Capstone project ~ Starting Data Visualization

Elena Tortosa

2/1/2019

Libraries and work directory

I loaded all required libraries and set the work directory.

```
library(gridExtra)
library(tidyr)
library(corrplot)
library(dplyr)
library(ggplot2)
library(gridExtra)
library(GGally)
library(d3heatmap)
library(gplots)
library(reshape2)
library(plotly)
setwd("/Users/Tortosae/Desktop/Data science course/Capstone_project")
require(knitr)
opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)
```

Data tables

All data was provided in 14 different excel sheets. I saved all as .csv and loaded them in R. They were named with a number (df#) followed by the name of the perturbation + treatment each table contains.

```
df1_cd3cd28 <- read.table(file = "1.cd3cd28.csv", sep = ",",
    header = TRUE)
df2_cd3cd28icam2 <- read.table(file = "2. cd3cd28icam2.csv",
    sep = ",", header = TRUE)
df3_cd3cd28aktinhib <- read.table(file = "3. cd3cd28aktinhib.csv",
    sep = ",", header = TRUE)
df4_cd3cd28g0076 <- read.table(file = "4. cd3cd28g0076.csv",
    sep = ",", header = TRUE)
df5_cd3cd28psitect <- read.table(file = "5. cd3cd28psitect.csv",
    sep = ",", header = TRUE)
df6_cd3cd28u0126 <- read.table(file = "6. cd3cd28u0126.csv",
    sep = ",", header = TRUE)
df7_cd3cd28ly <- read.table(file = "7. cd3cd28ly.csv", sep = ",",
    header = TRUE)
df8_pma <- read.table(file = "8. pma.csv", sep = ",", header = TRUE)
df9_b2camp <- read.table(file = "9. b2camp.csv", sep = ",", header = TRUE)
df10_cd3cd28icam2aktinhib <- read.table(file = "10. cd3cd28icam2aktinhib.csv",
    sep = ",", header = TRUE)
df11_cd3cd28icam2g0076 <- read.table(file = "11. cd3cd28icam2g0076.csv",
    sep = ",", header = TRUE)
df12_cd3cd28icam2psit <- read.table(file = "12. cd3cd28icam2psit.csv",
    sep = ",", header = TRUE)
```

```
df13_cd3cd28icam2u0126 <- read.table(file = "13. cd3cd28icam2u0126.csv",
    sep = ",", header = TRUE)
df14_cd3cd28icam2ly <- read.table(file = "14. cd3cd28icam2ly.csv",
    sep = ",", header = TRUE)
```

Column names

Only df8 contained two columns with different names (in lower case). I unfied column names.

```
df8_pma <- df8_pma %>% rename(PIP2 = pip2, PIP3 = pip3)
```

New column for perturbations

Measurements are obtained from two different perturbations ("general perturbation" : GP1 and GP2). At the same time, these perturbations are combined with different treatments (or treatment_nums) (see below: treatment_num column). I added a new column called GP to each table to classify the data depending on the general perturbation is applied : GP = 1 for GP1 and GP = 2 for GP2.

```
df1_cd3cd28 <- df1_cd3cd28 %>% mutate(treatment = "cd3cd28") %>%
    mutate(GP = 1)
df2_cd3cd28icam2 <- df2_cd3cd28icam2 %>% mutate(treatment = "cd3cd28icam2") %>%
    mutate(GP = 2)
df3_cd3cd28aktinhib <- df3_cd3cd28aktinhib %>% mutate(treatment = "cd3cd28aktinhib") %>%
    mutate(GP = 1)
df4_cd3cd28g0076 <- df4_cd3cd28g0076 %>% mutate(treatment = "cd3cd28g0076") %>%
    mutate(GP = 1)
df5_cd3cd28psitect <- df5_cd3cd28psitect %>% mutate(treatment = "cd3cd28psitect") %>%
    mutate(GP = 1)
df6_cd3cd28u0126 <- df6_cd3cd28u0126 %>% mutate(treatment = "cd3cd28u0126") %>%
    mutate(GP = 1)
df7_cd3cd28ly <- df7_cd3cd28ly %>% mutate(treatment = "cd3cd28ly") %>%
    mutate(GP = 1)
df8_pma <- df8_pma %>% mutate(treatment = "pma") %>% mutate(GP = 1)
df9_b2camp <- df9_b2camp %>% mutate(treatment = "b2camp") %>%
    mutate(GP = 1)
df10_cd3cd28icam2aktinhib <- df10_cd3cd28icam2aktinhib %>% mutate(treatment = "cd3cd28icam2aktinhib") %
    mutate(GP = 2)
df11_cd3cd28icam2g0076 <- df11_cd3cd28icam2g0076 %>% mutate(treatment = "cd3cd28icam2g0076") %>%
    mutate(GP = 2)
df12_cd3cd28icam2psit <- df12_cd3cd28icam2psit %>% mutate(treatment = "cd3cd28icam2psit") %>%
    mutate(GP = 2)
df13_cd3cd28icam2u0126 <- df13_cd3cd28icam2u0126 %>% mutate(treatment = "cd3cd28icam2u0126") %>%
    mutate(GP = 2)
df14_cd3cd28icam2ly <- df14_cd3cd28icam2ly %>% mutate(treatment = "cd3cd28icam2ly") %>%
    mutate(GP = 2)
```

New column for treatments (treatment_num)

As mentioned before, measurments are obtained from two different perturbations (GP1 and GP2). At the same time, these perturbations are combined with different treatments. I added a new column called “treatment_num” to each table to classify the data depending on the treatment applied : 0 <- no treatment 4 <- MEK_inh 1 <- Akt_inh1 5 <- Akt_inh2 2 <- PKC_inh 6 <- PKC_act 3 <- PIP2_inh 7 <- PKA_act

```
# Add new columns with treatment_nums names
df1_cd3cd28 <- df1_cd3cd28 %>% mutate(treatment = "cd3cd28") %>%
    mutate(treatment_num = 0)
df2_cd3cd28icam2 <- df2_cd3cd28icam2 %>% mutate(treatment = "cd3cd28icam2") %>%
    mutate(treatment_num = 0)
df3_cd3cd28aktinhib <- df3_cd3cd28aktinhib %>% mutate(treatment = "cd3cd28aktinhib") %>%
    mutate(treatment_num = 1)
df4_cd3cd28g0076 <- df4_cd3cd28g0076 %>% mutate(treatment = "cd3cd28g0076") %>%
    mutate(treatment_num = 2)
df5_cd3cd28psitect <- df5_cd3cd28psitect %>% mutate(treatment = "cd3cd28psitect") %>%
    mutate(treatment_num = 3)
df6_cd3cd28u0126 <- df6_cd3cd28u0126 %>% mutate(treatment = "cd3cd28u0126") %>%
    mutate(treatment_num = 4)
df7_cd3cd28ly <- df7_cd3cd28ly %>% mutate(treatment = "cd3cd28ly") %>%
    mutate(treatment_num = 5)
df8_pma <- df8_pma %>% mutate(treatment = "pma") %>% mutate(treatment_num = 6)
df9_b2camp <- df9_b2camp %>% mutate(treatment = "b2camp") %>%
    mutate(treatment_num = 7)
df10_cd3cd28icam2aktinhib <- df10_cd3cd28icam2aktinhib %>% mutate(treatment = "cd3cd28icam2aktinhib") %
    mutate(treatment_num = 1)
df11_cd3cd28icam2g0076 <- df11_cd3cd28icam2g0076 %>% mutate(treatment = "cd3cd28icam2g0076") %>%
    mutate(treatment_num = 2)
df12_cd3cd28icam2psit <- df12_cd3cd28icam2psit %>% mutate(treatment = "cd3cd28icam2psit") %>%
    mutate(treatment_num = 3)
df13_cd3cd28icam2u0126 <- df13_cd3cd28icam2u0126 %>% mutate(treatment = "cd3cd28icam2u0126") %>%
    mutate(treatment_num = 4)
df14_cd3cd28icam2ly <- df14_cd3cd28icam2ly %>% mutate(treatment = "cd3cd28icam2ly") %>%
    mutate(treatment_num = 5)
```

Unique table

I created a unique table for all the perturbations/treatments.

```
alldf <- bind_rows(df1_cd3cd28, df2_cd3cd28icam2, df3_cd3cd28aktinhib,
    df4_cd3cd28g0076, df5_cd3cd28psitect, df6_cd3cd28u0126, df7_cd3cd28ly,
    df8_pma, df9_b2camp, df10_cd3cd28icam2aktinhib, df11_cd3cd28icam2g0076,
    df12_cd3cd28icam2psit, df13_cd3cd28icam2u0126, df14_cd3cd28icam2ly)
```

Reorder columns

I reorder the columns to have the treatment names and dummy variables first, and after that all the measurments done.

```
alldf <- alldf %>% select(treatment, GP, treatment_num, everything())
```

Table visualization

```
head(alldf)
```

```
##   treatment GP treatment_num praf  pmek  plcg  PIP2  PIP3 p44.42 pakts473
## 1   cd3cd28  1             0 26.4 13.20  8.82 18.30 58.80   6.61     17.0
## 2   cd3cd28  1             0 35.9 16.50 12.30 16.80  8.13  18.60     32.5
## 3   cd3cd28  1             0 59.4 44.10 14.60 10.20 13.00  14.90     32.5
## 4   cd3cd28  1             0 73.0 82.80 23.10 13.50  1.29   5.83     11.8
## 5   cd3cd28  1             0 33.7 19.80  5.19  9.73 24.80  21.10     46.1
## 6   cd3cd28  1             0 18.8  3.75 17.60 22.10 10.90  11.90     25.7
##   PKA   PKC  P38 pjnk
## 1 414 17.00 44.9 40.0
## 2 352  3.37 16.5 61.5
## 3 403 11.40 31.9 19.5
## 4 528 13.70 28.6 23.1
## 5 305  4.66 25.7 81.3
## 6 610 13.70 49.1 57.8
```

Data subseting

I grouped the data to help in the data visualization

```
GP1 <- subset(alldf, GP == "1", select = c("treatment", "treatment_num",
    "praf", "pmek", "plcg", "PIP2", "PIP3", "p44.42", "pakts473",
    "PKA", "PKC", "P38", "pjnk"))
GP2 <- subset(alldf, GP == "2", select = c("treatment", "treatment_num",
    "praf", "pmek", "plcg", "PIP2", "PIP3", "p44.42", "pakts473",
    "PKA", "PKC", "P38", "pjnk"))
```

Summarise the data to see overall trends

```
stats_GP1 <- GP1 %>% group_by(treatment_num) %>% summarise_at(vars(praf:pjnk),
    mean, na.rm = TRUE)
head(stats_GP1)
```

```
## # A tibble: 6 x 12
##   treatment_num  praf  pmek   plcg   PIP2  PIP3 p44.42 pakts473   PKA
##           <dbl> <dbl> <dbl>  <dbl>  <dbl> <dbl>  <dbl>    <dbl> <dbl>
## 1             0  59.3  30.0  19.5    81.6  30.5  22.2      42.0 567.
## 2             1  57.1  29.7  15.1   167.    45.5  23.6      47.7 712.
## 3             2 412.  639.  392.   690.    18.6  59.7     374.   16.0
## 4             3  60.1  31.9   5.82   6.97  15.0  27.1      58.9 657.
## 5             4 390.  572.   18.3   75.2  30.3   6.06      70.2 587.
## 6             5  57.7  29.9  13.6    79.3  27.5  15.9      31.3 633.
## # ... with 3 more variables: PKC <dbl>, P38 <dbl>, pjnk <dbl>
```

```
stats_GP2 <- GP2 %>% group_by(treatment_num) %>% summarise_at(vars(praf:pjnk),
    mean, na.rm = TRUE)
head(stats_GP2)
```

```
## # A tibble: 6 x 12
##   treatment_num  praf  pmek  plcg  PIP2  PIP3 p44.42 pakts473   PKA   PKC
##           <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>    <dbl> <dbl> <dbl>
## 1             0  64.8  43.8  27.0 117.    21.1   24.9     46.7 885.  14.9
## 2             1  62.7  48.8  11.7 138.    31.7   35.9     67.1 825.   9.61
## 3             2 393.  576.  263.  491.    28.6  107.     559.  599.  39.6
## 4             3  57.2  44.9  11.9  98.8  16.7   20.6     57.4  772. 21.0
## 5             4  53.7  43.3  17.6 111.   27.3   28.5     63.0 1077. 18.1
## 6             5  49.2  36.5  12.3  86.4  21.6   38.6     74.9 1239. 17.5
## # ... with 2 more variables: P38 <dbl>, pjnk <dbl>
```

Data normalization

```
GP1.n <- stats_GP1 %>% select(-treatment_num)
GP1.n2 <- scale(GP1.n)
GP1.n2t <- t(GP1.n2)
GP2.n <- stats_GP2 %>% select(-treatment_num)
GP2.n2 <- scale(GP2.n)
```

I check mean and sd of normalized data

```
round(colMeans(GP1.n2), 1)
```

```
##     praf     pmek     plcg     PIP2     PIP3   p44.42 pakts473      PKA
##        0        0        0        0        0        0        0        0
##      PKC      P38     pjnk
##        0        0        0
```

```
apply(GP1.n2, 2, sd)
```

```
##     praf     pmek     plcg     PIP2     PIP3   p44.42 pakts473      PKA
##        1        1        1        1        1        1        1        1
##      PKC      P38     pjnk
##        1        1        1
```

```
round(colMeans(GP2.n2), 1)
```

```
##     praf     pmek     plcg     PIP2     PIP3   p44.42 pakts473      PKA
##        0        0        0        0        0        0        0        0
##      PKC      P38     pjnk
##        0        0        0
```

```
apply(GP2.n2, 2, sd)
```

```
##     praf     pmek     plcg     PIP2     PIP3   p44.42 pakts473      PKA
##        1        1        1        1        1        1        1        1
##      PKC      P38     pjnk
##        1        1        1
```

Data visualization

Means visualization

Lineplots

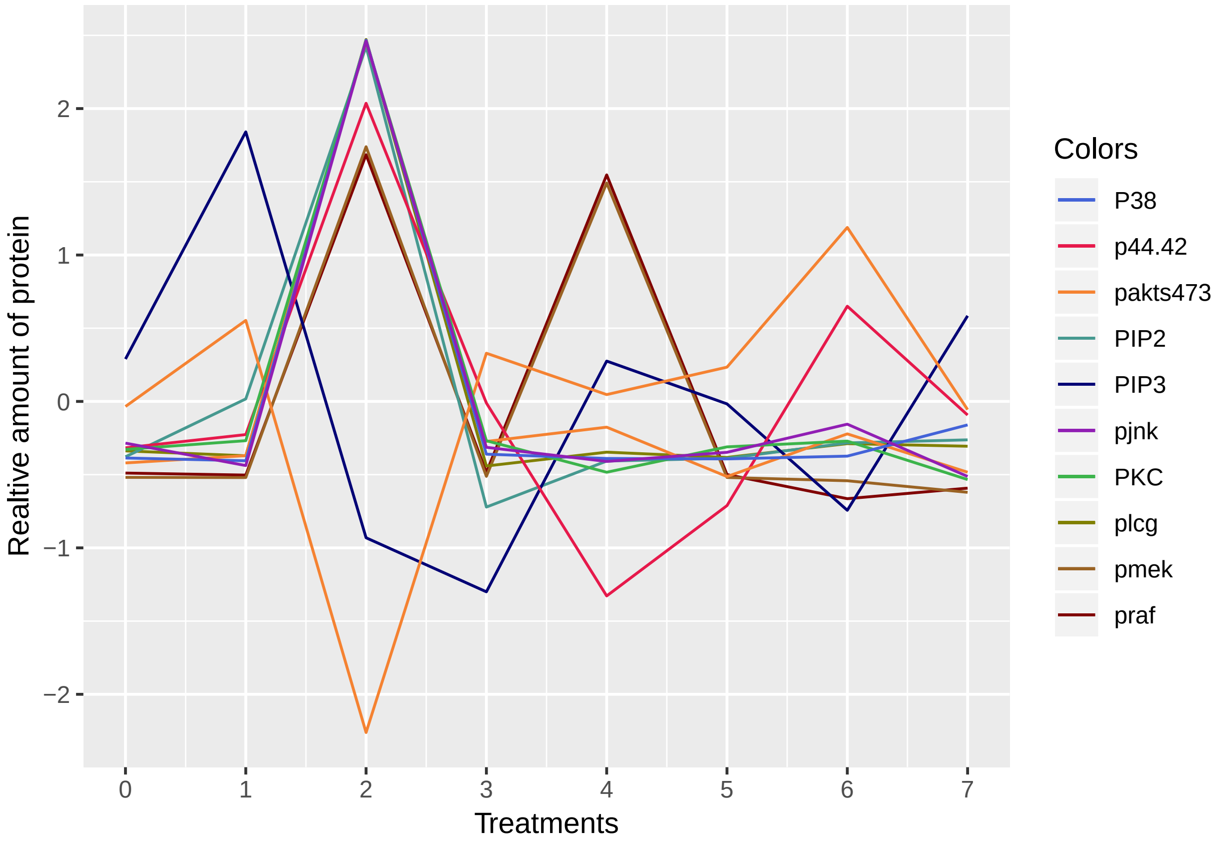
I ploted the means for each protein in each condition using a line plot.

Option#1

```
stats_GP1.n <- as.data.frame(scale(stats_GP1))
head(stats_GP1.n)
```

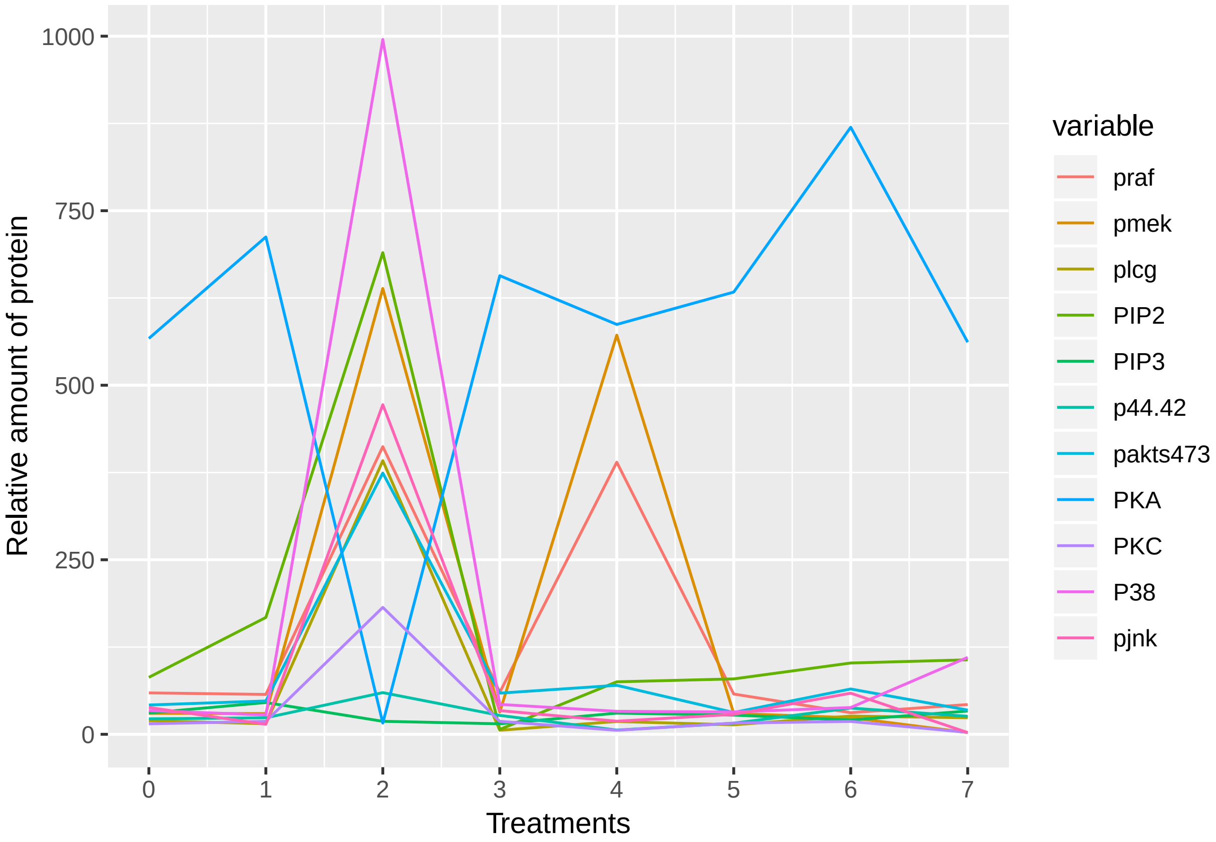
```
##   treatment_num       praf       pmek       plcg        PIP2        PIP3
## 1    -1.4288690 -0.4892117 -0.5185249 -0.3377290 -0.37794909  0.28969704
## 2    -1.0206207 -0.5029176 -0.5197683 -0.3708978  0.01718315  1.84085385
## 3    -0.6123724  1.6848022  1.7395333  2.4721112  2.42301762 -0.93085115
## 4    -0.2041241 -0.4842163 -0.5115113 -0.4408820 -0.72132722 -1.30000677
## 5     0.2041241  1.5471602  1.4917264 -0.3463778 -0.40721973  0.27552765
## 6     0.6123724 -0.4994162 -0.5191839 -0.3823181 -0.38818237 -0.01660319
##        p44.42   pakts473         PKA        PKC        P38       pjnk
## 1 -0.31687370 -0.4199101 -0.03433722 -0.3275042 -0.3861926 -0.2847274
## 2 -0.22649398 -0.3702036  0.55325808 -0.2679058 -0.4052076 -0.4378697
## 3  2.03637970  2.4567208 -2.26110318  2.4618970  2.4670203  2.4605050
## 4 -0.01080528 -0.2731784  0.32915740 -0.2683004 -0.3601210 -0.3133988
## 5 -1.32808608 -0.1757586  0.04679425 -0.4836028 -0.3895346 -0.4096616
## 6 -0.71142178 -0.5128200  0.23412288 -0.3103291 -0.3923804 -0.3474121
```

```
lab <- c("0", "1", "2", "3", "4", "5", "6", "7")
ggplot(stats_GP1.n) + geom_line(aes(x = c(0:7), y = praf, colour = "praf")) +
    geom_line(aes(c(0:7), y = pmek, colour = "pmek")) + geom_line(aes(c(0:7),
    y = plcg, colour = "plcg")) + geom_line(aes(c(0:7), y = PIP2,
    colour = "PIP2")) + geom_line(aes(c(0:7), y = PIP3, colour = "PIP3")) +
    geom_line(aes(c(0:7), y = p44.42, colour = "p44.42")) + geom_line(aes(c(0:7),
    y = pakts473, colour = "pakts473")) + geom_line(aes(c(0:7),
    y = PKA, colour = "pakts473")) + geom_line(aes(c(0:7), y = PKC,
    colour = "PKC")) + geom_line(aes(c(0:7), y = P38, colour = "P38")) +
    geom_line(aes(c(0:7), y = pjnk, colour = "pjnk")) + scale_x_continuous(name = "Treatments",
    breaks = c(0:7), labels = c(0:7)) + ylab("Realtive amount of protein") +
    xlab("Treatments") + scale_color_manual(name = "Colors",
    values = c(praf = "#800000", pmek = "#9A6324", plcg = "#808000",
        PIP2 = "#469990", PIP3 = "#000075", p44.42 = "#e6194B",
        pakts473 = "#f58231", PKA = "#ffe119", PKC = "#3cb44b",
        P38 = "#4363d8", pjnk = "#911eb4"))
```



Option#2

```
test_data_long <- melt(stats_GP1, id = "treatment_num")  # convert to long format
ggplot(data = test_data_long, aes(x = treatment_num, y = value,
    colour = variable)) + geom_line() + scale_x_continuous(name = "Treatments",
    breaks = c(0:7), labels = c(0:7)) + scale_y_continuous(name = "Relative amount of protein")
```



Protein and treatment clustering

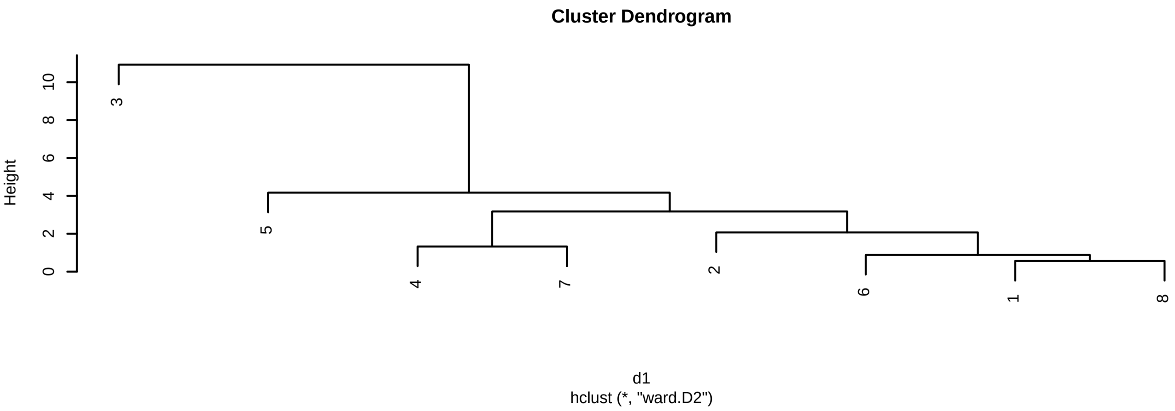
I clustered the proteins and the treatments based on their mean values.

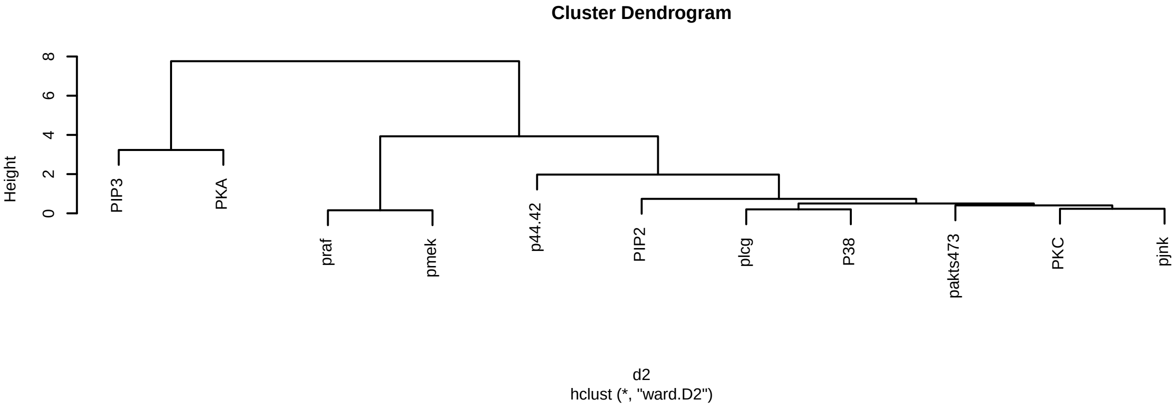
1. Calculate distance between experiments and protein in rows and cluster the data based on these distances.

```
d1 <- dist(GP1.n2,method = "euclidean", diag = FALSE, upper = FALSE)
round(d1,3)
```

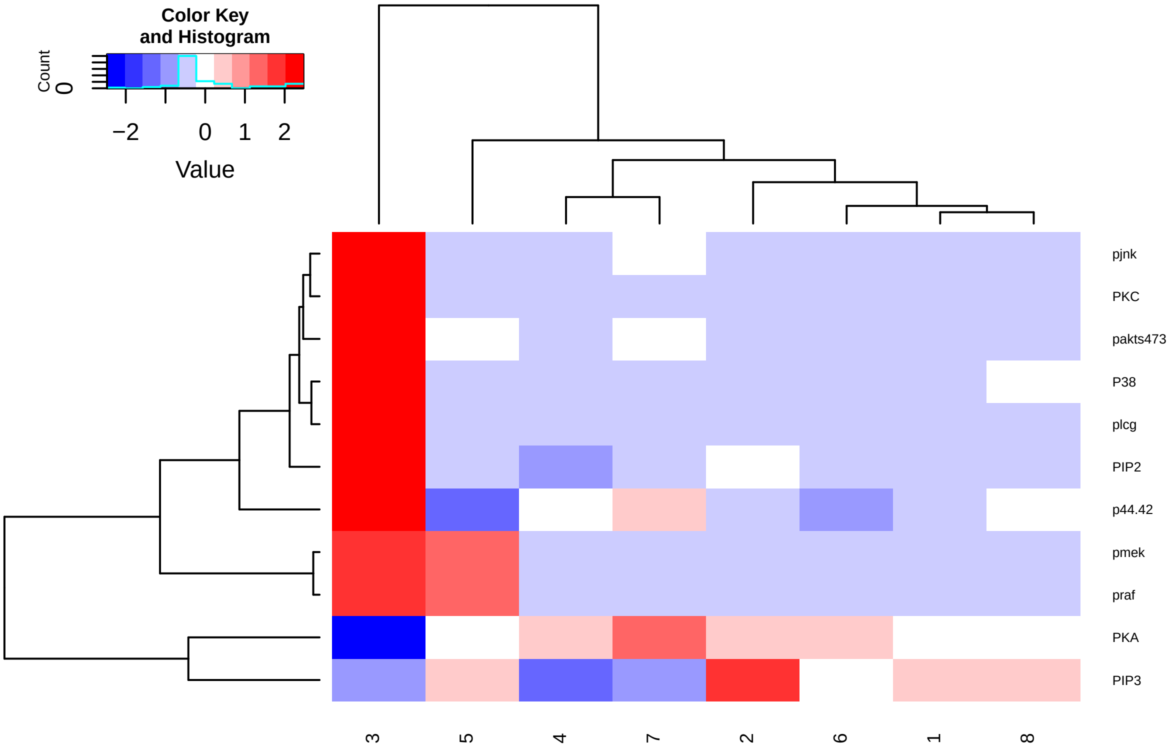
```
##       1     2     3     4     5     6     7
## 2 1.717
## 3 8.324 8.751
## 4 1.705 3.246 8.353
## 5 3.053 3.526 8.133 3.551
## 6 0.580 1.996 8.550 1.524 2.980
## 7 1.897 2.842 8.311 1.327 3.926 1.861
## 8 0.568 1.489 8.410 2.038 3.294 1.000 2.046
```

```
d2 <- dist(GP1.n2t,method = "euclidean", diag = FALSE, upper = TRUE)
# Clustering distance between experiments using Ward linkage
c1 <- hclust(d1, method = "ward.D2", members = NULL)
# Clustering distance between proteins using Ward linkage
c2 <- hclust(d2, method = "ward.D2", members = NULL)
# Check clustering by plotting dendrograms
par(mfrow=c(2,1),cex=0.5) # Make 2 rows, 1 col plot frame and shrink labels
plot(c1); plot(c2) # Plot both cluster dendrograms
```





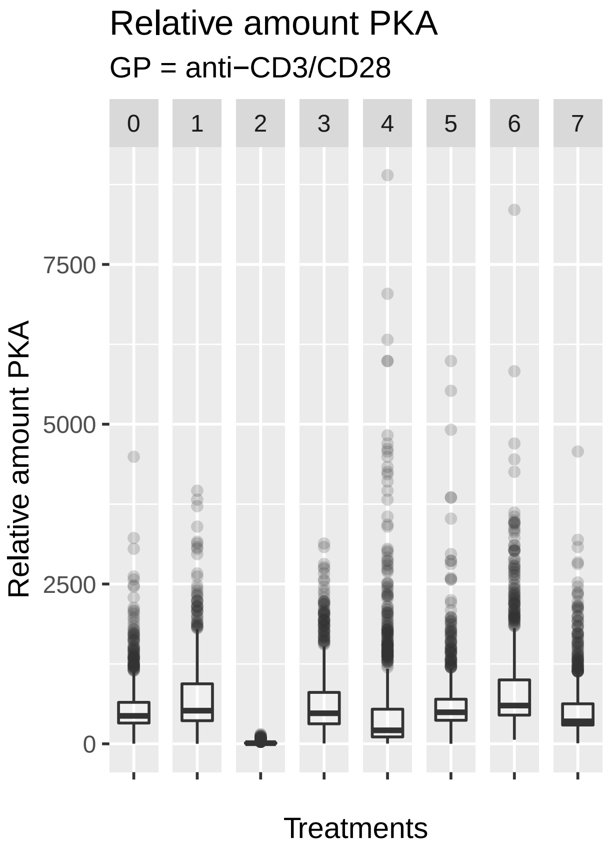
```
GP1.m <- as.matrix (GP1.n2t)
# Set colours for heatmap, 25 increments
my_palette <- colorRampPalette(c("blue","white","red"))(n = 11)

# Plot heatmap with heatmap.2
par(cex.main=0.75) # Shrink title fonts on plot
heatmap.2(GP1.m,                     # Tidy, normalised data
          Colv=as.dendrogram(c1),     # Experiments clusters in cols
          Rowv=as.dendrogram(c2),     # Protein clusters in rows
          density.info="histogram",   # Plot histogram of data and colour key
          trace="none",               # Turn of trace lines from heat map
          col = my_palette,           # Use my colour scheme
          cexRow=0.5,cexCol=0.75)     # Amend row and column label fonts
```

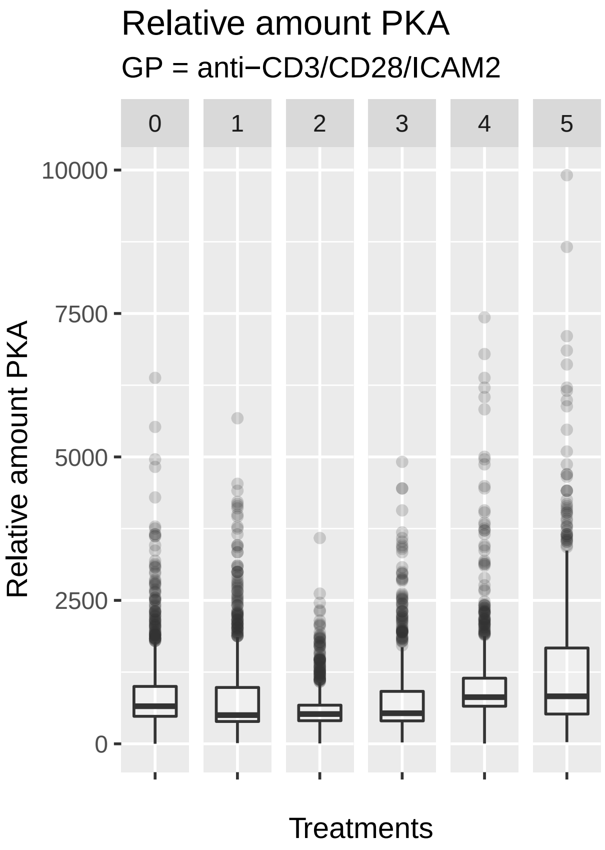


Box-plots per protein (PKA, PKC, p38, JNK,. . .)

I represented a graph for each perturbation (GP1 and GP2) and for each single treatment (in this case only for PKA). Here, I put just few examples. The idea would be to do similar graphs for the different variables.

```
PKA_GP1 <- ggplot(GP1, aes(x = "", y = PKA)) + geom_boxplot(aes(),
    alpha = 0.2) + facet_grid(. ~ treatment_num) + labs(title = "Relative amount PKA",
    subtitle = "GP = anti-CD3/CD28", x = "Treatments", y = "Relative amount PKA")

PKA_GP2 <- ggplot(GP2, aes(x = "", y = PKA)) + geom_boxplot(aes(),
    alpha = 0.2) + facet_grid(. ~ treatment_num) + labs(title = "Relative amount PKA",
    subtitle = "GP = anti-CD3/CD28/ICAM2", x = "Treatments",
    y = "Relative amount PKA")

grid.arrange(PKA_GP1, PKA_GP2, nrow = 1)
```

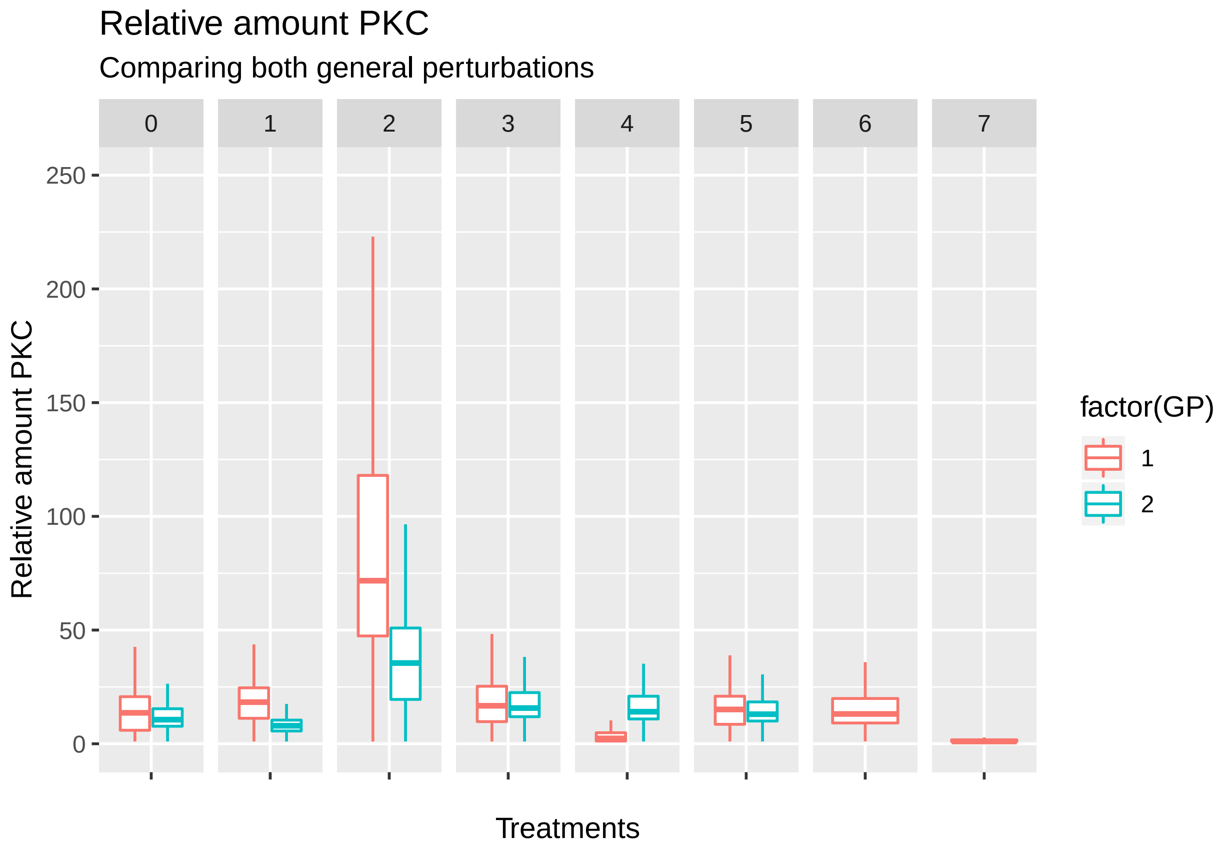




Comparing perturbations/treatment_nums for PKC

Why are values in X different for depending where I put PKC (X or y)?

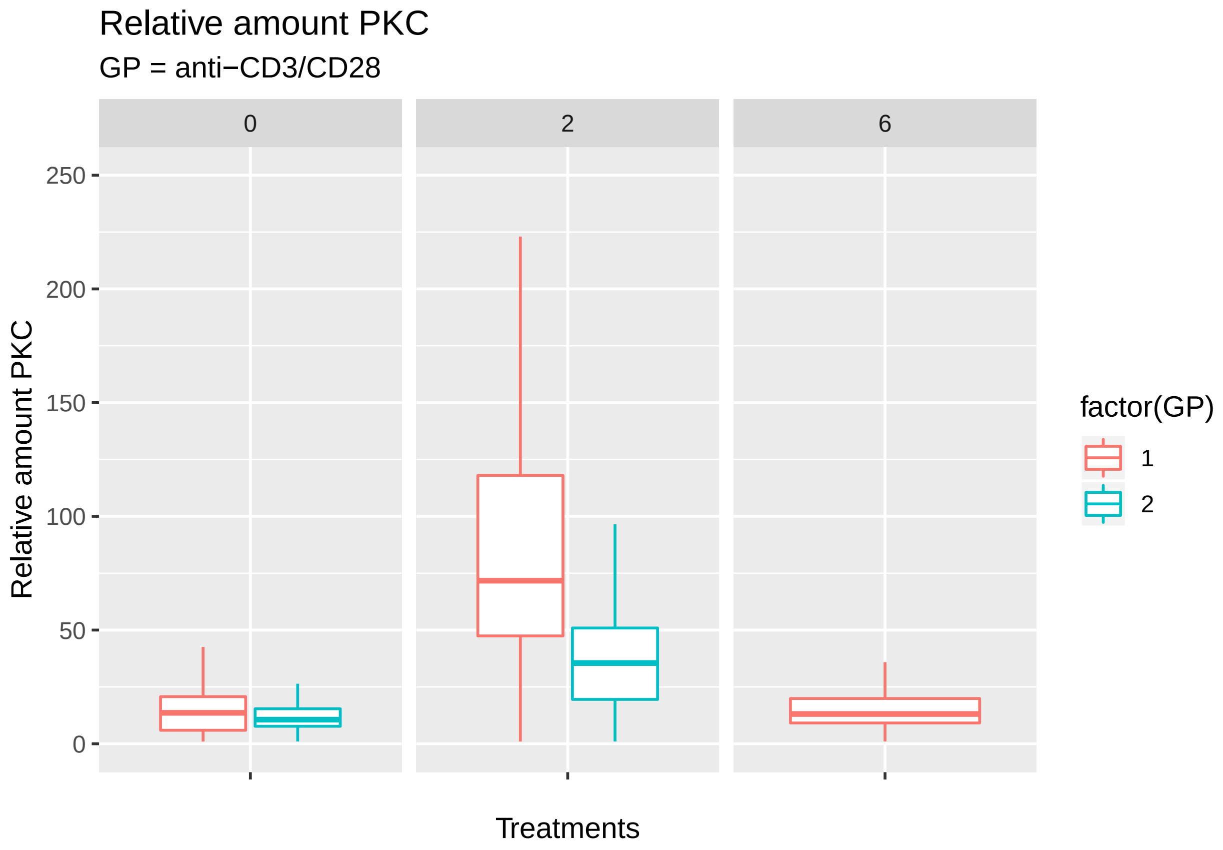
```
ggplot(alldf, aes(x = "", y = PKC, color = factor(GP))) + geom_boxplot(outlier.shape = NA) +
    ylim(0, 250) + facet_grid(. ~ treatment_num) + labs(title = "Relative amount PKC",
    subtitle = "Comparing both general perturbations", x = "Treatments",
    y = "Relative amount PKC")
```



Checking activation vs inhibition

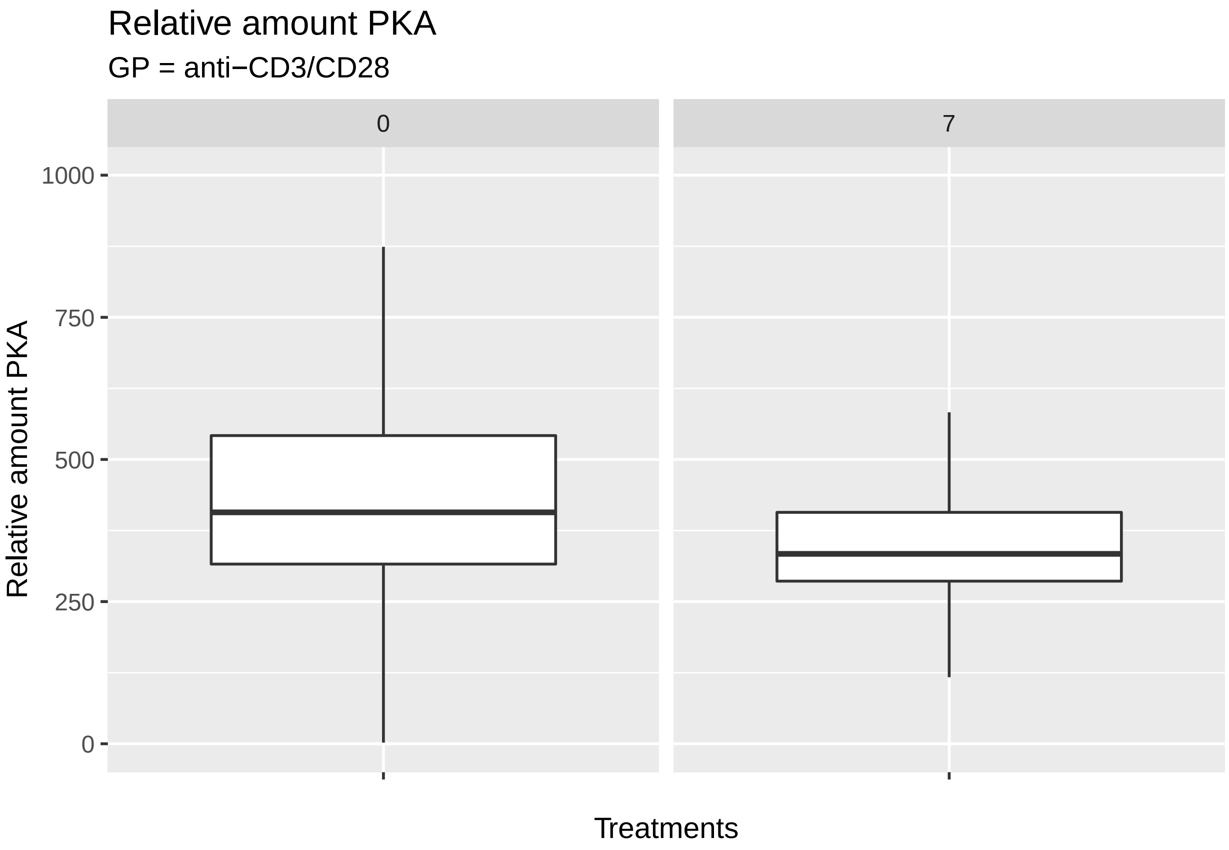
PKC control (cond = 0), activated (cond = 6) and inhibited (cond = 2)

```
PKC_actvsinh <- alldf %>% filter(treatment_num %in% c("0", "2",
    "6"))
ggplot(PKC_actvsinh, aes(x = "", y = PKC, color = factor(GP))) +
    geom_boxplot(outlier.shape = NA) + ylim(0, 250) + facet_grid(. ~
    treatment_num) + labs(title = "Relative amount PKC", subtitle = "GP = anti-CD3/CD28",
    x = "Treatments", y = "Relative amount PKC")
```



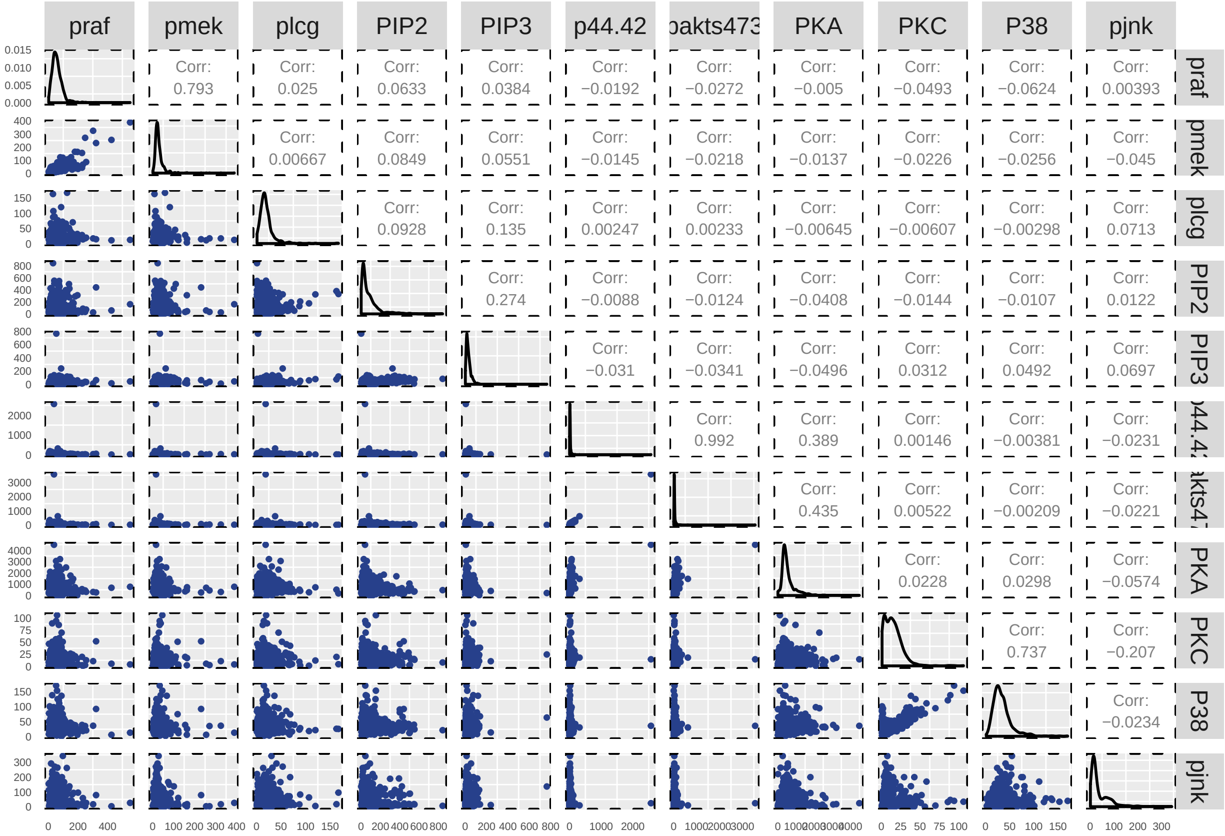
PKA control (cond = 0) vs activated (cond = 7)

```
PKA_act <- GP1 %>% filter(treatment_num %in% c("0", "7"))
ggplot(PKA_act, aes(x = "", y = PKA)) + geom_boxplot(outlier.shape = NA) +
    ylim(0, 1000) + facet_grid(. ~ treatment_num) + labs(title = "Relative amount PKA",
    subtitle = "GP = anti-CD3/CD28", x = "Treatments", y = "Relative amount PKA")
```



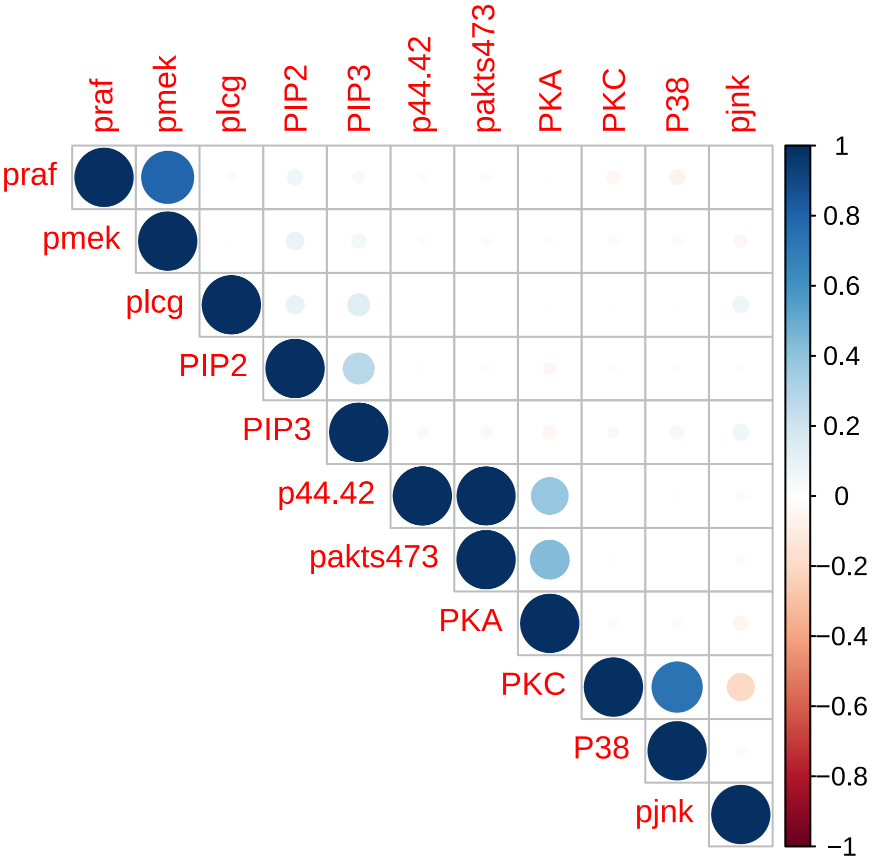
Looking at some correlations

```
CorrPlot <- GP1 %>% filter(treatment_num == "0")
ggpairs(CorrPlot, columns = 3:ncol(CorrPlot), upper = list(continuous = wrap("cor",
    size = 2)), lower = list(continuous = wrap("points", size = 0.5,
    color = "royalblue4"))) + theme(legend.position = "none",
    panel.grid.major = element_blank(), axis.text = element_text(size = 4),
    axis.ticks = element_blank(), panel.border = element_rect(linetype = "dashed",
        colour = "black", fill = NA))
```

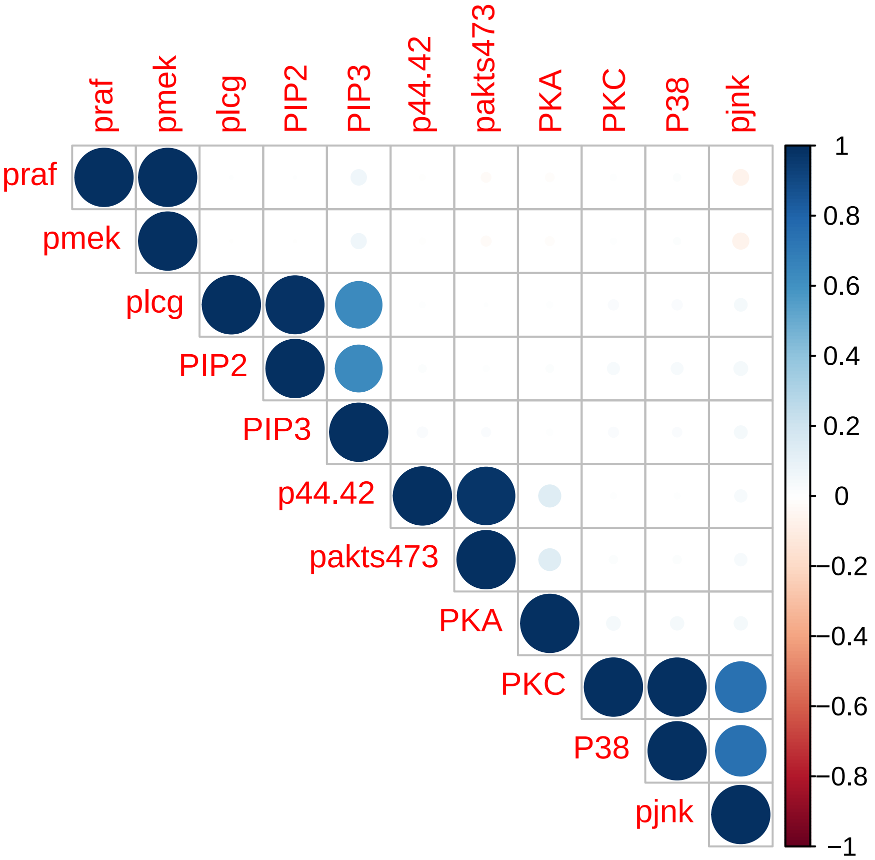


Option2

```
CM_0 <- GP1 %>% filter(treatment_num == "0") %>% select(-treatment,
    -treatment_num) %>% cor()
corrplot(CM_0, type = "upper")
```



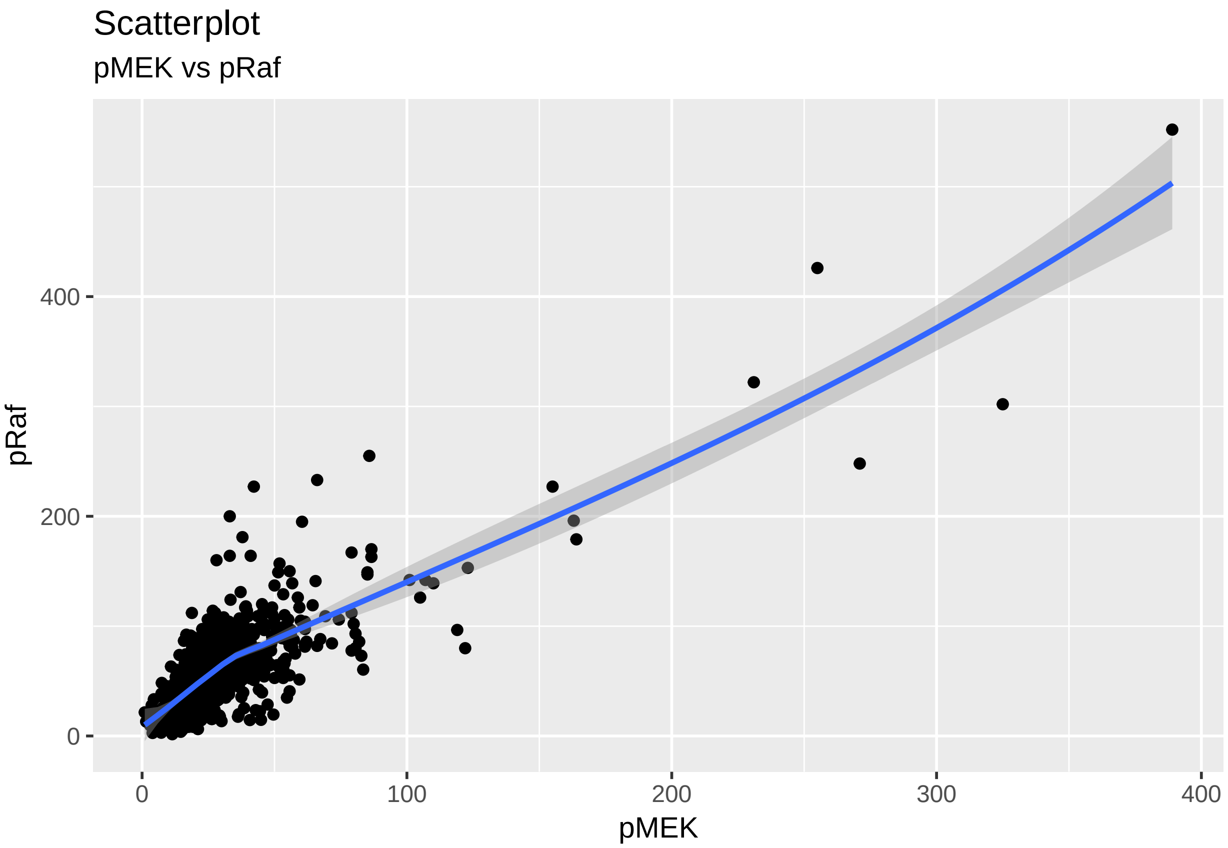
```
CM_2 <- GP1 %>% filter(treatment_num == "2") %>% select(-treatment,
    -treatment_num) %>% cor()
corrplot(CM_2, type = "upper")
```



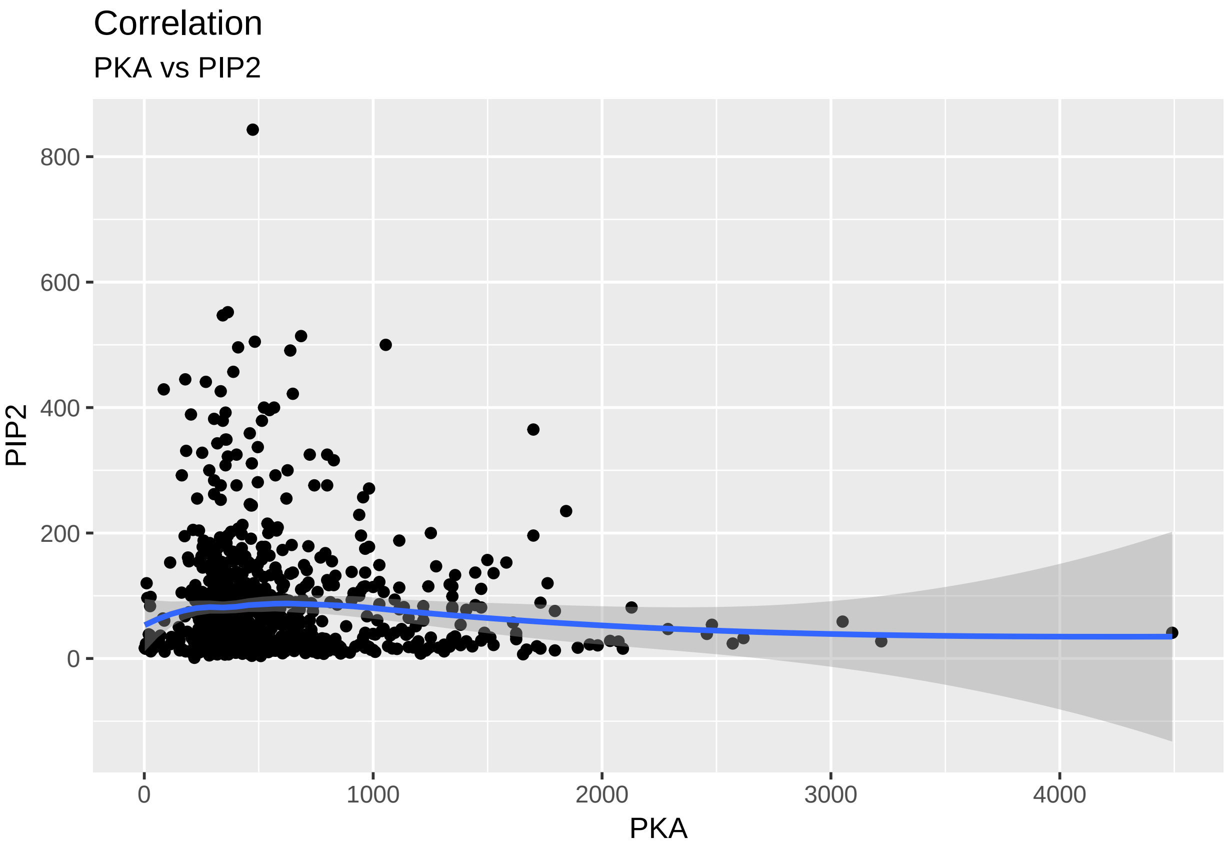
Some examples of good and bad correlation

How do correlations changes among proteins in the same condition? Mek vs Raf and PKA vs PIP2 in treatment = 0

```
a <- pmekvspraf <- GP1 %>% filter(treatment_num == "0")
ggplot(pmekvspraf, aes(x = pmek, y = praf)) + geom_point() +
    geom_smooth(method = "loess") + labs(subtitle = "pMEK vs pRaf",
    y = "pRaf", x = "pMEK", title = "Scatterplot")
```



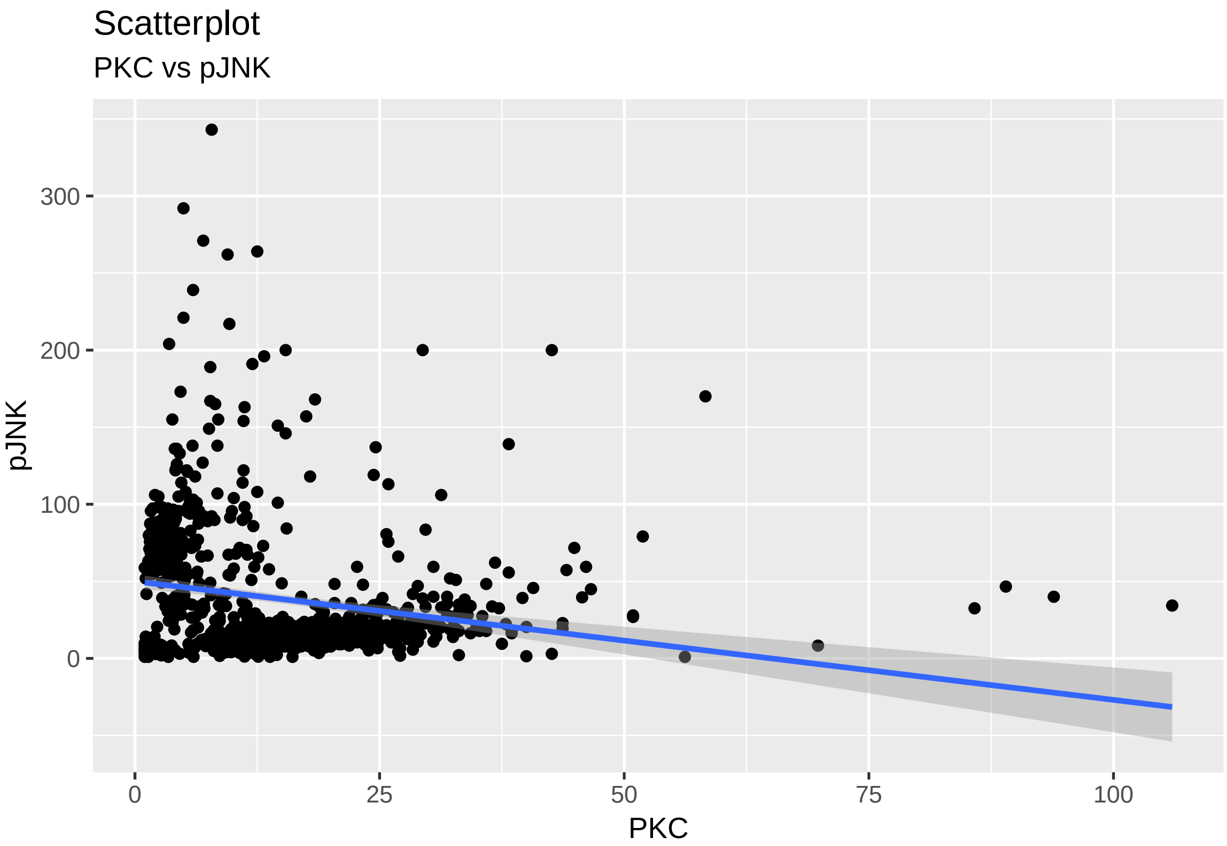
```
b <- PKAvsPIP2 <- GP1 %>% filter(treatment_num == "0")
ggplot(PKAvsPIP2, aes(x = PKA, y = PIP2)) + geom_point() + geom_smooth(method = "loess") +
    labs(subtitle = "PKA vs PIP2", y = "PIP2", x = "PKA", title = "Correlation")
```



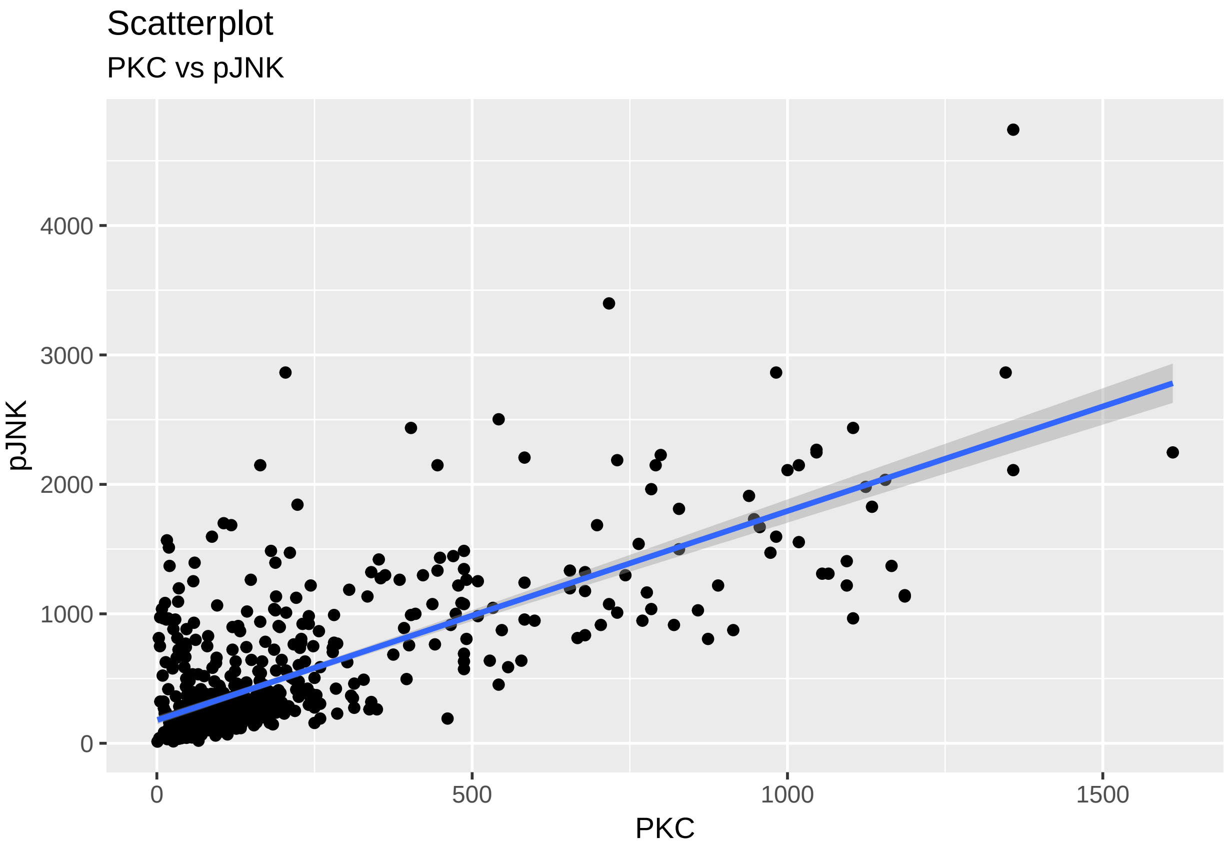
```
# tg1 <- tableGrob(a) tg2 <- tableGrob(b)

# grid.arrange(tg1,tg2, nrow=2, ncol=1)
```

How do correlations changes among treatments? PKC vs pJNK in treatment = 0 and treatment =2

```
PKCvspJNK_0 <- GP1 %>% filter(treatment_num == "0")
ggplot(PKCvspJNK_0, aes(x = PKC, y = pjnk)) + geom_point() +
    geom_smooth(method = "lm") + labs(subtitle = "PKC vs pJNK",
    y = "pJNK", x = "PKC", title = "Scatterplot")
```



```
PKCvspJNK_2 <- GP1 %>% filter(treatment_num == "2")
ggplot(PKCvspJNK_2, aes(x = PKC, y = pjnk)) + geom_point() +
    geom_smooth(method = "lm") + labs(subtitle = "PKC vs pJNK",
    y = "pJNK", x = "PKC", title = "Scatterplot")
```

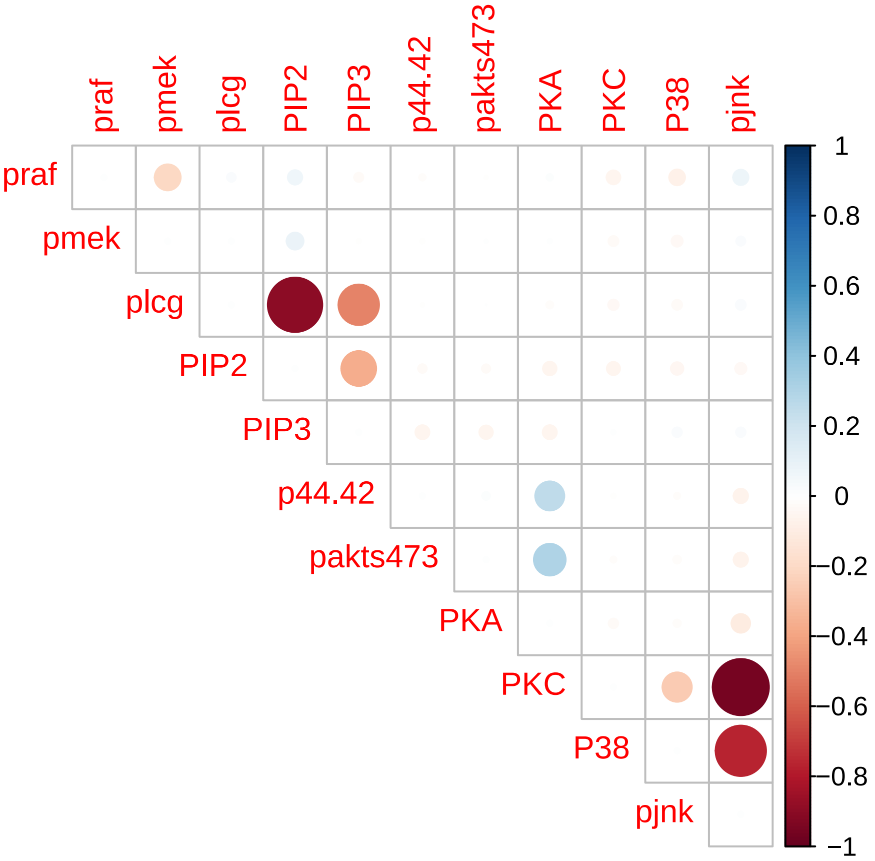


```
# grid.arrange(cond0, cond2, nrow = 1)
```

Checking how correlations change among treatments

How do the correlations change between different treatments?

```
CM2m0 <- CM_0 - CM_2
corrplot(CM2m0, type = "upper")
```



Save table

I saved the tabble as .csv document (capstone_project.csv)

```
write.table(alldf, file = "capstone_project.csv", sep = ",",
    col.names = NA)
```