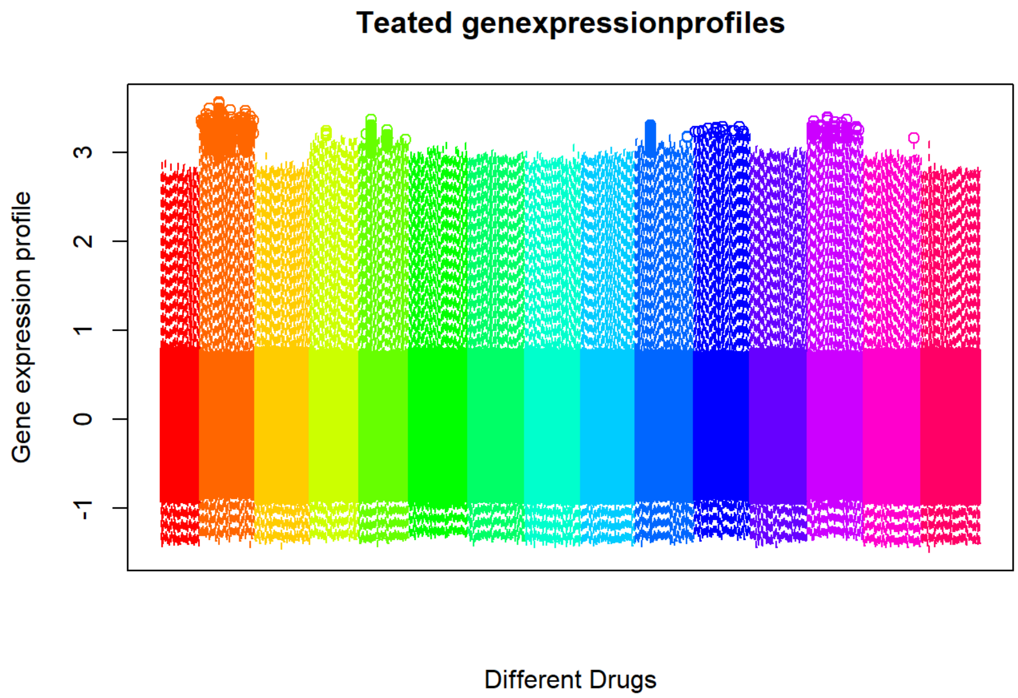


```r
boxplot(Untreated, ylab = "Gene expression profile", main = "Untreated genexpressionprofiles",xaxt = "n")
```



```
boxplot(Treated[,1:10], ylab = "Gene expression profile", main = "First 10 reated genexpressionprofiles")
```

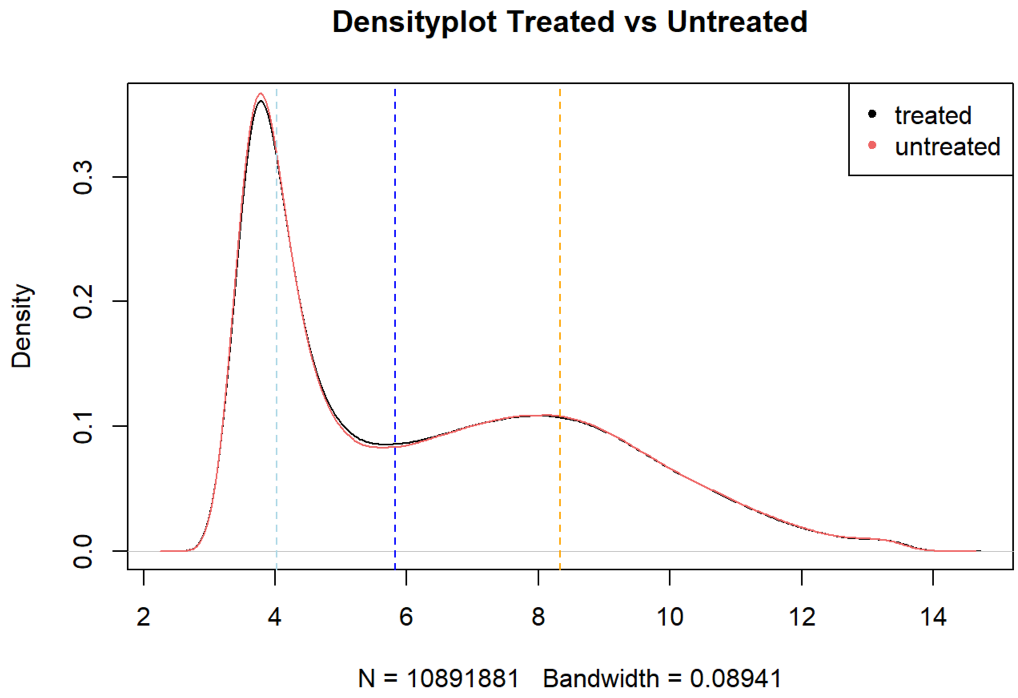




## Densityplot

The abline shows the 3 quantiles ( 25% 50% 75% )

```
NCI_TPW_gep_treated = readRDS(paste0(wd, "/Data/NCI_TPW_gep_treated.rds"))
NCI_TPW_gep_untreated = readRDS(paste0(wd, "/Data/NCI_TPW_gep_untreated.rds"))
plot(density(NCI_TPW_gep_treated), "Densityplot Treated vs Untreated")
lines(density(NCI_TPW_gep_untreated), col = "indianred2")
legend("topright", legend = c("treated", "untreated"), col = c("black", "indianred2"), pch = 20)
abline(v = quantile(NCI_TPW_gep_treated)[2:4], col = c("lightblue", "blue",  "orange"), lty = 2)
```



## k-means clustering

To look for clusters in the raw data we performed a k-menas clustering and searched for potentially clusters.

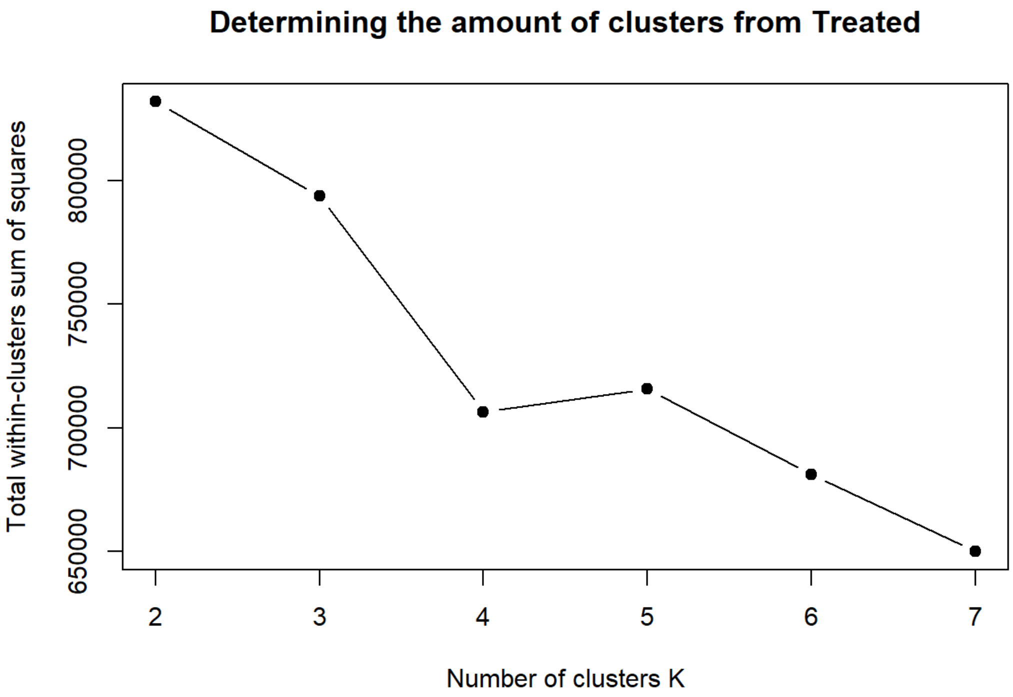
```
##      Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.002893 0.029461 0.069002 0.124300 0.135476 2.138284
```

```
## [1] 3325  819
```

```
## [1] 758323.6
```

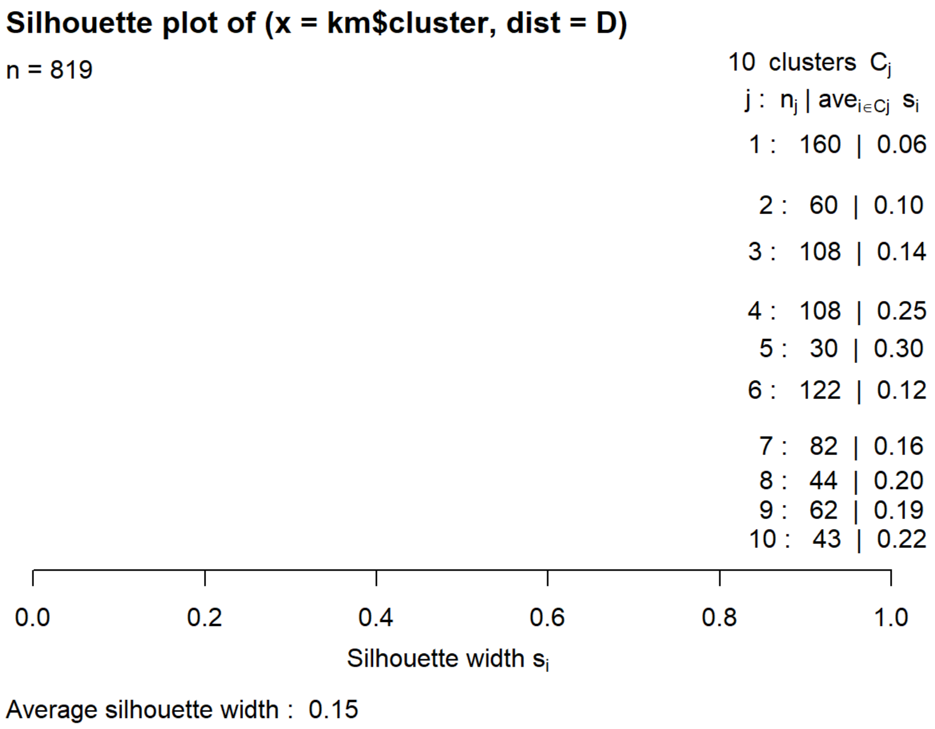
```
## [1] 832093.5
```

```
#running a loop for the best n (searching for "ellbow")
wss = sapply(2:7, function(k) {
kmeans(x = t(topVarTreated75), centers = k)$tot.withinss})
plot(2:7, wss, type = "b", pch = 19, xlab = "Number of clusters K", ylab = "Total within-clusters sum of
squares", main = "Determining the amount of clusters from Treated")
```



As we wanted an “ellbow”to get a good result we can say in a way that our data are not really good to cluster. To look in a other way, we also provided the clusters by the silhouette-method.

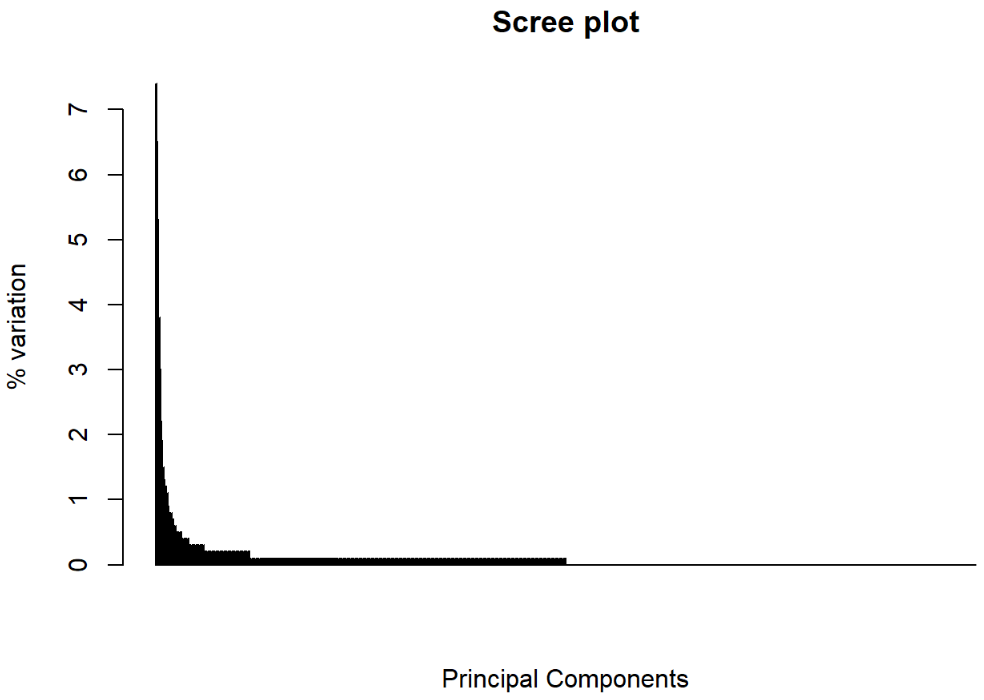
```
# Using the silhouett method
D = dist(t(topVarTreated75))
km = kmeans(x = t(topVarTreated75), centers = 10, nstart = 10)
s = silhouette(km$cluster, D)
plot(s)
```



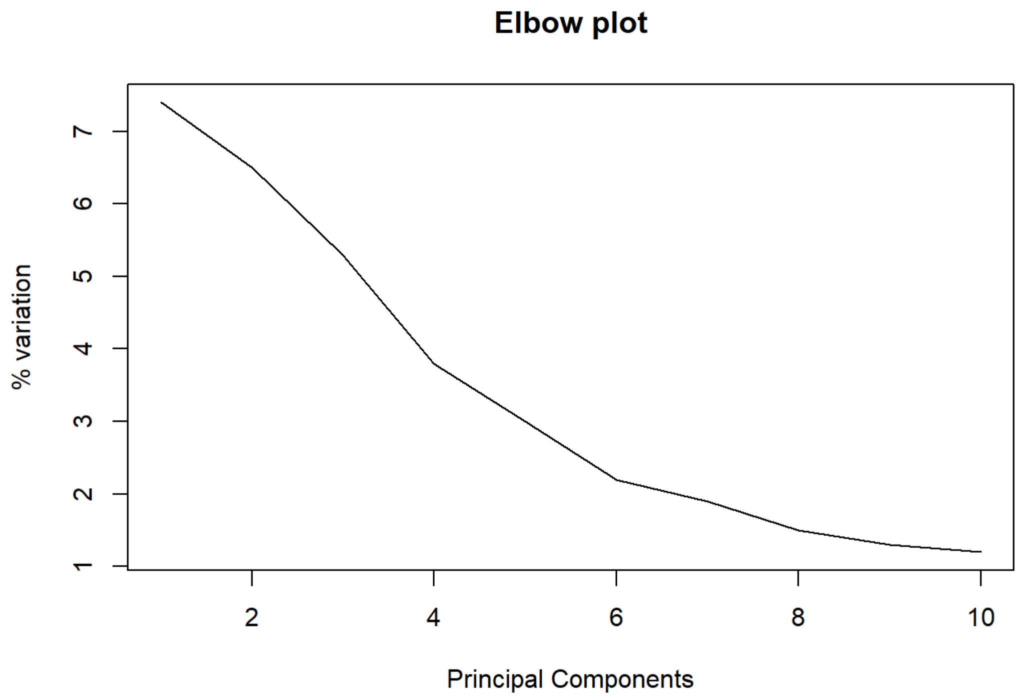
## PCA

```
pca <- prcomp(t(Fold_Change), scale = TRUE)
```

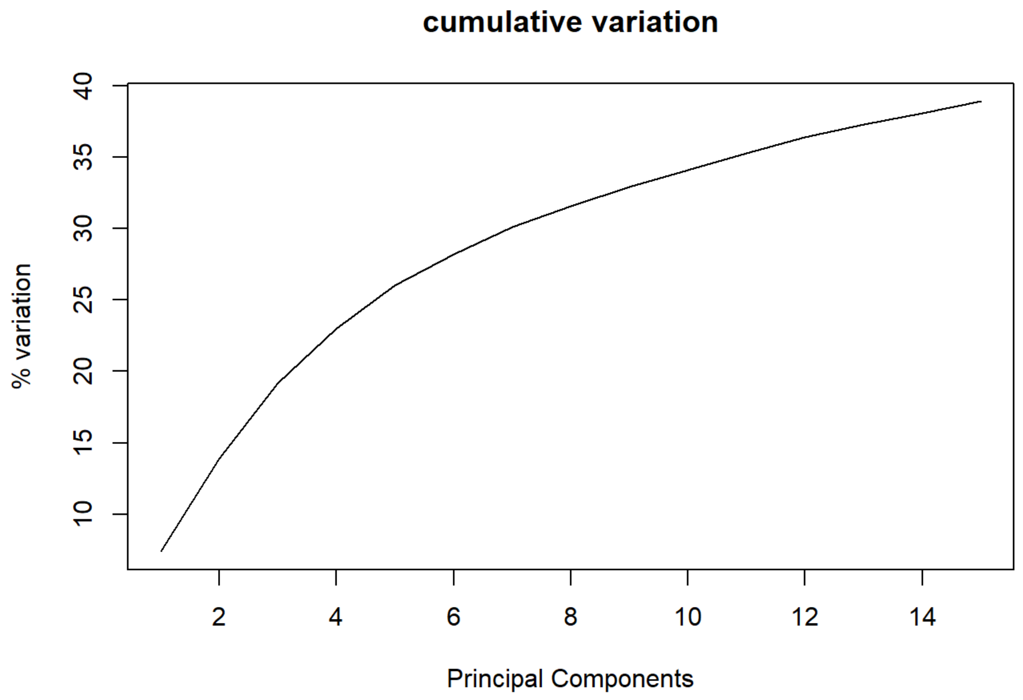
```
# sdev calculates variation each PC accounts for
pca.var <- pca$sdev^2
# since percentages make more sense then normal variation values
# calculate % or variation, which is much more interesing
pca.var.per <- round(pca.var/sum(pca.var)*100, 1)

barplot(pca.var.per, main = "Scree plot", xlab = "Principal Components", ylab = "% variation")
```



```
plot(pca.var.per[1:10], main = "Elbow plot", type = "l", xlab = "Principal Components", ylab = "% variation")
```



```
plot(cumsum(pca.var.per[1:15]), main = "cumulative variation", type = "l", xlab = "Principal Components",
ylab = "% variation")
```



```
#creating data frame with all pcs
#cleaning up sample names as they differed between matrices
pca.data <- data.frame(pca$x)
rownames(pca.data) <- gsub(x = rownames(pca.data), pattern = "X786", replacement = "786")
pca.data <- cbind(sample =rownames(pca.data), pca.data)
```

```
## get names of top 10 genes that contribute most to pc1
loading_scores_1 <- pca$rotation[,1]
gene_score <- abs(loading_scores_1) ## sort magnitude
gene_score_ranked <- sort(gene_score, decreasing = TRUE)


top_10_genes <- names(gene_score_ranked[1:10])
top_10_genes # show names of top 10 genes
```

```
##  [1] "DNAJC2"  "NGDN"    "GTPBP4"  "CCDC59"  "DNTTIP2" "AKAP8"   "PAPSS1" 
##  [8] "TRMT1"   "BRF2"    "YRDC"
```

```
### Metadata color matrix for coloring
Metadata$sample <- gsub(x = Metadata$sample, pattern = "-", replacement = ".")

metad.cl <- subset(Metadata, Metadata$sample %in% pca.data$sample)
## adjust row length of metadata to pca.data


metad.cl$mechanism <- Drug_Annotation$Mechanism[match(metad.cl$drug, Drug_Annotation$Drug)]
metad.cl$msi <- Cellline_Annotation$Microsatellite_instability_status[match(metad.cl$cell, Cellline_Annot
ation$Cell_Line_Name)]
```

```
library(viridis)
```

```
## Warning: package 'viridis' was built under R version 3.5.3
```

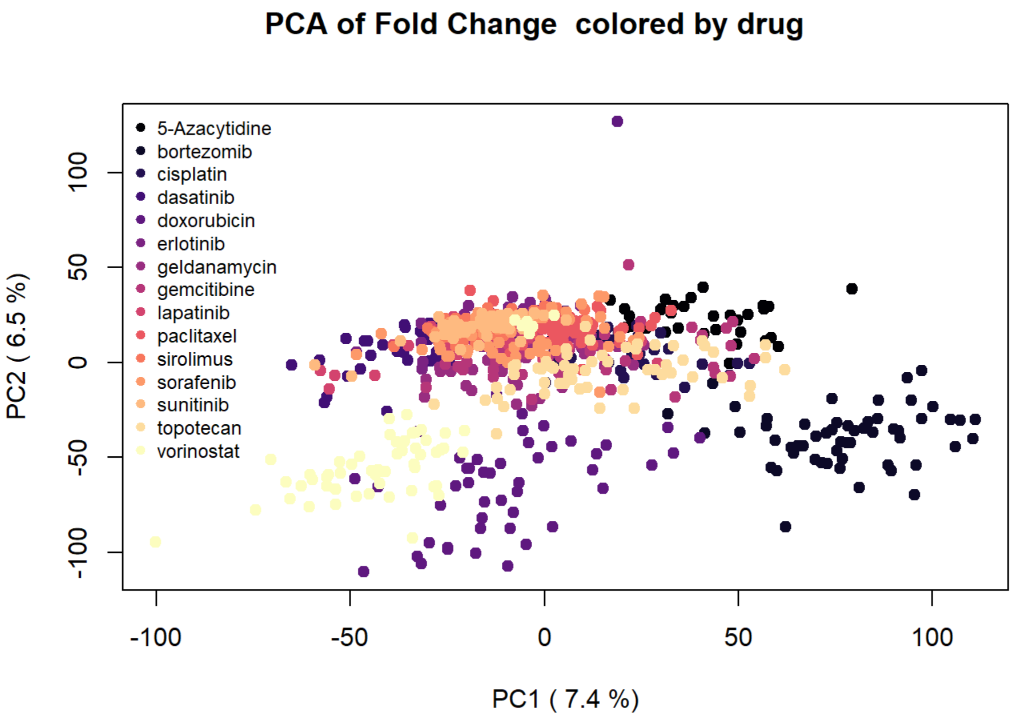
```
## Loading required package: viridisLite
```

```
## Warning: package 'viridisLite' was built under R version 3.5.3
```

```
# plotting all informative PCs
#color vectors for coloring by drug and tissue
viridis <- viridis(9)
color_tissue = viridis[metad.cl$tissue]
tissue <- levels(metad.cl$tissue)

magma <- magma(15)
color_drug = magma[metad.cl$drug]
drug <- levels(metad.cl$drug)


## colored by drug
#plot PC1 and PC2
plot(pca$x[,1],
     pca$x[,2],
     col = color_drug,
     pch = 19,
     xlab = paste("PC1 (",pca.var.per[1],"%)"),
     ylab = paste("PC2 (",pca.var.per[2],"%)"))
#create legend
legend("topleft",
       legend = drug,
       col = magma,
       pch = 19,
       xpd = "TRUE",
       bty = "n",
       cex = 0.75
)
```

```
## Warning in par(xpd = xpd): NAs durch Umwandlung erzeugt
```

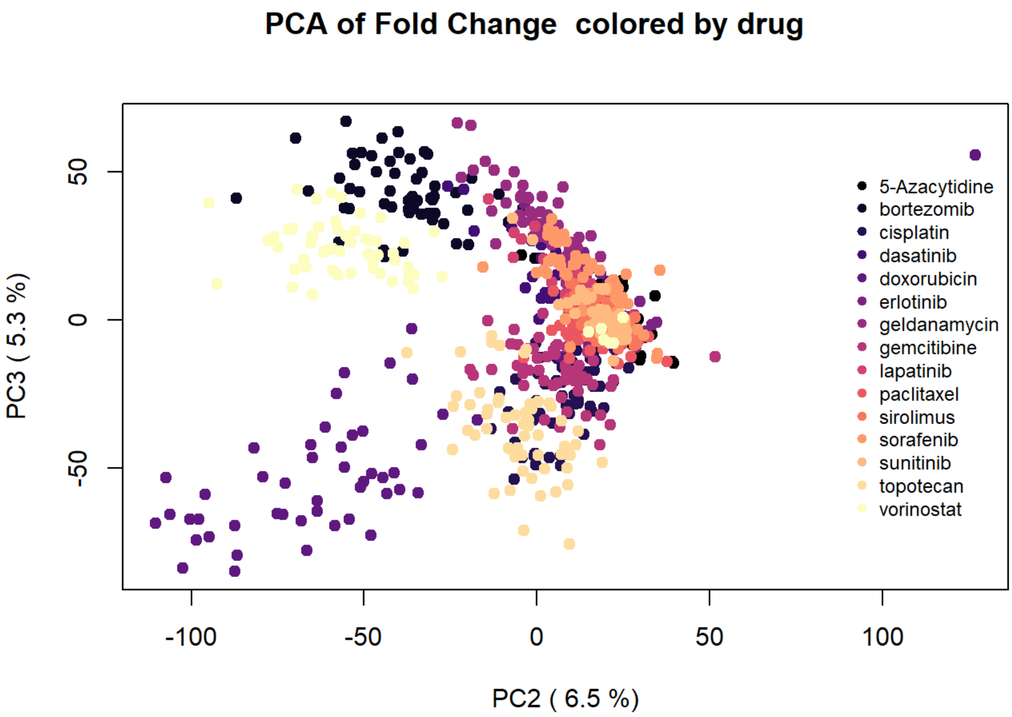
```
#create title
mtext("PCA of Fold Change  colored by drug",
      side = 3,
      line = -2,
      cex = 1.2,
      font = 2,
      outer = TRUE)
```



```
#plot PC2 and PC3
plot(pca$x[,2],
     pca$x[,3],
     col = color_drug,
     pch = 19,
     xlab = paste("PC2 (",pca.var.per[2],"%)"),
     ylab = paste("PC3 (",pca.var.per[3],"%)"))
#create legend
legend("right",
       legend = drug,
       col = magma,
       pch = 19,
       xpd = "TRUE",
       bty = "n",
       cex = 0.75,
       inset = c(0, 2)
)
```

```
## Warning in par(xpd = xpd): NAs durch Umwandlung erzeugt
```

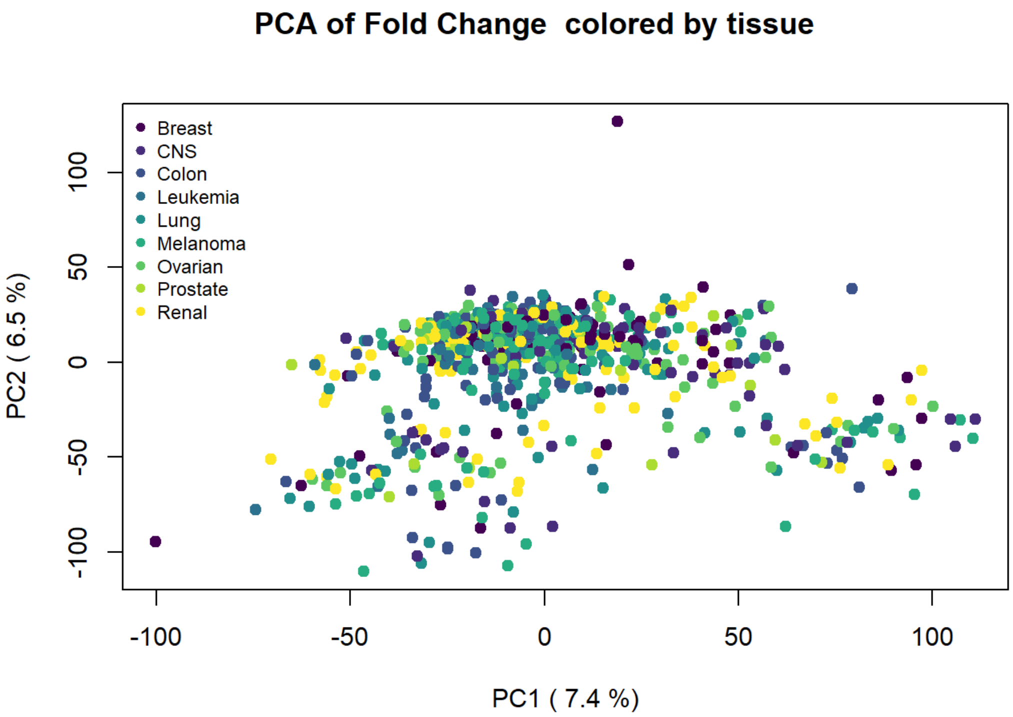
```
#create title
mtext("PCA of Fold Change  colored by drug",
      side = 3,
      line = -2,
      cex = 1.2,
      font = 2,
      outer = TRUE)
```



```
## colored by tissue
#plot PC1 and PC2
plot(pca$x[,1],
     pca$x[,2],
     col = color_tissue,
     pch = 19,
     xlab = paste("PC1 (",pca.var.per[1],"%)"),
     ylab = paste("PC2 (",pca.var.per[2],"%)"))
#create legend
legend("topleft",
       legend = tissue,
       col = viridis,
       pch = 19,
       xpd = "TRUE",
       bty = "n",
       cex = 0.75
)
```

```
## Warning in par(xpd = xpd): NAs durch Umwandlung erzeugt
```

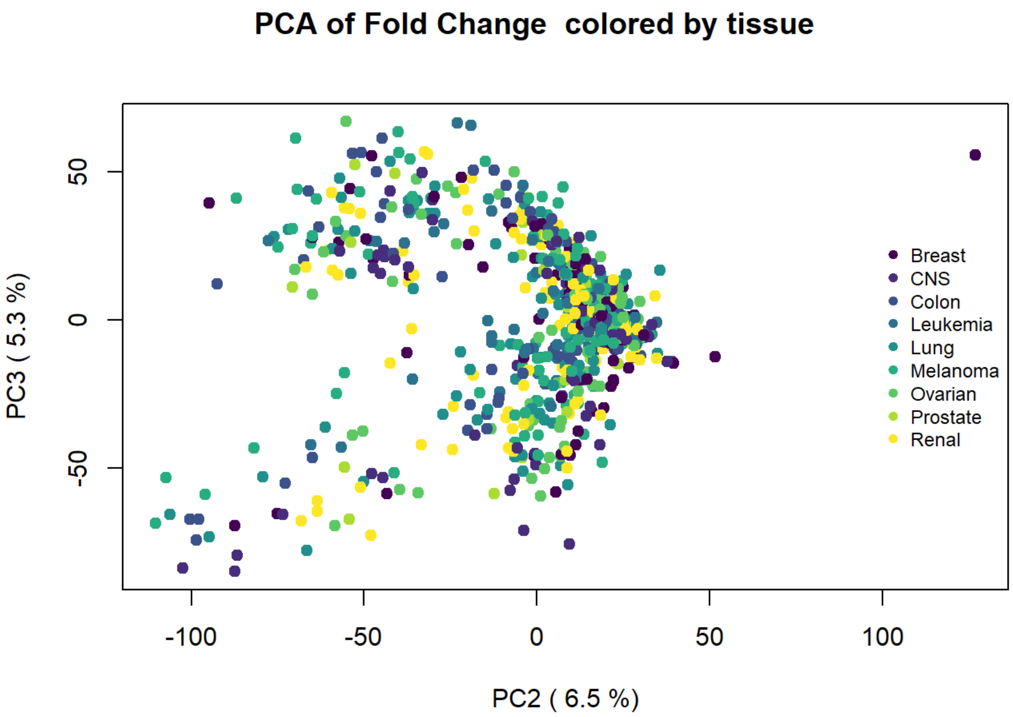
```
#create title
mtext("PCA of Fold Change  colored by tissue",
      side = 3,
      line = -2,
      cex = 1.2,
      font = 2,
      outer = TRUE)
```



```
#plot PC2 and PC3
plot(pca$x[,2],
     pca$x[,3],
     col = color_tissue,
     pch = 19,
     xlab = paste("PC2 (",pca.var.per[2],"%)"),
     ylab = paste("PC3 (",pca.var.per[3],"%)"))
#create legend
legend("right",
       legend = tissue,
       col = viridis,
       pch = 19,
       xpd = "TRUE",
       bty = "n",
       cex = 0.75,
       inset = c(0, 2)
)
```

```
## Warning in par(xpd = xpd): NAs durch Umwandlung erzeugt
```

```
#create title
mtext("PCA of Fold Change  colored by tissue",
      side = 3,
      line = -2,
      cex = 1.2,
      font = 2,
      outer = TRUE)
```


PCA of Fold Change  colored by tissue
PC3 ( 5.3 %)
50
0
-50
-100
-50
0
50
100
PC2 ( 6.5 %)
● Breast
● CNS
● Colon
● Leukemia
● Lung
● Melanoma
● Ovarian
● Prostate
● Renal