Broad Analysis complete documentation.

Franziska Heinkele and Laura Plutowski

26 Juni 2019

Introduction

Analyzing big data frames becomes more and more important for working on biological issues. This is the reason why we want to have a look at such an exploratory analysis. Therefore, we focus on data that report different cellular responses to drug perturbations in cancer treatment. First of all, we will perform a broad exploratory analysis over the whole data set. Then we will continue looking at the specific anticancer drug vorinostat.

1.Broad Analysis

First of all, we want to explore the whole data from all 15 drug responses including the treated and untreated data sets. Therefore we perform the following analysis steps:

- Boxplot with treated data
- Finding highly variable genes using Seurat
- Analysis of Fold Change matrix
- Different PCAs

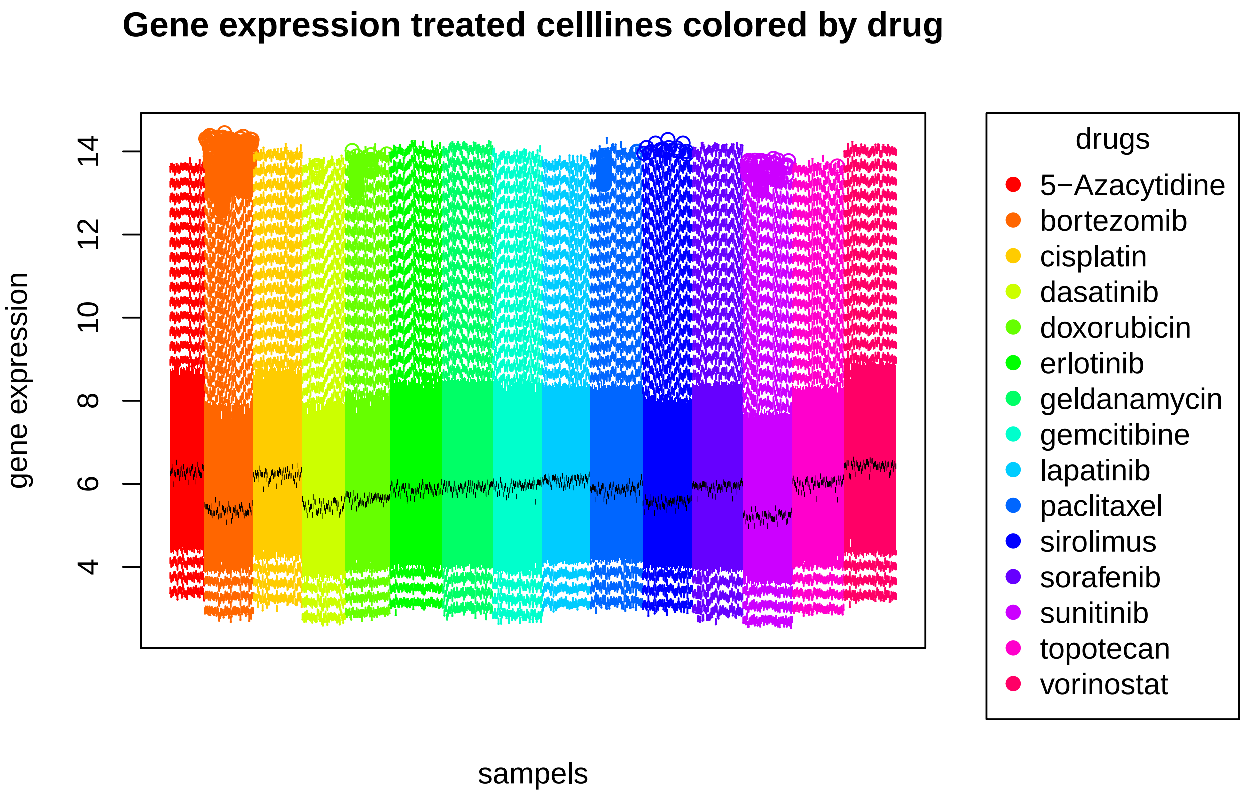
Before we are able to start the broad analysis, we load the needed data. The data sets are taken from the NCI Transcriptional Pharmacodynamics Workbench, including the effect on 13.299 genes from 61 cell lines treated with 15 different anticancer agents.

```
library(readr)

Untreated <- readRDS(paste0(wd,"/data/NCI_TPW_gep_untreated.rds"))
Treated <- readRDS(paste0(wd,"/data/NCI_TPW_gep_treated.rds"))
Metadata = read.table(paste0(wd,"/data/NCI_TPW_metadata.tsv"),
                      header = TRUE, sep ="\t", stringsAsFactors = TRUE)

Treated <- as.data.frame(Treated)
Untreated <- as.data.frame(Untreated)
```

1.1 BOXPLOT OF TREATED DATA

To visualize our data, we perform a boxplot of the treated data over all 15 drugs.

```
# levels for coloring
drug <- Metadata$drug
# 15 diffrent colors, for each drug one
palette(rainbow(15))
# Boxplot
par(mar=c(5, 4, 5, 9))
boxplot(Treated, medcol="black", border = drug, col= drug,
        xlab="sampels", ylab="gene expression",
        main= "Gene expression treated celllines colored by drug",
        names= FALSE, xaxt= "n", boxwex=1, boxlty =0)

#add a legend to see which color corresponds to which drug:
levels <- as.factor(levels(drug))
legend("topright", inset = c(-0.4,0), legend= levels(drug), xpd = TRUE, pch=19,
       col = levels, title = "drugs")
```

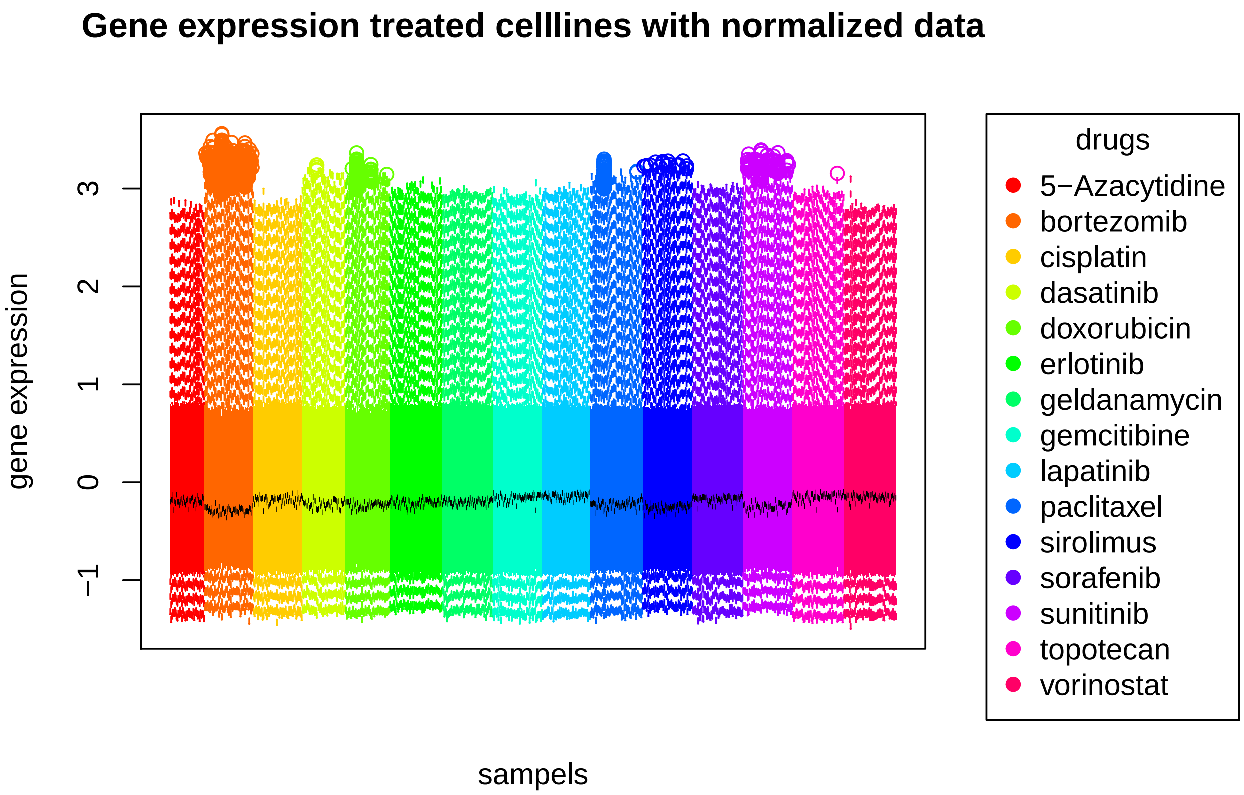


The different colors indicate the 15 different anticancer drugs. We can can clearly identify 15 different boxes, each box belonging to one medicine. This indicates that we have batches between all 15 drugs. One reason for this could be, that the drug treatments were performed on different days under slightly different conditions like air pressure or room temperature.

If we normalize the data, we can remove the batch effects. Our plot will change in the following way:

```
# normalize the data
Untreated_norm <- apply(Untreated, 2, function(x){
  (x - mean(x)) / sd(x)
})


Treated_norm <- apply(Treated, 2, function(x){
  (x - mean(x)) / sd(x)
})
```

```
# repeat creation of boxplot

# levels for coloring
drug <- Metadata$drug
# 15 diffrent colors, for each drug one
palette(rainbow(15))
# Boxplot
par(mar=c(5, 4, 5, 9))
boxplot(Treated_norm, medcol="black", border = drug, col= drug,
        xlab="sampels", ylab="gene expression",
        main= "Gene expression treated celllines with normalized data",
        names= FALSE, xaxt= "n", boxwex=1, boxlty =0)

#add a legend to see which color corresponds to which drug:
levels <- as.factor(levels(drug))
legend("topright", inset = c(-0.4,0), legend= levels(drug), xpd = TRUE,
       pch=19, col = levels, title = "drugs")
```



We see that after normalization of the data the batches were removed.

1.2 HIGHLY VARIABLE GENE ANALYSIS

We aim to find genes which vary the most in Untreated and Treated celllines. Therefore we work with Seurat packages, which need to be installed: install.packages('Seurat') For the seurat library to be available, digest packages need to be installed: install.packages("digest")

```
library(Seurat)
```

In the following code, we use the abbreviation **"BroadAnU"** representative for "Broad analysis untreated" to achieve a better readability. First we create a new object that contains the Untreated data. Then 2000 most variable genes are selected from the Untreated data.

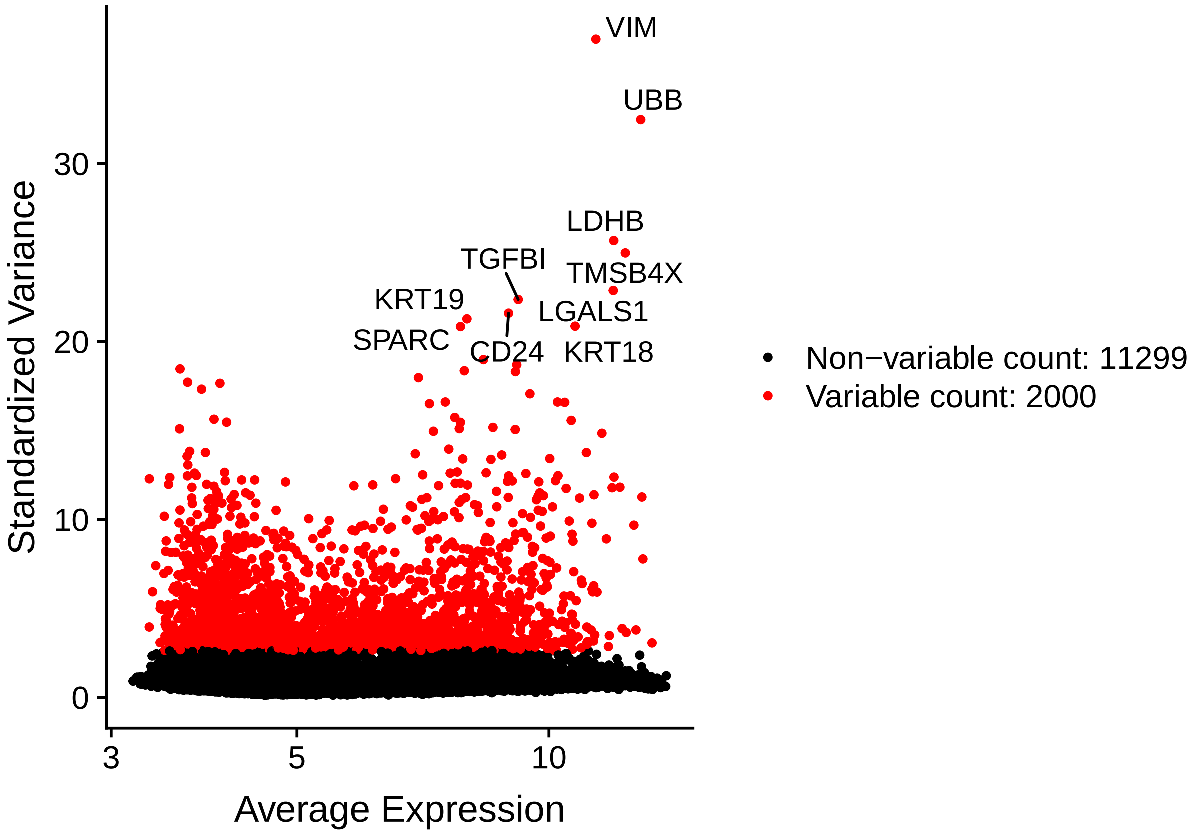
```
BroadAnU <- CreateSeuratObject(counts = Untreated, project = "BroadAnU")
BroadAnU <- FindVariableFeatures(BroadAnU, selection.method = "vst", nfeatures= 2000)
```

Next we want to identify the 10 most highly variable genes.

```
top10Untreated <- head(VariableFeatures(BroadAnU), 10)
```

First we plot variable features without labels, in a second step we add labels to the 10 most variable genes.

```
plotUntreated <- VariableFeaturePlot(BroadAnU)
plotUntreatedlabeled <- LabelPoints(plot = plotUntreated, points = top10Untreated,
                                    repel = TRUE, xnudge = 0, ynudge = 0)

plotUntreatedlabeled
```



For a comparison, we do the same procedure with the Treated data:

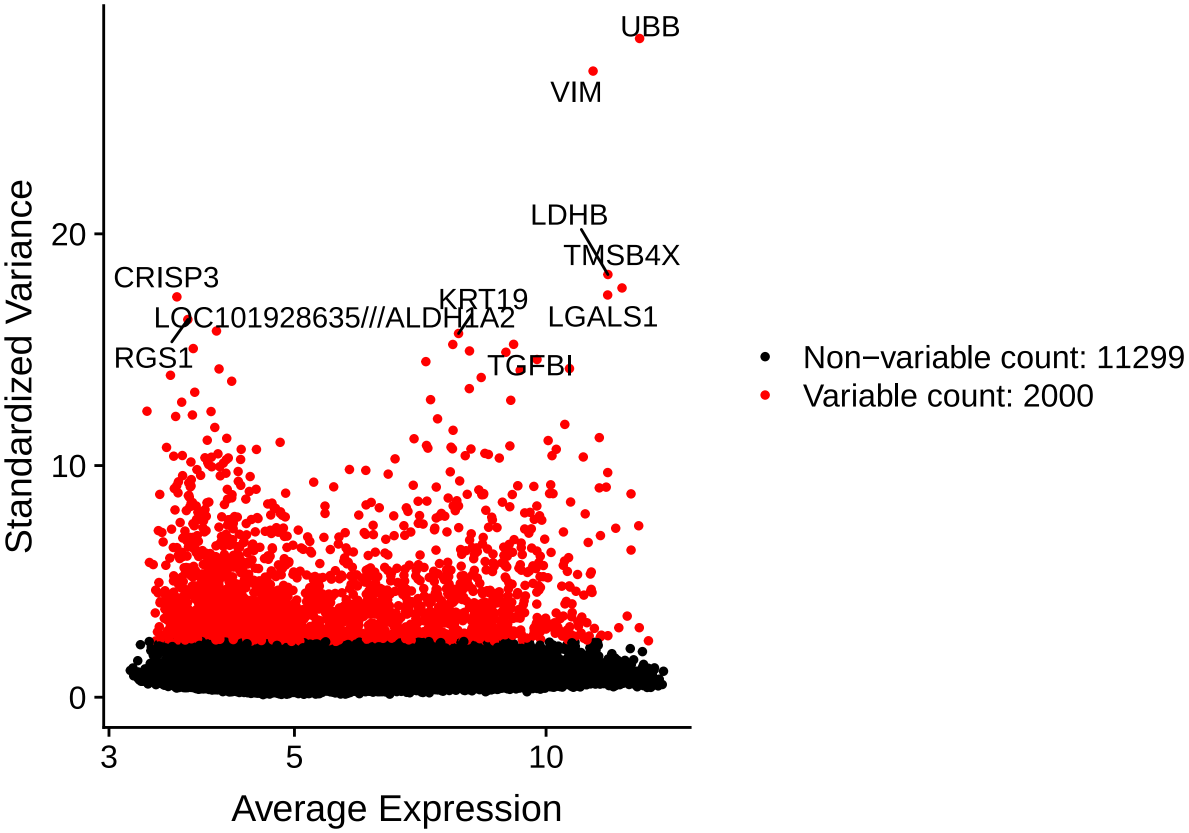
```
BroadAnT <- CreateSeuratObject(counts = Treated, project = "BroadAnT")
BroadAnT <- FindVariableFeatures(BroadAnT, selection.method = "vst", nfeatures= 2000)
```

Identify the 10 most highly variable genes:

```
top10Treated <- head(VariableFeatures(BroadAnT), 10)
```

Plot variable features with and without labels

```
plotTreated <- VariableFeaturePlot(BroadAnT)
plotTreatedlabeled <- LabelPoints(plot = plotTreated, points = top10Treated,
                                  repel = TRUE, xnudge = 0, ynudge = 0)
plotTreatedlabeled
```



Next we test, whether there are matches between Treated and Untreated most variable gene expression:

```
top10Untreated %in% top10Treated
```

```
##  [1]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE
```

7 matches were found! Which genes are in the Untreated but not in the Treated top 10?

```r
setdiff(top10Untreated, top10Treated)
```

```
## [1] "CD24"  "KRT18" "SPARC"
```

Which genes are in the Treated but not in the Untreated top 10?

```r
setdiff(top10Treated, top10Untreated)
```

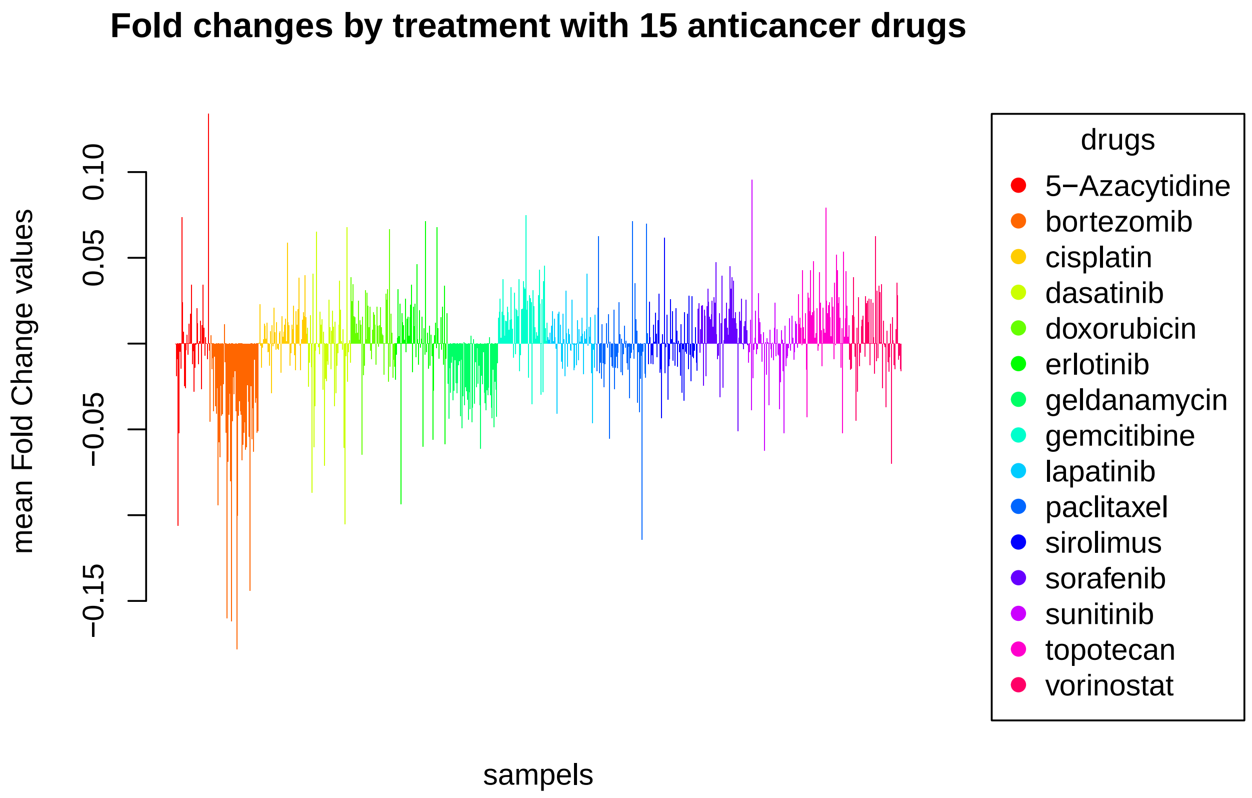
```
## [1] "CRISP3"                 "RGS1"
## [3] "LOC101928635///ALDH1A2"
```

We can conclude, that the treatment did not change the high variation for the matching genes. The matching genes might be not clearly affected by the drugs since their expression varies in treated celllines as well as in unterated celllines.

1.3 FOLD CHANGE ANALYSIS

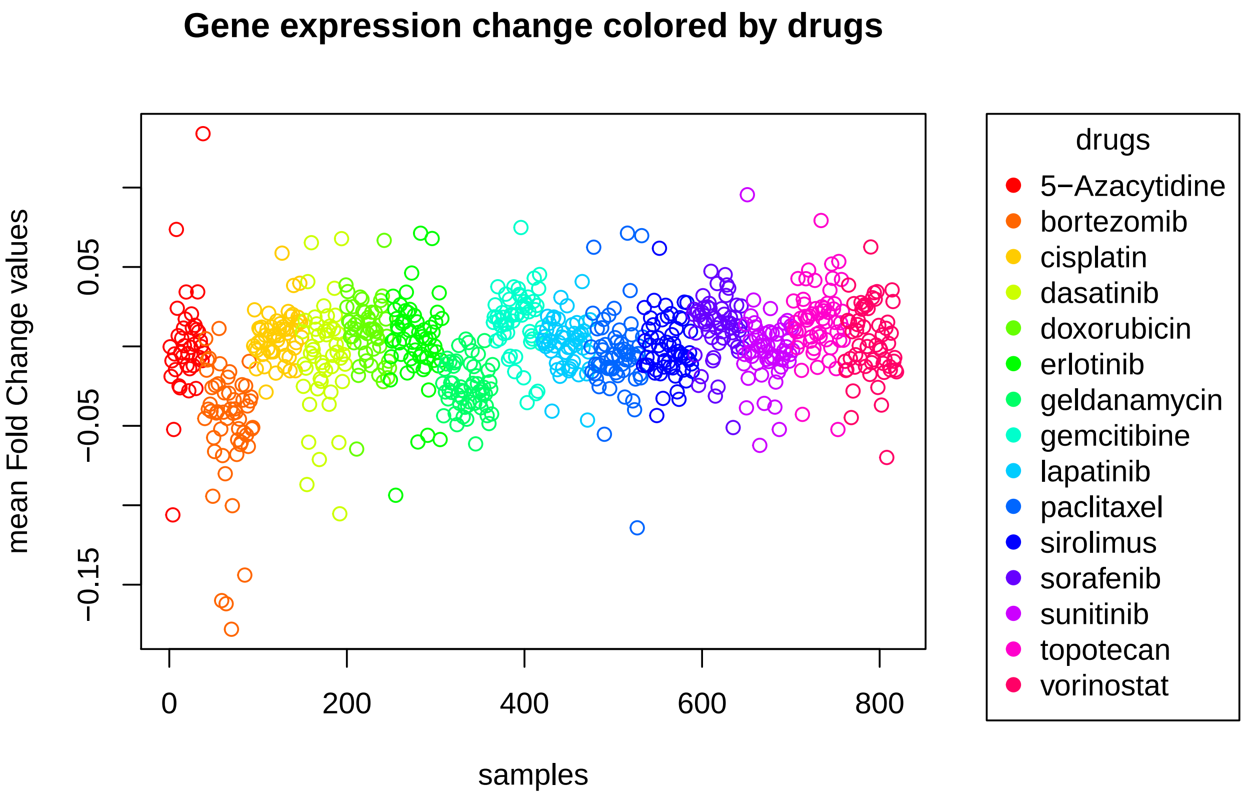
The fold change matrix (fold change = FC) describes the changes in gene expression between the treated and the untreated cell lines. First of all we visualize the FC with a bar chart which is colored by different drugs.

```r
# create FC data
FC_all = (Treated - Untreated)
FC_all_mean = colMeans(FC_all)

# create levels for coloring
drug <- Metadata$drug
palette(rainbow(15))

# create boxplot
par(mar=c(5, 4, 5, 9))
barplot( height = FC_all_mean, names= FALSE, col = drug, border = NA,
         main= "Fold changes by treatment with 15 anticancer drugs",
         xlab="sampels", ylab="mean Fold Change values")

# create a legend
levels <- as.factor(levels(drug))
legend("topright", inset = c(-0.4,0.0), legend= levels(drug), xpd = TRUE,
       pch=19, col = levels, title = "drugs")
```



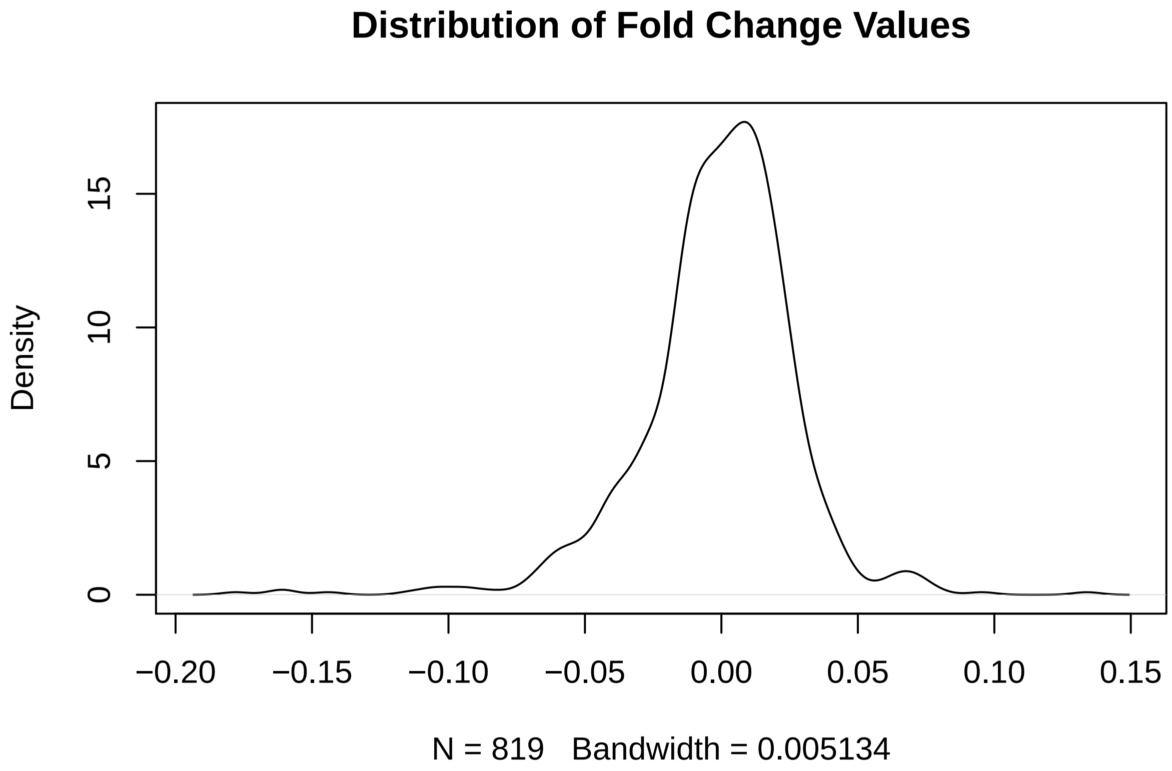
Here we can see that most FC values are in a similar value range. This becomes even clearer when looking at the following scatter plots. Only 5-Azacytidine and bortezomib have clear outliers.

```
# scatter plot
par(mar=c(5, 4, 5, 9))
plot(FC_all_mean, col= Metadata$drug, main="Gene expression change colored by drugs",
     xlab="samples",ylab="mean Fold Change values")

# creat legend
drug <- Metadata$drug
levels <- as.factor(levels(drug))
legend("topright", inset = c(-0.4,0), legend= levels(drug), xpd = TRUE, pch=19,
       col = levels, title = "drugs")
```



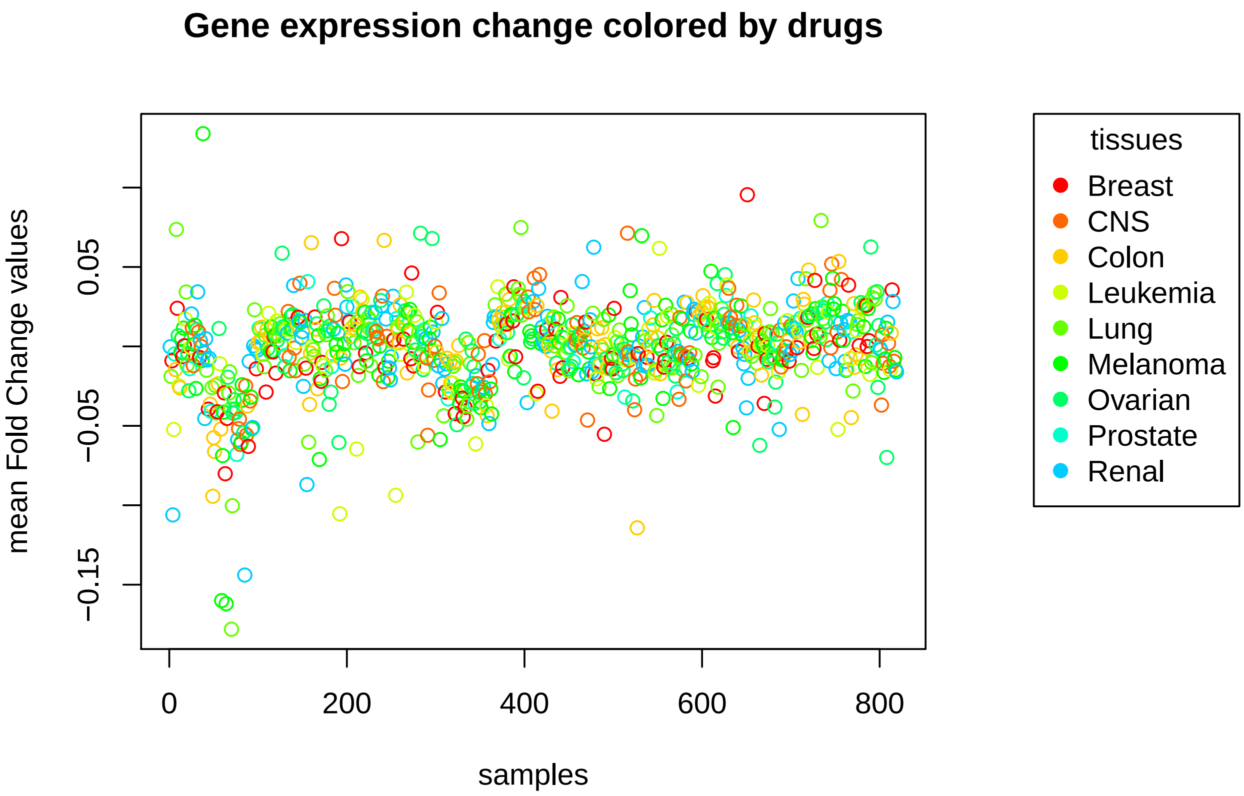
The shape of the density plot of the mean FC values shows that the data is approximately normally distributed:

```
plot(density(FC_all_mean), main= "Distribution of Fold Change Values")
```



Coloring the scatter plot according to the tissue type shows that there is no correlation between FC values and tissue type, if we look at all drugs:

```
# scatter plot
par(mar=c(5, 4, 5, 9))
plot(FC_all_mean, col= Metadata$tissue, main="Gene expression change colored by drugs",
     xlab="samples",ylab="mean Fold Change values")

# legend
tissue <- Metadata$tissue
levels <- as.factor(levels(tissue))
legend("topright", inset = c(-0.4,0), legend= levels(tissue), xpd = TRUE, pch=19, col = levels,
       title = "tissues")
```



We can identify the Top 10 values from the FC matrix, which indicate the ten most up and down regulated genes by a specific drug in a specific cell line.

```
### find all min and max
FC_all_min= (apply(FC_all,2,min))
FC_all_max= (apply(FC_all,2,max))

### sort min and max

# most down regulated genes
largest10_FC_all_min <- (sort(FC_all_min, decreasing = F)[1:10])
largest10_FC_all_min =as.data.frame(largest10_FC_all_min)
knitr::kable(largest10_FC_all_min, caption = "10 smallest FC values")
```

Table 1: 10 smallest FC values

	largest10_FC_all_min
T-47D__vorinostat__5000nM__24h	-9.062011
OVCAR-4__sorafenib__10000nM__24h	-7.601793
NCI-H522__geldanamycin__1000nM__24h	-7.280544
MALME-3M__vorinostat__5000nM__24h	-6.864732
OVCAR-4__doxorubicin__1000nM__24h	-6.843617
UACC-257__vorinostat__5000nM__24h	-6.813984
OVCAR-4__bortezomib__100nM__24h	-6.608203
NCI-H226__bortezomib__100nM__24h	-6.447166

	largest10_FC_all_min
COLO-205__bortezomib__100nM__24h	-6.391190
OVCAR-3__sirolimus__100nM__24h	-6.353995

```r
# most up regulated genes
largest10_FC_all_max <- (sort(FC_all_max, decreasing = T)[1:10])
largest10_FC_all_max =as.data.frame(largest10_FC_all_max)
knitr::kable(largest10_FC_all_max, caption = "10 highest FC values")
```

Table 2: 10 highest FC values

	largest10_FC_all_max
T-47D__bortezomib__100nM__24h	9.493492
T-47D__vorinostat__5000nM__24h	9.162674
SNB-75__bortezomib__100nM__24h	9.078432
SK-MEL-28__vorinostat__5000nM__24h	8.906626
NCI-H226__doxorubicin__1000nM__24h	8.833774
SF-268__bortezomib__100nM__24h	8.823692
OVCAR-3__bortezomib__100nM__24h	8.681822
HS-578T__bortezomib__100nM__24h	8.604213
COLO-205__geldanamycin__1000nM__24h	8.578627
786-0__bortezomib__100nM__24h	8.529357

Here we can see, that vorinostat (occurs 3 times for downregulated and 2 times for up regulated), bortezomib (occurs 3 times for downregulated and 6 times for up regulated) seem to have great effects on the gene expression in the celllines. Moreover, the OVCAR-4 cell line, which bellows to the ovarian cancer, occurs the most (3 times) in the down regulated data.

---

1. PCA ANALYSIS

Description of this file:

The PCA analysis was performed for two matrices:

- Treated data
- Fold change data

For both PCAs, the celllines were colored by 2 different features:

- tissue-type
- drug-type

Treated data PCA

Execute the PCA for Treated data:

```
treated.pca = prcomp(Treated_norm, center=T, scale. = T)
```

Hereinafter, we want to use **information from Metadata** to color different celllines in the PCA. Therefore we need to check, if the celllines in the sample-column of Metadata are in the same order as in the Treated matrix. First of all we test, if the number of samples is equal.

Load Metadata:

```
library(readr)
  Metadata = read_tsv(paste0(wd,"/data/NCI_TPW_metadata.tsv"))
```

```
identical(nrow(Metadata), ncol(Treated_norm))
```

```
## [1] FALSE
```

```
nrow(Metadata)
```

```
## [1] 1638
```

```
ncol(Treated)
```

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

Metadata consists of twice as much celllines as the Treated matrix since Metadata contains information for treated and untreated celllines. We want to print those rows from Metadata which do not contain a zero concentration because they belong to the treated samples.

```
TreatedrowsMetadata <- grep(Metadata$sample, pattern = "_0nM_", invert = TRUE)
```

Check, if the sample order is equal in the Treated-matrix and in Metadata:

```
Metadata <- as.data.frame(Metadata)
Metadatasamples <- Metadata[TreatedrowsMetadata,"sample"]
all(colnames(Treated_norm)== Metadatasamples)
```

```
## [1] TRUE
```

Consequently the drug information of the Metadata-matrix can be assigned to the samples in the Treated-matrix sequentially. For better readability, we assign the column of interest to the name “Metadatadrugs”:

```
Metadatadrugs <- Metadata[TreatedrowsMetadata,"drug"]
```

Add Metadatadrugs as a new row to the Treated-matrix:

```
Treatedwithdrugs <- rbind(Treated_norm, Metadatadrugs)
```

Save drug information as factors so it can be used for coloring:

```
drugfactor <- as.factor(Treatedwithdrugs["Metadatadrugs",])
```

Now we can go on with coloring!

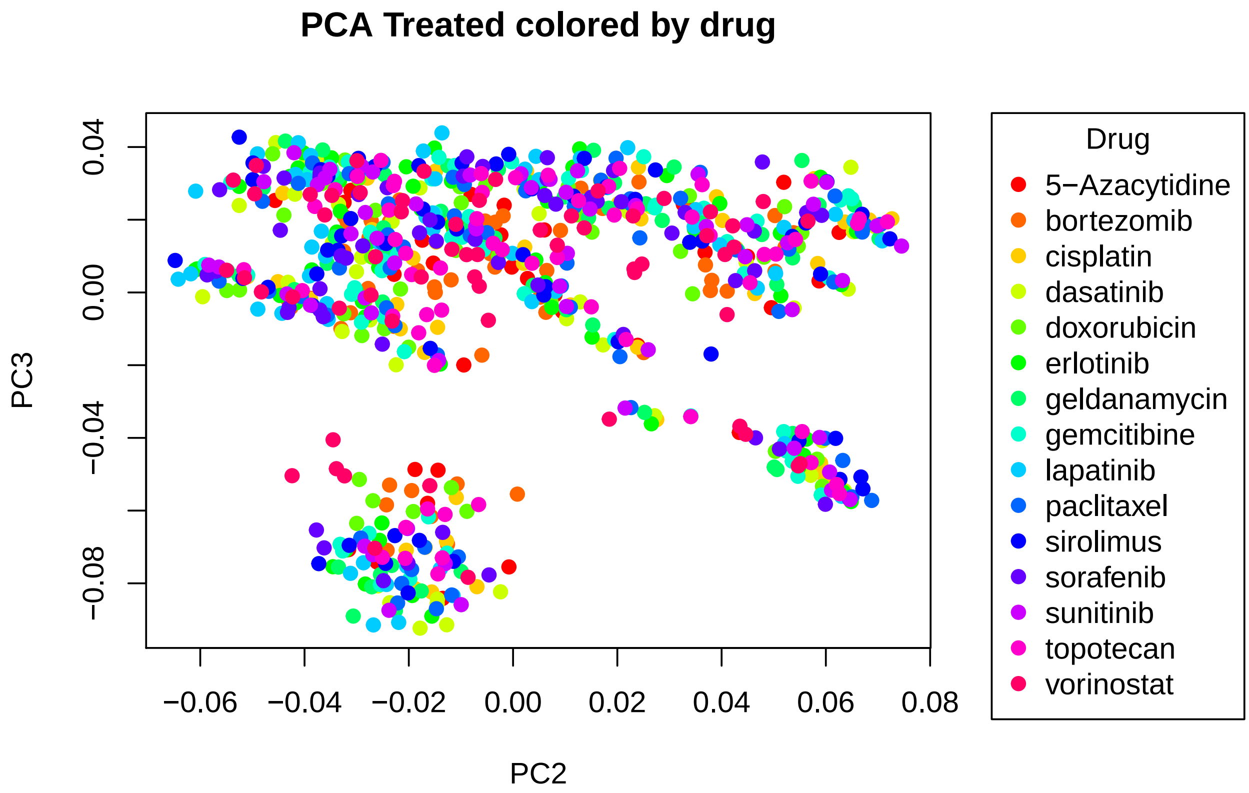
PLOT PCA & COLOR ACCORDING TO DRUGS:

Since we have 15 different drugs we need 15 different colors:

```
palette(rainbow(15))
```

Plot Principal component 1 and 2 and add a legend to the plot. To see the PCA plot and the legend next to each other, the graphical parameters are setted by the par() function.

```
par(mar=c(5, 4, 5, 9))
plot(treated.pca$rotation[, 2], treated.pca$rotation[, 3], pch = 19, xlab = "PC2",
     ylab = "PC3", col= drugfactor, main = "PCA Treated colored by drug")
druglevels <- as.factor(levels(drugfactor))
legend("topright", inset = c(-0.4,0),levels(drugfactor), xpd = TRUE, pch=19,
       col = druglevels, title = "Drug")
```



-> We do not see, that samples treated with the same drug form groups in the plot. However, we did not expect that, since we only look at the final expression and not at the expression change.

PLOT PCA & COLOR ACCORDING TO TISSUE:

The information which is needed for coloring is summarized as Metadatatissue:

```
Metadatatissue <- Metadata[TreatedrowsMetadata,"tissue"]
```

Bind Metadatatissue as a new row to the Treated matrix:

```
Treatedwithtissue <- rbind(Treated_norm, Metadatatissue)
```

Save tissue information as factors so it can be used for coloring:

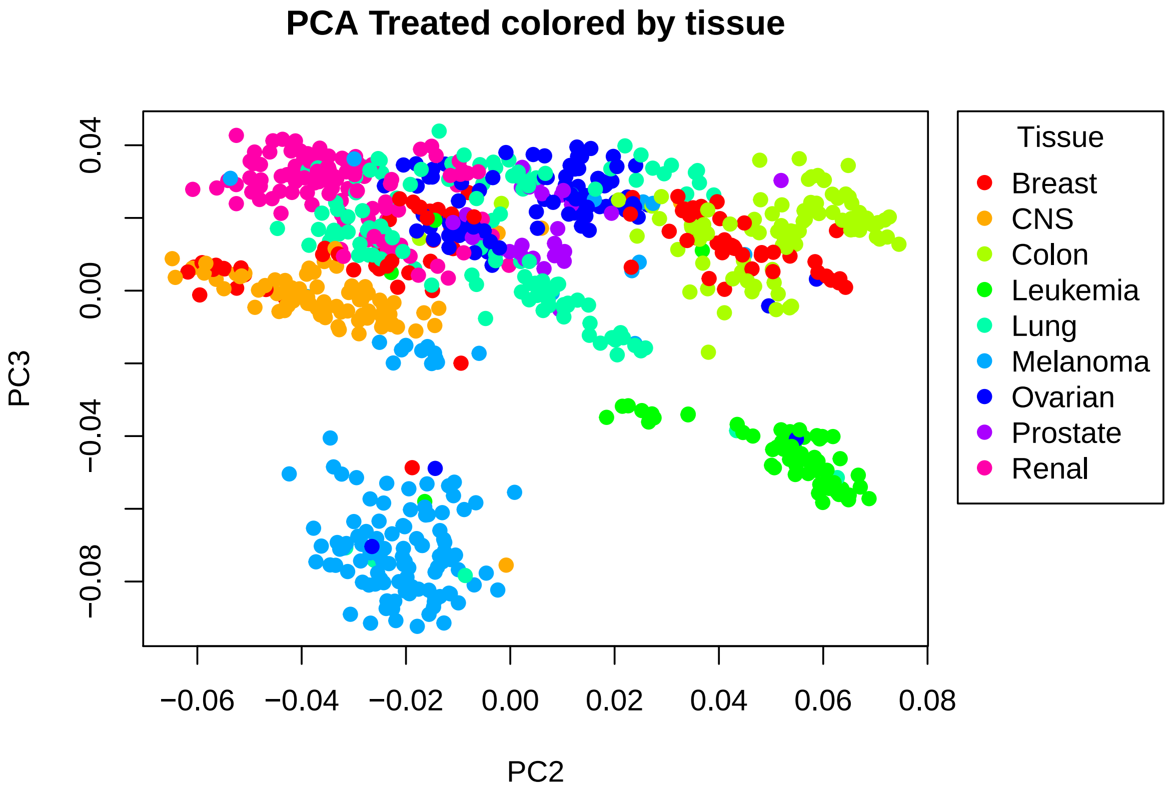
```
tissuefactor <- as.factor(Treatedwithtissue["Metadatatissue",])
```

Since we have 9 different tissue types we need 9 different colors:

```
palette(rainbow(9))
```

Plot PC 2 and PC 3 and add a legend:

```
par(mar=c(5, 4, 5, 9))
plot(treated.pca$rotation[, 2], treated.pca$rotation[, 3], pch = 19, xlab = "PC2",
     ylab = "PC3", col= tissuefactor, main= "PCA Treated colored by tissue")
levels <- as.factor(levels(tissuefactor))
tissuelevels <- as.factor(levels(tissuefactor))
legend("topright", inset = c(-0.3,0), levels(tissuefactor), xpd = TRUE, pch=19,
       col = tissuelevels, title = "Tissue")
```



-> PC 2 and PC 3 group the treated celllines as well as other PC combinations. Thus, most of the celllines of the same tissue-type seem to have similarities regarding their gene expression.

FC data PCA

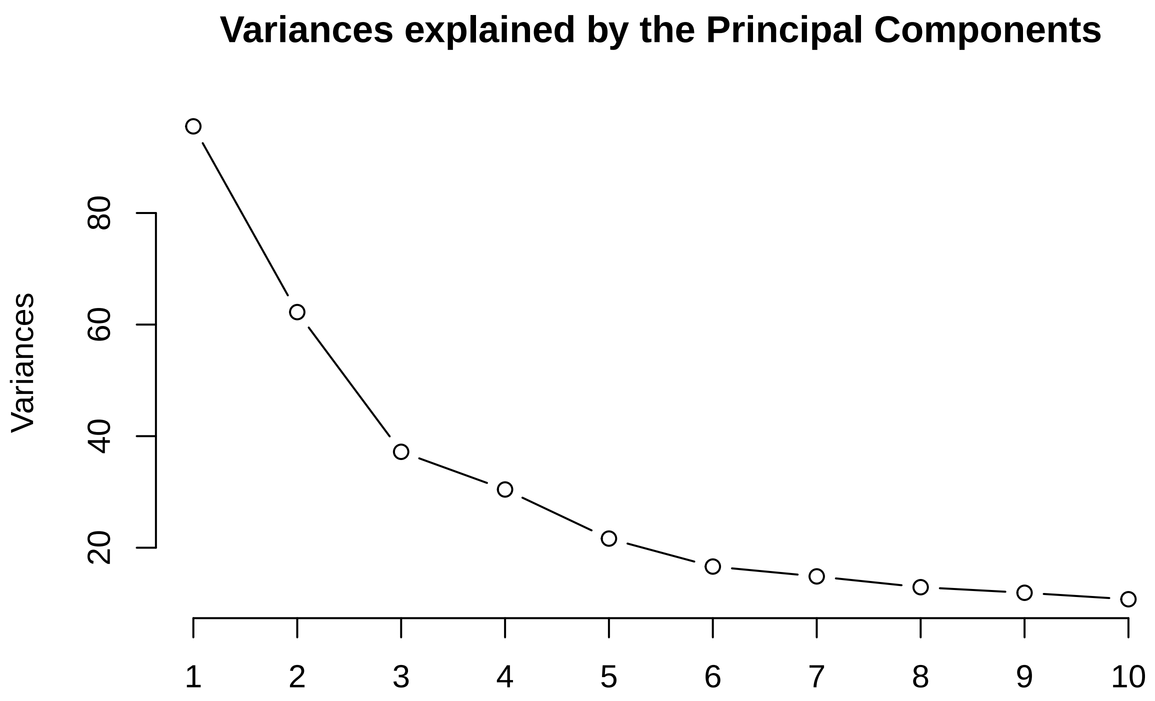
We execute the PCA with normalized FC data:

```
FC <- Treated - Untreated
FC_norm <- apply(FC, 2, function(x){
  (x - mean(x)) / sd(x)
})
```

```
pca.FC = prcomp(FC_norm, center = T, scale. = T)
```

We want to see how much variance is explained by each principle component:

```
plot(pca.FC, type = "l", main = "Variances explained by the Principal Components")
```



We can interpret, that **PC 1-3** explain most of the variance because an "elbow" can be seen after the third PC. Nevertheless, we should not exclude other PCs from our further analysis.

PLOT PCA & COLOR ACCORDING TO TISSUE

Bind the tissue-information as a new row to the FC matrix:

```
FCwithtissue <- rbind(FC_norm, Metadatatissue)
```

Save tissue information as factors so it can be used for coloring:

```
tissuefactorFC <- as.factor(FCwithtissue["Metadatatissue",])
```

9 different tissue types require 9 different colors:

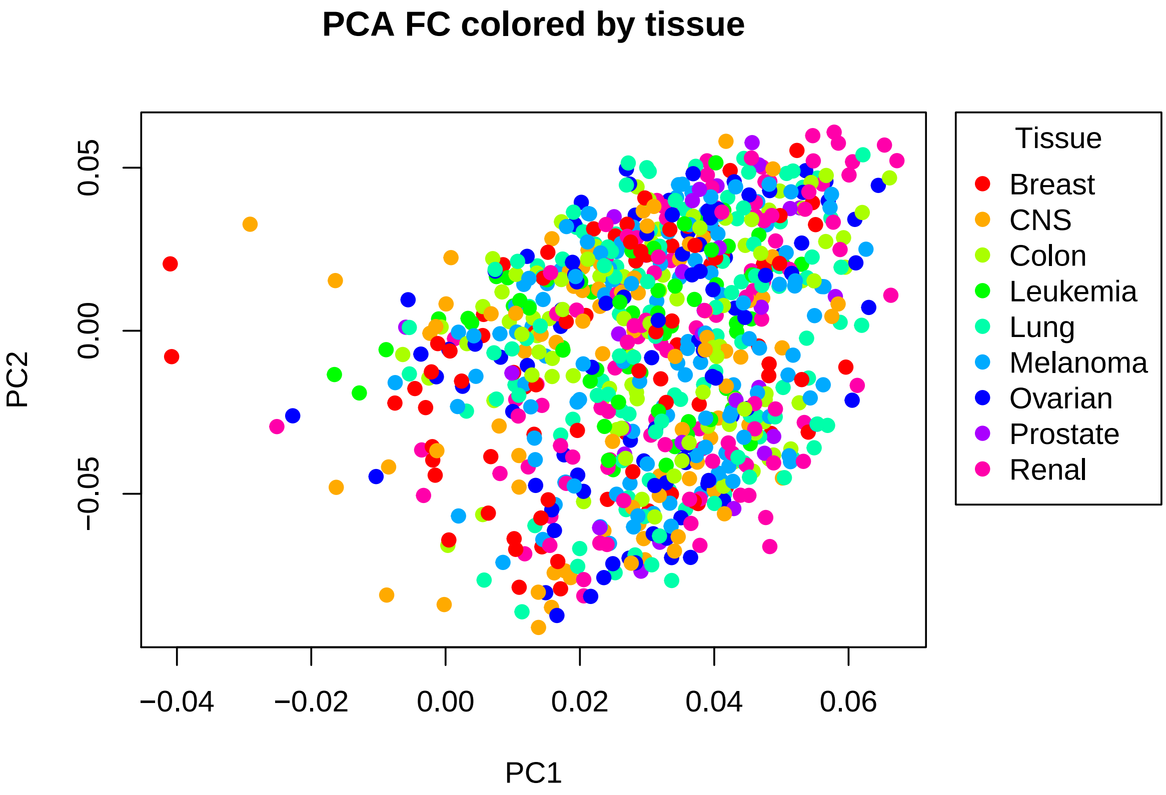
```
palette(rainbow(9))
```

Different PCs are plotted to see which combination groups the samples best. However, different tissues do not seem to group the points in any of the following PC combinations:

```
   PC1PC2 <- plot(pca.FC$rotation[, 1], pca.FC$rotation[, 2], col= tissuefactor, pch = 19,
                  xlab = "PC1", ylab = "PC2")
   PC2PC3 <- plot(pca.FC$rotation[, 2], pca.FC$rotation[, 3], col= tissuefactor, pch = 19,
                  xlab = "PC2", ylab = "PC3")
   PC3PC4 <- plot(pca.FC$rotation[, 3], pca.FC$rotation[, 4], col= tissuefactor, pch = 19,
                  xlab = "PC3", ylab = "PC4")
```

Example: PC1 and PC2 do not group celllines of same tissue-type:

```
par(mar=c(5, 4, 5, 9))
plot(pca.FC$rotation[, 1], pca.FC$rotation[, 2], col = tissuefactor, pch = 19, xlab = "PC1",
     ylab = "PC2", main = "PCA FC colored by tissue")
levels <- as.factor(levels(tissuefactorFC))
legend("topright", inset = c(-0.3,0), levels(tissuefactorFC), xpd = TRUE, pch=19,
       col = tissuelevels, title = "Tissue")
```



-> Since we are not able to identify groups of celllines of the same tissue, fold changes might be not very tissue-specific.

PLOT PCA & COLOR ACCORDING TO DRUGS

Create a new matrix ("FCwithdrugs") where the druginformation is added as a new row to the FC-matrix:

```
FCwithdrugs <- rbind(FC_norm, Metadatadrugs)
```

Save drug information as factors so it can be used for coloring:

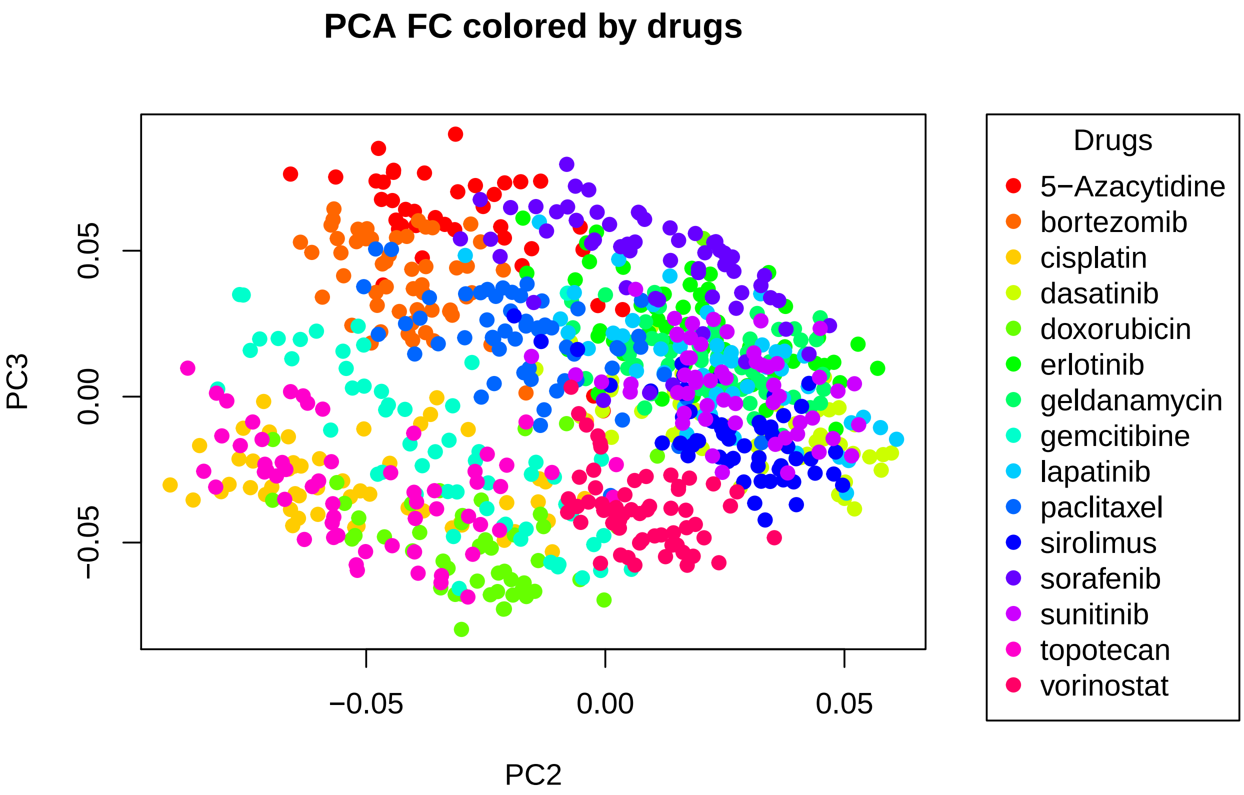
```
drugfactorFC <- as.factor(FCwithdrugs["Metadatadrugs",])
```

According to 15 different tissue types we need 15 different colors:

```
palette(rainbow(15))
```

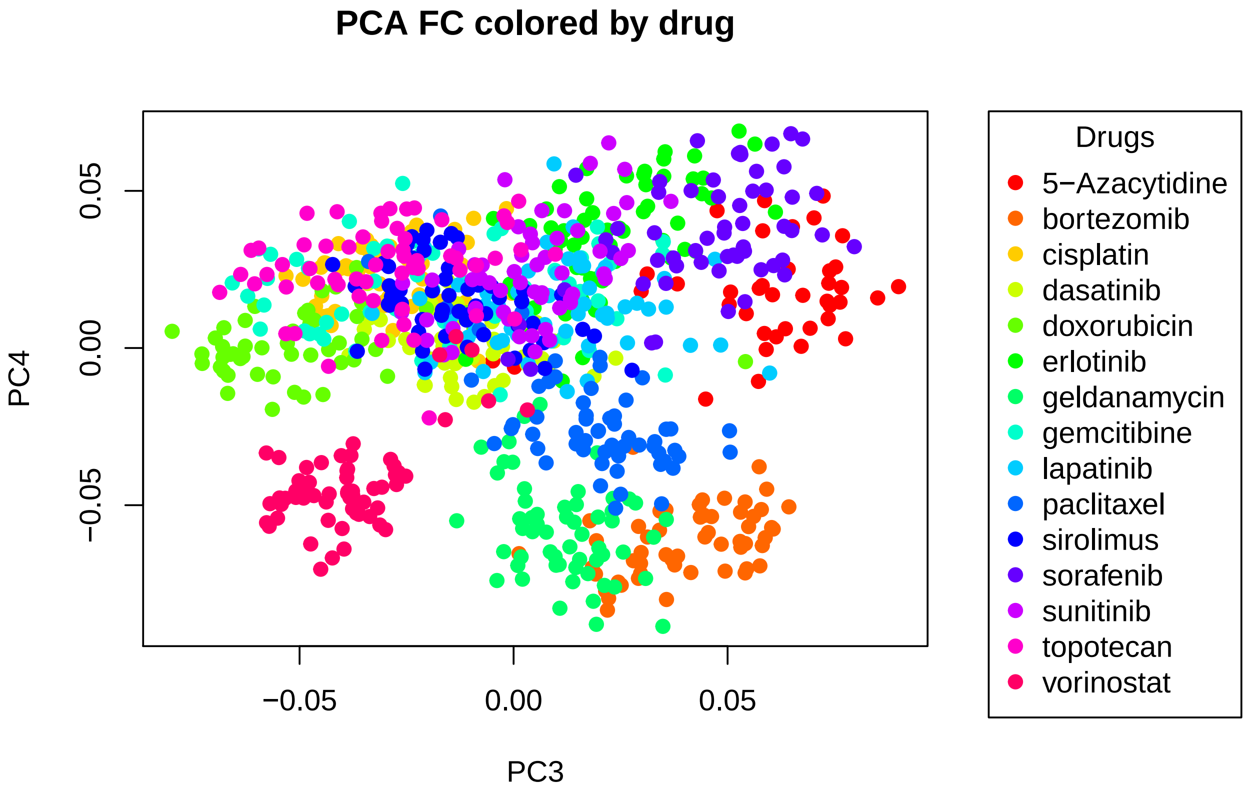
Plot PC 2 & PC 3:

```
par(mar=c(5, 4, 5, 9))
plot(pca.FC$rotation[, 2], pca.FC$rotation[, 3], col = drugfactor , pch = 19, xlab = "PC2"
         , ylab = "PC3", main = "PCA FC colored by drugs")
levels <- as.factor(levels(drugfactorFC))
legend("topright", inset = c(-0.4,0), levels(drugfactorFC), xpd = TRUE, pch=19,
       col = druglevels, title = "Drugs")
```



->Plot PC 3 and PC 4:

```
par(mar=c(5,4,5,9))
plot(pca.FC$rotation[, 3], pca.FC$rotation[, 4], col = drugfactor , pch = 19, xlab = "PC3",
     ylab = "PC4", main = "PCA FC colored by drug")
levels <- as.factor(levels(drugfactorFC))
legend("topright", inset = c(-0.4,0), levels(drugfactorFC), xpd = TRUE, pch=19,
       col = druglevels, title = "Drugs")
```



-> Many combinations of Principal Components clearly group celllines treated with the same drug. Consequently, the FC of celllines seems to be drug-specific. ***

HIGHLIGHT VORINOSTAT

Since we are going to analyze the effects of Vorinostat in our specific analysis we want to plot a PCA that highlights exclusively those celllines which belong to Vorinostattreatment.

Therefore we use the ifelse-function:

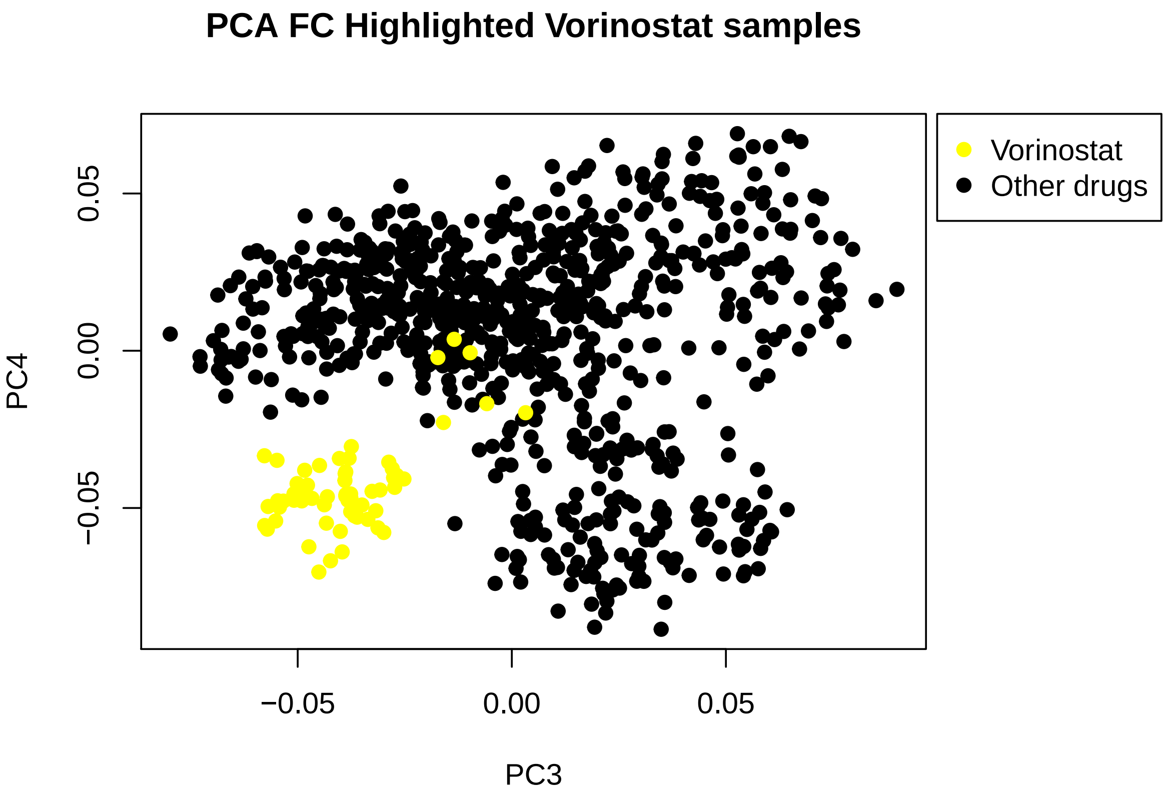
```
Metadata <-as.data.frame(Metadata)
Marking <- ifelse(Metadata$drug == "vorinostat", "yellow", "black")
```

Add the information, whether samples belong to Vorinostat, to the FC matrix:

```
HighlightVorinostat <- cbind(`FC_norm` = Marking)
```

Plot PC 3 and PC 4:

```
par(mar=c(5, 4, 5, 9))
plot(pca.FC$rotation[, 3], pca.FC$rotation[, 4], col = HighlightVorinostat, pch = 19,
     xlab = "PC3", ylab = "PC4", main = "PCA FC Highlighted Vorinostat samples")
legend("topright", inset = c(-0.3,0), legend = c("Vorinostat","Other drugs"),
       xpd = TRUE, pch=19, col = c("yellow", "black"))
```


PCA FC Highlighted Vorinostat samples
● Vorinostat
● Other drugs
PC4
0.05
0.00
−0.05
−0.05
0.00
0.05
PC3