irCLIP-RNP dataset of CAPRIN1 during stress granule formation

Luca Ducoli

01 July, 2024

This is the pipeline used to analyze the irCLIP-RNP of CAPRIN1 during stress granule formation from the whole RNP zone ranging from 70 to 350kDa. The experiment was performed in HEK293T. Three time points (0, 30, and 60min) were collected.

1. Prepare the dataset

```
#Needed libraries
library(DEP2)
library(tidyverse)
library(ggplot2)
library(data.table)
library(pheatmap)
library(RColorBrewer)
library(gplots)
library(hrbrthemes)
library(pacman)
library(textshape)
library(ggExtra)
library(viridis)
library(purrr)
library(hexbin)
library(DESeq2)
library(ggpubr)
library(UpSetR)
library(dplyr)
library(Clipper)
library(factoextra)
library(paletteer)
library(corrplot)
library(psych)
library(ggpmisc)
library(gprofiler2)
library(viridis)
library(GGally)
library(igraph)
library(rstatix)
library(limma)
library(HDMD)
library(cluster)
library(eulerr)
```

```
#Open proteinGroups.txt results from MaxQuant
data <- read.delim("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_timecourse/0_Data/
    proteinGroups.txt")

#Remove RPL proteins
data <- data[-grep("RPL", data$Gene.names),]
```

```
#Generate unique names and ids
unique_pg <- make_unique(data, name = "Gene.names", ids = "Protein.IDs")
unique_pg <- unique_pg %>% arrange(name)

#Get the columns
ecols <- grep("LFQ.intensity.", colnames(unique_pg))

#Keep isoform with higher LFQ intensity
iso <- grep("\\.\\d+$", unique_pg$name)
rbp <- gsub("\\.1", "", c(unique_pg$name[iso]))

#Find original row name of the isoform with higher intensity
find_max_value <- function(rbp) {
  filtered_df <- unique_pg[unique_pg$name %like% rbp, grep("LFQ.intensity.", colnames(unique_pg))
    ]
  filtered_df$rowSums <- rowSums(filtered_df[, grep("LFQ.intensity.", colnames(filtered_df))])
  max_value <- which.max(filtered_df$rowSums)
  rownames <- rownames(filtered_df)[-max_value]
  return(rownames)
}
max_iso <- c(unlist(lapply(rbp, find_max_value)))

#Remove low intensity isoforms
unique_pg <- unique_pg[!(rownames(unique_pg) %in% max_iso),]
```

```
#Load design matrix
design <- read.delim("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_timecourse/0_Data/
    Design.txt")
design
```

```
##                label condition replicate  stimulation crosslinking
## 1 LFQ.intensity.BZ30      noUV         1 unstimulated         noUV
## 2 LFQ.intensity.BZ31        T0         1 unstimulated          UVC
## 3 LFQ.intensity.BZ32        T0         2 unstimulated          UVC
## 4 LFQ.intensity.BZ33       T30         1   stimulated          UVC
## 5 LFQ.intensity.BZ34       T30         2   stimulated          UVC
## 6 LFQ.intensity.BZ35       T60         1   stimulated          UVC
## 7 LFQ.intensity.BZ36       T60         2   stimulated          UVC
```

2. Determine the UVC-erniched proteins

We used ClippeR R package (PMID: 34635147) to determined the UVC-enriched proteins.

```
#Create a SummarizedExperiment
ecols <- grep("LFQ.intensity.", colnames(unique_pg))
se <- make_se(unique_pg, columns = ecols, expdesign = design)
se_UVC <- se[,se$crosslinking != "noUV" ]
se_UVC <- filter_se(se_UVC, thr = 0, filter_formula = ~ Reverse != '+' & Potential.contaminant !=
    "+" & Peptides > 1 & Unique.peptides > 0)
se <- se[rownames(se) %in% rownames(se_UVC),]
write.table(se@assays@data@listData, file="~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/
    CAPRIN1_timecourse/2_DEP/CAPRIN_LFQ_intensity_raw.txt", quote = F, row.names = T, sep = "\t")

set.seed(3)
clipper <- function(flt, time) {
se_time <- flt[,flt$condition == time]
se <- se[rownames(se_time), se$condition %in% c("noUV", time) ]
```

```
imputed <- DEP2::impute(se, fun = "QRILC")
data <- as.data.frame(assay(imputed))
clipper = Clipper(score.exp = as.matrix(data[,c(2,3)]), score.back = as.matrix(data[,-c(2,3)]),
    FDR = 0.05, analysis = "e")
data$FDR <- clipper$q
data <- cbind(data, rowMeans(data[,c(2,3)])-data[1])
data$TP <- time
data$name <- rownames(data)
rownames(data) <- NULL
colnames(data)[5] <- c("logFC")
deg <- subset(data, FDR < 0.1 & logFC > log2(3))
return(list(data = data, deg = deg))
}

T0_clipper <-  clipper(se, "T0")
T30_clipper <- clipper(se, "T30")
T60_clipper <- clipper(se, "T60")

write.table(T0_clipper$data, file="~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_
    timecourse/2_DEP/CAPRIN_clipper_results_T0.txt", quote = F, row.names = F, sep = "\t")
write.table(T30_clipper$data, file="~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_
    timecourse/2_DEP/CAPRIN_clipper_results_T30.txt", quote = F, row.names = F, sep = "\t")
write.table(T60_clipper$data, file="~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_
    timecourse/2_DEP/CAPRIN_clipper_results_T60.txt", quote = F, row.names = F, sep = "\t")

CAPRIN_UVC <- rbind(T0_clipper$deg[,4:7], T30_clipper$deg[,4:7], T60_clipper$deg[,4:7])
CAPRIN_UVC_genes <- unique(CAPRIN_UVC$name)
CAPRIN_UVC
```

```
##              FDR    logFC TP       name
## 3     0.03773585 2.169631 T0       ACIN1
## 8     0.03773585 2.772824 T0       ATXN2
## 17    0.03773585 2.230398 T0         CAD
## 22    0.03773585 4.694068 T0       CDCA2
## 31    0.03773585 3.085259 T0       CSDE1
## 34    0.03773585 3.280484 T0         DCD
## 44    0.03773585 2.046629 T0       DDX3X
## 51    0.03773585 2.026989 T0       DDX56
## 59    0.03773585 3.251407 T0       DNMT1
## 62    0.03773585 5.121508 T0      EEF1A1
## 67    0.03773585 1.988573 T0      ELAVL1
## 72    0.04878049 1.892060 T0        FMR1
## 75    0.03773585 3.621223 T0         FUS
## 76    0.03773585 3.748800 T0        FXR1
## 77    0.03773585 3.738969 T0        FXR2
## 78    0.03773585 4.764645 T0       G3BP1
## 86    0.03773585 2.916895 T0    HIST1H1C
## 87    0.04878049 1.745264 T0    HIST3H2A
## 88    0.03773585 2.370417 T0     HNRNPA1
## 89    0.03773585 3.353308 T0   HNRNPA2B1
## 90    0.03773585 2.945718 T0      HNRNPC
## 91    0.03773585 5.545154 T0      HNRNPF
## 92    0.03773585 4.927953 T0     HNRNPH1
## 93    0.03773585 2.312214 T0      HNRNPK
## 94    0.03773585 2.141483 T0      HNRNPL
## 95    0.03773585 2.241991 T0      HNRNPM
## 96    0.03773585 2.681650 T0      HNRNPR
## 99    0.03773585 2.821590 T0    HSP90AB1
## 100   0.03773585 2.132053 T0      HSPA1A
## 101   0.04878049 1.886912 T0     IGF2BP1
## 103   0.03773585 5.338276 T0       IGHG1
## 107   0.03773585 2.556717 T0       KHSRP
## 113   0.03773585 5.715124 T0      LRPPRC
## 119   0.03773585 2.597072 T0        MDC1
## 124   0.03773585 5.358020 T0         NCL
## 127   0.04878049 1.660961 T0       NFXL1
```

```
## 133   0.03773585 2.633291   T0      NOP14
## 137   0.03773585 4.857316   T0       NPM1
## 141   0.03773585 2.894352   T0     PABPC1
## 149   0.04878049 1.838487   T0       PHF6
## 162   0.03773585 2.861047   T0      RPS11
## 164   0.03773585 5.738320   T0       RPS2
## 166   0.04878049 1.857855   T0      RPS24
## 167   0.03773585 2.968374   T0      RPS26
## 168   0.03773585 4.191498   T0       RPS3
## 169   0.03773585 2.524558   T0      RPS3A
## 170   0.03773585 2.505033   T0      RPS4X
## 172   0.03773585 3.111346   T0       RPS7
## 176   0.03773585 3.758833   T0      RRP12
## 180   0.03773585 4.052574   T0     SERBP1
## 181   0.03773585 4.498833   T0       SFPQ
## 183   0.03773585 3.023340   T0    SMARCA5
## 184   0.03773585 3.854731   T0       SND1
## 185   0.03773585 2.065353   T0    SNRNP200
## 189   0.03773585 2.021333   T0      SRPK1
## 193   0.03773585 3.560074   T0     SYNCRIP
## 199   0.04878049 1.675527   T0       TOP1
## 210   0.03773585 4.082811   T0     UBAP2L
## 212   0.04878049 1.773150   T0      UPF3B
## 219   0.03773585 2.971594   T0       YBX1
## 221   0.04878049 1.628244   T0     ZC3H11A
## 228   0.04878049 1.819437   T0     ZNF280C
## 230   0.03773585 2.650931   T0      ZNF48
## 10    0.07692308 1.605606  T30       BAZ1B
## 11    0.02857143 3.708702  T30       BAZ2A
## 28    0.05555556 2.000559  T30       CKAP2
## 311   0.02857143 6.133811  T30       CSDE1
## 441   0.05555556 1.912250  T30       DDX3X
## 54    0.05555556 2.092090  T30       DHX30
## 591   0.02857143 3.206238  T30       DNMT1
## 671   0.02857143 4.086809  T30      ELAVL1
## 71    0.05555556 1.914577  T30         FBL
## 721   0.02857143 2.653106  T30        FMR1
## 74    0.02857143 4.560017  T30       FUBP3
## 751   0.02857143 3.113553  T30         FUS
## 761   0.05555556 2.123598  T30        FXR1
## 771   0.06896552 1.820970  T30        FXR2
## 781   0.02857143 3.232470  T30       G3BP1
## 861   0.02857143 3.189342  T30    HIST1H1C
## 881   0.06896552 1.624580  T30      HNRNPA1
## 891   0.02857143 5.895342  T30   HNRNPA2B1
## 901   0.02857143 2.823643  T30      HNRNPC
## 921   0.02857143 6.059913  T30     HNRNPH1
## 931   0.05555556 2.005732  T30      HNRNPK
## 941   0.02857143 3.038330  T30      HNRNPL
## 951   0.02857143 3.269821  T30      HNRNPM
## 991   0.05555556 2.076075  T30    HSP90AB1
## 1001  0.02857143 2.696694  T30      HSPA1A
## 1011  0.05555556 2.244918  T30     IGF2BP1
## 1031  0.02857143 4.261668  T30       IGHG1
## 1131  0.02857143 4.852124  T30      LRPPRC
## 118   0.02857143 5.313008  T30        MARS
## 1191  0.05263158 2.510758  T30        MDC1
## 1241  0.02857143 5.645755  T30         NCL
## 1271  0.02857143 3.053412  T30       NFXL1
## 1331  0.05555556 2.019744  T30      NOP14
## 1371  0.02857143 4.236133  T30       NPM1
## 1411  0.05555556 2.282763  T30     PABPC1
## 147   0.06896552 1.609178  T30       PHF2
## 1491  0.05263158 2.539820  T30       PHF6
## 153   0.02857143 2.748150  T30      PTBP1
## 1621  0.05555556 2.168289  T30      RPS11
## 1641  0.02857143 6.776957  T30       RPS2
## 1671  0.05263158 2.522224  T30      RPS26
```

```
## 1691 0.02857143 3.567995  T30      RPS3A
## 1701 0.06896552 1.702162  T30      RPS4X
## 174  0.05555556 2.055656  T30       RPS9
## 1761 0.05555556 2.325200  T30      RRP12
## 1801 0.02857143 5.191610  T30     SERBP1
## 1811 0.02857143 3.936344  T30       SFPQ
## 1831 0.02857143 2.813502  T30    SMARCA5
## 1891 0.02857143 4.760515  T30      SRPK1
## 192  0.02857143 2.633799  T30      STT3B
## 1931 0.02857143 3.459988  T30    SYNCRIP
## 195  0.02857143 4.952770  T30      TDRD3
## 198  0.02857143 2.751729  T30      TIAL1
## 1991 0.05555556 1.896960  T30       TOP1
## 202  0.05555556 2.357456  T30      TOP3B
## 207  0.02857143 4.080510  T30      U2AF2
## 2121 0.05555556 1.882293  T30      UPF3B
## 2191 0.02857143 4.543290  T30       YBX1
## 2301 0.02857143 2.750881  T30      ZNF48
## 4    0.05263158 2.935222  T60      ACTA1
## 111  0.05263158 2.867001  T60     BAZ2A
## 312  0.05263158 5.602257  T60      CSDE1
## 341  0.05263158 3.507318  T60        DCD
## 442  0.09803922 1.743731  T60      DDX3X
## 592  0.05263158 2.648051  T60      DNMT1
## 621  0.06976744 2.091934  T60      EEF1A1
## 672  0.05263158 4.085978  T60     ELAVL1
## 711  0.05263158 2.360801  T60        FBL
## 722  0.05263158 2.346995  T60       FMR1
## 741  0.05263158 3.158318  T60      FUBP3
## 752  0.05263158 3.020144  T60        FUS
## 772  0.09803922 1.895343  T60       FXR2
## 782  0.05263158 3.038115  T60      G3BP1
## 862  0.05263158 3.030668  T60   HIST1H1C
## 892  0.05263158 6.107055  T60 HNRNPA2B1
## 902  0.05263158 3.060640  T60     HNRNPC
## 922  0.05263158 6.001094  T60    HNRNPH1
## 932  0.06976744 2.204697  T60     HNRNPK
## 942  0.05263158 3.597231  T60     HNRNPL
## 952  0.05263158 3.512366  T60     HNRNPM
## 992  0.05263158 2.280296  T60   HSP90AB1
## 1002 0.09803922 1.891784  T60     HSPA1A
## 1012 0.06976744 2.165613  T60    IGF2BP1
## 1181 0.05263158 3.937063  T60       MARS
## 1192 0.05263158 2.604335  T60       MDC1
## 1242 0.05263158 5.708128  T60        NCL
## 1372 0.05263158 4.347834  T60       NPM1
## 1412 0.09803922 1.965583  T60     PABPC1
## 1492 0.05263158 2.413857  T60       PHF6
## 1531 0.05263158 2.583643  T60      PTBP1
## 1622 0.08888889 1.981721  T60       RPS11
## 1642 0.05263158 6.576949  T60       RPS2
## 1692 0.05263158 3.638107  T60      RPS3A
## 1702 0.09803922 1.856029  T60      RPS4X
## 1741 0.05263158 2.339771  T60       RPS9
## 1762 0.08888889 2.001339  T60      RRP12
## 1802 0.05263158 5.167012  T60     SERBP1
## 1812 0.05263158 3.670013  T60       SFPQ
## 1832 0.05263158 2.844567  T60    SMARCA5
## 1921 0.05263158 2.405711  T60      STT3B
## 1932 0.05263158 2.805514  T60    SYNCRIP
## 1951 0.05263158 3.674443  T60      TDRD3
## 1992 0.06976744 2.084685  T60       TOP1
## 201  0.05263158 3.107122  T60      TOP2B
## 2021 0.05263158 2.735602  T60      TOP3B
## 2071 0.05263158 4.124639  T60      U2AF2
## 2122 0.09803922 1.750227  T60      UPF3B
## 2192 0.05263158 4.426913  T60       YBX1
## 2281 0.06976744 2.137822  T60    ZNF280C
```

```
## 2302 0.05263158 2.626781 T60      ZNF48
```

3. Differential enrichment analysis

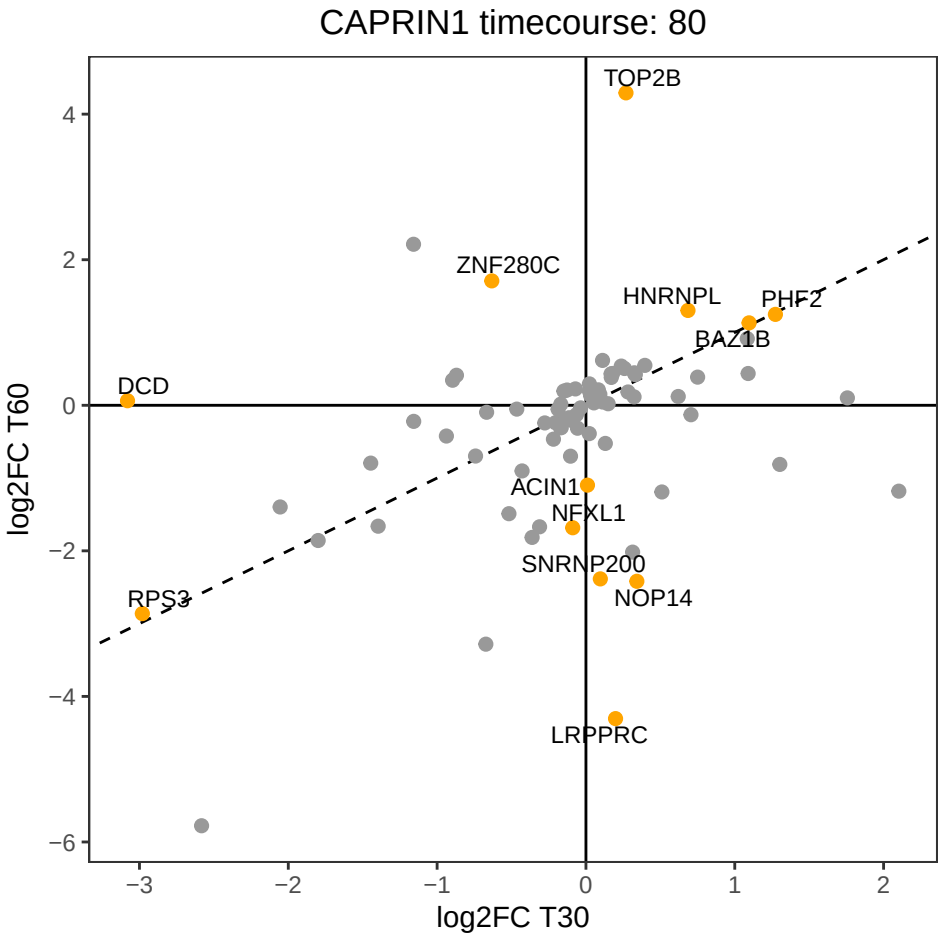
We performed differential enrichment analysis using Limma and generate the scatterplot.

```
#Prepare the SummarizedExperiment
norm <- normalize_vsn(se_UVC)
imputed <- DEP2::impute(norm, fun = "QRILC")

model_vsn <- model.matrix(~ condition, colData(imputed))

#Interaction analysis
CAPRIN1_fit1_norm_int_vsn = lmFit(assay(imputed), design = model_vsn)
CAPRIN1_fit2_norm_int_vsn <- eBayes(CAPRIN1_fit1_norm_int_vsn)
CAPRIN1_int_norm_vsn_both <- topTable(CAPRIN1_fit2_norm_int_vsn, coef = c("conditionT30","
    conditionT60"), number = length(rownames(imputed)))
write.table(CAPRIN1_int_norm_vsn_both, file="~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/
    CAPRIN1_timecourse/2_DEP/CAPRIN_DEP_results_onefactor.txt", quote = F, row.names = T, sep = "
    \t")
CAPRIN1_int_norm_vsn_T30 <- topTable(CAPRIN1_fit2_norm_int_vsn, coef = c("conditionT30"), number
    = length(rownames(imputed)))
CAPRIN1_int_norm_vsn_T60 <- topTable(CAPRIN1_fit2_norm_int_vsn, coef = c("conditionT60"), number
    = length(rownames(imputed)))
CAPRIN1_int_norm_vsn_both$name <- rownames(CAPRIN1_int_norm_vsn_both)
CAPRIN1_int_norm_vsn_T30$name <- rownames(CAPRIN1_int_norm_vsn_T30)
CAPRIN1_int_norm_vsn_T60$name <- rownames(CAPRIN1_int_norm_vsn_T60)
CAPRIN1_sign_prot <- subset(CAPRIN1_int_norm_vsn_both, adj.P.Val < 0.1)
CAPRIN1_sign_prot <- subset(CAPRIN1_sign_prot, abs(conditionT30) > 1 | abs(conditionT60) > 1)

#Subset with the UVC proteins
CAPRIN1_int_norm_vsn_both <- subset(CAPRIN1_int_norm_vsn_both, rownames(CAPRIN1_int_norm_vsn_both
    ) %in% CAPRIN_UVC_genes)

#Scatterplot
CAPRIN1_int_norm_vsn_both$int_sign <- CAPRIN1_int_norm_vsn_both$name %in% rownames(CAPRIN1_sign_
    prot)
CAPRIN1_colors <- c("FALSE" = "#999999","TRUE" =  "orange")
```

Volcano plot

```
ggplot <- ggplot(data=CAPRIN1_int_norm_vsn_both, aes(x=conditionT30, y=conditionT60)) + geom_
    vline(xintercept = 0) + geom_hline(yintercept = 0) +  geom_abline(intercept = 0, linetype=2)
    +
  geom_point(shape=19, size=2, aes(col = int_sign)) +
  labs(title = paste("CAPRIN1 timecourse:", nrow(CAPRIN1_int_norm_vsn_both))  , x = expression("
    log2FC T30"), y = expression("log2FC T60")) +
  scale_color_manual(values = CAPRIN1_colors) +
  ggrepel::geom_text_repel(data = CAPRIN1_int_norm_vsn_both[CAPRIN1_int_norm_vsn_both$int_sign ==
    "TRUE",], aes(label = name), size = 3, box.padding = unit(0.1, "lines"), point.padding =
    unit(0.1, "lines"), segment.size = 0.5,max.overlaps = Inf) +
  theme_bw() +
  theme(legend.position = "none", panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_blank(),
        plot.title = element_text(hjust = 0.5))# +
```

```
ggplot
```

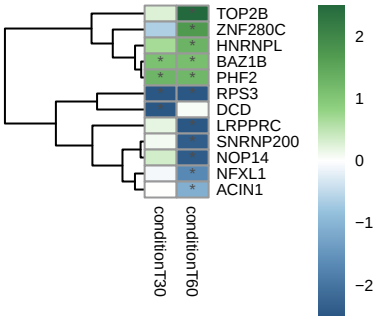



```
#Save the plot as pdf
pdf("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_timecourse/2_DEP/CAPRIN1_DEP_plot.
    pdf", height = 5, width = 5)
ggplot
dev.off()
```

Heatmap of significant UVC-enriched proteins

```
#Heatmap
CAPRIN1_sign_T30 <- subset(CAPRIN1_int_norm_vsn_T30,   adj.P.Val < 0.1 & abs(logFC) > 1)
CAPRIN1_sign_T60 <- subset(CAPRIN1_int_norm_vsn_T60,   adj.P.Val < 0.1 & abs(logFC) > 1)

CAPRIN1_sign_T30 <- subset(CAPRIN1_sign_T30, rownames(CAPRIN1_sign_T30) %in% CAPRIN_UVC_genes)
CAPRIN1_sign_T60 <- subset(CAPRIN1_sign_T60, rownames(CAPRIN1_sign_T60) %in% CAPRIN_UVC_genes)

lt.tsk = list(T30 = rownames(CAPRIN1_sign_T30),
              T60 = rownames(CAPRIN1_sign_T60))

fromList <- function (input) {
```

```
  # Same as original fromList()...
  elements <- unique(unlist(input))
  data <- unlist(lapply(input, function(x) {
    x <- as.vector(match(elements, x))
  }))
  data[is.na(data)] <- as.integer(0)
  data[data != 0] <- as.integer(1)
  data <- data.frame(matrix(data, ncol = length(input), byrow = F))
  data <- data[which(rowSums(data) != 0), ]
  names(data) <- names(input)
  # ... Except now it conserves your original value names!
  row.names(data) <- elements
  return(data)
}

#Get significant genes
sign.proteins <- fromList(lt.tsk)

#Matrix
CAPRIN1_sign_prot_HM <- CAPRIN1_sign_prot[,c(1,2)]
CAPRIN1_sign_prot_HM <- subset(CAPRIN1_sign_prot_HM, rownames(CAPRIN1_sign_prot_HM) %in% CAPRIN_
    UVC_genes)

#Annotation and color
my.breaks <- c(seq(-2.5, 2.5, by=0.01))
my.colors <- c(rev(paletteer_c("ggthemes::Green-Blue-White Diverging", length(my.breaks))))

labels <- sign.proteins
colnames(labels) <- colnames(CAPRIN1_sign_prot_HM)
labels[labels == 1] <- "*"
labels[labels == 0] <- ""
labels <- labels[match(rownames(CAPRIN1_sign_prot_HM), rownames(labels)),]

pheatmap(
  mat                = CAPRIN1_sign_prot_HM,
  cellwidth = 12,
  cellheight = 6,
  display_numbers = labels,
  fontsize_number=5.5,
  color = my.colors,
  breaks = my.breaks,
  clustering_distance_cols = "euclidean",
  show_colnames      = TRUE,
  show_rownames      = TRUE,
  drop_levels        = TRUE,
  fontsize           = 5.5,
  cluster_rows       = TRUE,
  cluster_cols       = FALSE
)
```

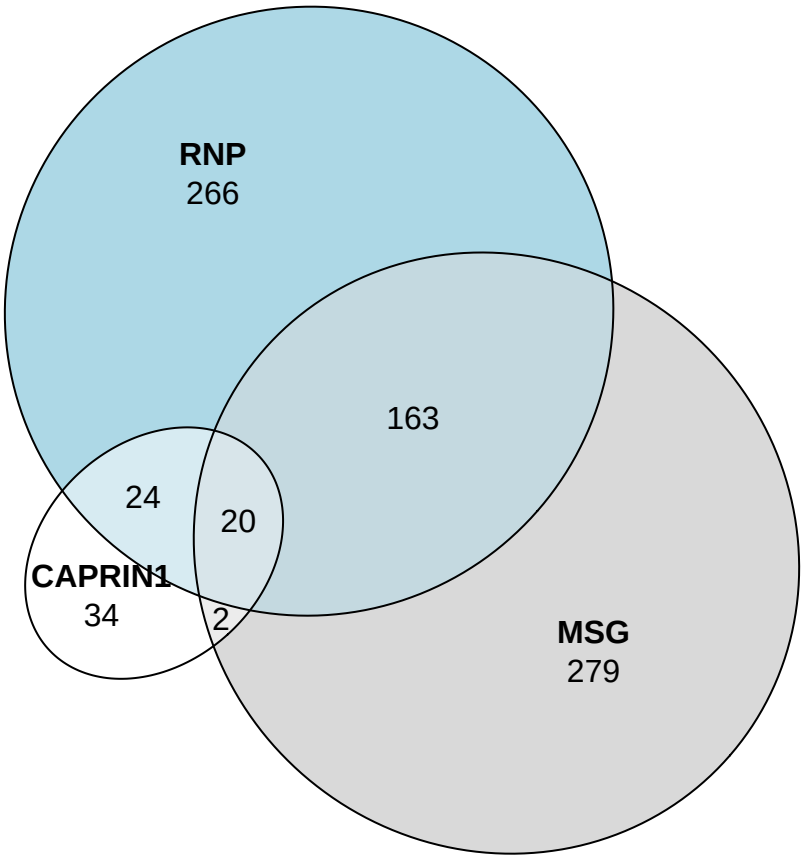



```
#Save the plot as pdf
pheatmap(
  mat                = CAPRIN1_sign_prot_HM,
  cellwidth = 12,
  cellheight = 6,
  display_numbers = labels,
  fontsize_number=5.5,
  color = my.colors,
  breaks = my.breaks,
  clustering_distance_cols = "euclidean",
  show_colnames      = TRUE,
  show_rownames      = TRUE,
  drop_levels        = TRUE,
  fontsize           = 5.5,
  cluster_rows       = TRUE,
  cluster_cols       = FALSE,
  filename = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_timecourse/2_DEP/CAPRIN1_
    sign_HM.pdf",
  width = 5,
  height = 5
)
```

Overlap with existing databases.

```
#Load files
```

```
res2 <- read.delim("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_timecourse/3_SG_
    Database/1_Venn_Diagram/MSGP_database_proteins.txt", header = TRUE, dec = ".")
res3 <- read.delim("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_timecourse/3_SG_
    Database/1_Venn_Diagram/RNP_granule_database.txt", header = TRUE, dec = ".")

s4 <- list(CAPRIN1 = CAPRIN_UVC_genes,
           MSG = res2$Gene,
           RNP = res3$Gene)

fromList <- function (input) {
  elements <- unique(unlist(input))
  data <- unlist(lapply(input, function(x) {
    x <- as.vector(match(elements, x))
  }))
  data[is.na(data)] <- as.integer(0)
  data[data != 0] <- as.integer(1)
  data <- data.frame(matrix(data, ncol = length(input), byrow = F))
  data <- data[which(rowSums(data) != 0), ]
  names(data) <- names(input)
  row.names(data) <- elements
  return(data)
}

sign.proteins <- fromList(s4)
write.table(sign.proteins, file = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_
    timecourse/3_SG_Database/1_Venn_Diagram/CAPRIN1_upset_database.txt", row.names = TRUE, sep =
    "\t")

plot(euler(s4, shape = "ellipse"), quantities = TRUE)
```



```
pdf("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/CAPRIN1_timecourse/3_SG_Database/1_Venn_
    Diagram/CAPRIN1_MSGP_venn.pdf", height = 7, width = 7)
plot(euler(s4, shape = "ellipse"), quantities = TRUE)
dev.off()
```

```
## pdf
##   2
```

All the visualizations were saved as pdf and modified in illustrator.

```
sessionInfo()
```

```
## R version 4.2.1 (2022-06-23)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur ... 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.2/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.2/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats4    stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] eulerr_7.0.1                cluster_2.1.6
##  [3] HDMD_1.2                    MASS_7.3-60.0.1
##  [5] rstatix_0.7.2               igraph_2.0.3
##  [7] GGally_2.2.1                gprofiler2_0.2.3
##  [9] ggpmisc_0.5.5               ggpp_0.5.6
## [11] psych_2.4.3                 corrplot_0.92
## [13] paletteer_1.6.0             factoextra_1.0.7
## [15] Clipper_0.0.0.9000          UpSetR_1.4.0
## [17] ggpubr_0.6.0                DESeq2_1.38.3
## [19] hexbin_1.28.3               viridis_0.6.5
## [21] viridisLite_0.4.2           ggExtra_0.10.1
## [23] textshape_1.7.3             pacman_0.5.1
## [25] hrbrthemes_0.8.7            gplots_3.1.3.1
## [27] RColorBrewer_1.1-3          pheatmap_1.0.12
## [29] data.table_1.15.2           lubridate_1.9.3
## [31] forcats_1.0.0               stringr_1.5.1
## [33] dplyr_1.1.4                 purrr_1.0.2
## [35] readr_2.1.5                 tidyr_1.3.1
## [37] tibble_3.2.1                ggplot2_3.5.0
## [39] tidyverse_2.0.0             DEP2_0.4.8.24
## [41] R6_2.5.1                    limma_3.54.2
## [43] MSnbase_2.24.2              ProtGenerics_1.30.0
## [45] mzR_2.32.0                  Rcpp_1.0.12
## [47] MsCoreUtils_1.10.0          SummarizedExperiment_1.28.0
## [49] Biobase_2.58.0              GenomicRanges_1.50.2
## [51] GenomeInfoDb_1.34.9         IRanges_2.32.0
## [53] S4Vectors_0.36.2            BiocGenerics_0.44.0
## [55] MatrixGenerics_1.10.0       matrixStats_1.2.0
##
## loaded via a namespace (and not attached):
##   [1] SparseM_1.81                ggthemes_5.1.0
##   [3] missForest_1.5              bit64_4.0.5
##   [5] knitr_1.45                  DelayedArray_0.24.0
```

```
##   [7] KEGGREST_1.38.0              RCurl_1.98-1.14
##   [9] AnnotationFilter_1.22.0      doParallel_1.0.17
##  [11] generics_0.1.3               preprocessCore_1.60.2
##  [13] RSQLite_2.3.5                proxy_0.4-27
##  [15] bit_4.0.5                    tzdb_0.4.0
##  [17] httpuv_1.6.14                assertthat_0.2.1
##  [19] TCseq_1.22.6                 xfun_0.42
##  [21] hms_1.1.3                    evaluate_0.23
##  [23] promises_1.2.1               fansi_1.0.6
##  [25] caTools_1.18.2               htmlwidgets_1.6.4
##  [27] DBI_1.2.2                    geneplotter_1.76.0
##  [29] ellipsis_0.3.2               RSpectra_0.16-1
##  [31] QFeatures_1.8.0              backports_1.4.1
##  [33] fontLiberation_0.1.0         prismatic_1.1.1
##  [35] annotate_1.76.0              fontBitstreamVera_0.1.1
##  [37] vctrs_0.6.5                  imputeLCMD_2.1
##  [39] quantreg_5.97                abind_1.4-5
##  [41] cachem_1.0.8                 withr_3.0.0
##  [43] itertools_0.1-3              GenomicAlignments_1.34.1
##  [45] fdrtool_1.2.17               MultiAssayExperiment_1.24.0
##  [47] mnormt_2.1.1                 lazyeval_0.2.2
##  [49] crayon_1.5.2                 crul_1.4.0
##  [51] labeling_0.4.3               glmnet_4.1-8
##  [53] edgeR_3.40.2                 pkgconfig_2.0.3
##  [55] nlme_3.1-164                 rlang_1.1.3
##  [57] lifecycle_1.0.4              miniUI_0.1.1.1
##  [59] sandwich_3.1-0               MatrixModels_0.5-3
##  [61] downloader_0.4               fontquiver_0.2.1
##  [63] httpcode_0.3.0               affyio_1.68.0
##  [65] extrafontdb_1.0              polyclip_1.10-6
##  [67] randomForest_4.7-1.1         rngtools_1.5.2
##  [69] Matrix_1.6-5                 carData_3.0-5
##  [71] zoo_1.8-12                   GlobalOptions_0.1.2
##  [73] png_0.1-8                    rjson_0.2.21
##  [75] bitops_1.0-7                 KernSmooth_2.23-22
##  [77] Biostrings_2.66.0            blob_1.2.4
##  [79] doRNG_1.8.6                  shape_1.4.6.1
##  [81] tmvtnorm_1.6                 ggsignif_0.6.4
##  [83] scales_1.3.0                 memoise_2.0.1
##  [85] magrittr_2.0.3               plyr_1.8.9
##  [87] zlibbioc_1.44.0              compiler_4.2.1
##  [89] pcaMethods_1.90.0            clue_0.3-65
##  [91] Rsamtools_2.14.0             cli_3.6.2
##  [93] affy_1.76.0                  XVector_0.38.0
##  [95] tidyselect_1.2.1             vsn_3.66.0
##  [97] stringi_1.8.3                highr_0.10
##  [99] yaml_2.3.8                   norm_1.0-11.1
## [101] askpass_1.2.0                locfit_1.5-9.9
## [103] MALDIquant_1.22.2            ggrepel_0.9.5
## [105] grid_4.2.1                   ggstats_0.5.1
## [107] polynom_1.4-1                tools_4.2.1
## [109] timechange_0.3.0             parallel_4.2.1
## [111] circlize_0.4.16              rstudioapi_0.15.0
## [113] foreach_1.5.2                gridExtra_2.3
## [115] farver_2.1.1                 mzID_1.36.0
## [117] Rtsne_0.17                   digest_0.6.35
## [119] BiocManager_1.30.22          shiny_1.8.0
## [121] gfonts_0.2.0                 car_3.1-2
## [123] broom_1.0.5                  later_1.3.2
## [125] ncdf4_1.22                   httr_1.4.7
## [127] gdtools_0.3.5                AnnotationDbi_1.60.2
## [129] ComplexHeatmap_2.14.0        colorspace_2.1-0
## [131] polylabelr_0.2.0             XML_3.99-0.16.1
## [133] reticulate_1.35.0            umap_0.2.10.0
## [135] splines_4.2.1                rematch2_2.1.2
## [137] gmm_1.8                      plotly_4.10.4
## [139] systemfonts_1.0.5            xtable_1.8-4
## [141] jsonlite_1.8.8               pillar_1.9.0
```

```
## [143] htmltools_0.5.7              mime_0.12
## [145] glue_1.7.0                   fastmap_1.1.1
## [147] BiocParallel_1.32.6          class_7.3-22
## [149] codetools_0.2-19             mvtnorm_1.2-4
## [151] utf8_1.2.4                   lattice_0.22-5
## [153] curl_5.2.1                   gtools_3.9.5
## [155] openssl_2.1.1                Rttf2pt1_1.3.12
## [157] survival_3.5-8               rmarkdown_2.26
## [159] munsell_0.5.0                e1071_1.7-14
## [161] GetoptLong_1.0.5             GenomeInfoDbData_1.2.9
## [163] iterators_1.0.14             impute_1.72.3
## [165] reshape2_1.4.4               gtable_0.3.4
## [167] extrafont_0.19
```