irCLIP-RNP dataset to compare 1U/µL vs 0.02U/µL RNase I digestions

Luca Ducoli

This is the pipeline used to analyze the irCLIP-RNP datasets after digestion with two different RNase doses (1U/µL and 0.02U/µL), named here as "high": 1U/µL and "low": 0.02U/µL. Gel sections ranging from 70 to 350KDa (a.k.a "whole RNP zone") were subjected to MS. The experiment was performed in HEK293T cells.

1. Prepare the dataset

```r
#Load the libraries
library(formatR)
library(DEP2)
library(tidyverse)
library(ggpubr)
library(Clipper)
library(viridis)
library(patchwork)
library(hrbrthemes)
library(igraph)
library(ggraph)
library(colormap)
library(UpSetR)
library(ggplot2)
library(arcdiagram)
library(pheatmap)
library(grid)
library(DESeq2)
library(data.table)
library(eulerr)
library(SuperExactTest)
```

In the first step, we prepared the dataset to create a SummarizedExperiment object starting from the proteinGroups.txt output file from MaxQuant.

```r
# Open proteinGroups.txt results from MaxQuant
data <- read.delim("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/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 LFQ intensity
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), ]

head(unique_pg, n = 2)
```

```
##       Protein.IDs Majority.protein.IDs Peptide.counts..all.
## 1 Q8NE71;Q8NE71-2      Q8NE71;Q8NE71-2                23;22
## 2 Q9UG63;Q9UG63-2      Q9UG63;Q9UG63-2                  3;3
##   Peptide.counts..razor.unique. Peptide.counts..unique.
## 1                         23;22                   23;22
## 2                           3;3                     3;3
##                                Protein.names Gene.names
## 1 ATP-binding cassette sub-family F member 1      ABCF1
## 2 ATP-binding cassette sub-family F member 2      ABCF2
##

                                          Fasta.headers
## 1 sp|Q8NE71|ABCF1_HUMAN ATP-binding cassette sub-family F member 1 OS=Homo sapiens OX=9606 GN=
    ABCF1 PE=1 SV=2;sp|Q8NE71-2|ABCF1_HUMAN Isoform 2 of ATP-binding cassette sub-family F member
     1 OS=Homo sapiens OX=9606 GN=ABCF1
## 2 sp|Q9UG63|ABCF2_HUMAN ATP-binding cassette sub-family F member 2 OS=Homo sapiens OX=9606 GN=
    ABCF2 PE=1 SV=2;sp|Q9UG63-2|ABCF2_HUMAN Isoform 2 of ATP-binding cassette sub-family F member
     2 OS=Homo sapiens OX=9606 GN=ABCF2
##   Number.of.proteins Peptides Razor...unique.peptides Unique.peptides
## 1                  2       23                      23              23
## 2                  2        3                       3               3
##   Peptides.BZ10 Peptides.BZ11 Peptides.BZ12 Peptides.BZ13 Peptides.BZ14
## 1            14            21            16            16            16
## 2             1             2             1             1             1
##   Peptides.BZ15 Peptides.BZ16 Peptides.BZ17 Peptides.BZ18 Peptides.BZ19
## 1            14             0             1             0             0
## 2             2             0             0             0             0
##   Peptides.BZ20 Peptides.BZ21 Peptides.BZ6 Peptides.BZ7 Peptides.BZ8
## 1             0             0           14            0            0
## 2             0             0            0            0            0
##   Peptides.BZ9 Razor...unique.peptides.BZ10 Razor...unique.peptides.BZ11
## 1           15                           14                           21
## 2            0                            1                            2
##   Razor...unique.peptides.BZ12 Razor...unique.peptides.BZ13
## 1                           16                           16
## 2                            1                            1
##   Razor...unique.peptides.BZ14 Razor...unique.peptides.BZ15
## 1                           16                           14
## 2                            1                            2
##   Razor...unique.peptides.BZ16 Razor...unique.peptides.BZ17
## 1                            0                            1
## 2                            0                            0
##   Razor...unique.peptides.BZ18 Razor...unique.peptides.BZ19
## 1                            0                            0
## 2                            0                            0
##   Razor...unique.peptides.BZ20 Razor...unique.peptides.BZ21
## 1                            0                            0
## 2                            0                            0
##   Razor...unique.peptides.BZ6 Razor...unique.peptides.BZ7
## 1                          14                           0
## 2                           0                           0
##   Razor...unique.peptides.BZ8 Razor...unique.peptides.BZ9 Unique.peptides.BZ10
## 1                           0                          15                   14
## 2                           0                           0                    1
##   Unique.peptides.BZ11 Unique.peptides.BZ12 Unique.peptides.BZ13
## 1                   21                   16                   16
## 2                    2                    1                    1
##   Unique.peptides.BZ14 Unique.peptides.BZ15 Unique.peptides.BZ16
## 1                   16                   14                    0
## 2                    1                    2                    0
##   Unique.peptides.BZ17 Unique.peptides.BZ18 Unique.peptides.BZ19
## 1                    1                    0                    0
## 2                    0                    0                    0
##   Unique.peptides.BZ20 Unique.peptides.BZ21 Unique.peptides.BZ6
## 1                    0                    0                  14
## 2                    0                    0                   0
##   Unique.peptides.BZ7 Unique.peptides.BZ8 Unique.peptides.BZ9
## 1                   0                   0                  15
## 2                   0                   0                   0
##   Sequence.coverage.... Unique...razor.sequence.coverage....
## 1                  31.2                                 31.2
## 2                   6.6                                  6.6
##   Unique.sequence.coverage.... Mol..weight..kDa. Sequence.length
## 1                         31.2            95.925             845
## 2                          6.6            71.289             623
##   Sequence.lengths Q.value    Score Sequence.coverage.BZ10....
## 1          845;807       0 173.4200                       19.8
## 2          623;634       0   6.3927                        2.4
##   Sequence.coverage.BZ11.... Sequence.coverage.BZ12....
## 1                       27.5                       22.7
## 2                        4.3                        2.2
##   Sequence.coverage.BZ13.... Sequence.coverage.BZ14....
## 1                       24.1                       22.6
## 2                        2.4                        2.4
##   Sequence.coverage.BZ15.... Sequence.coverage.BZ16....
## 1                       21.3                          0
## 2                        4.3                          0
##   Sequence.coverage.BZ17.... Sequence.coverage.BZ18....
## 1                        1.8                          0
## 2                        0.0                          0
##   Sequence.coverage.BZ19.... Sequence.coverage.BZ20....
## 1                          0                          0
## 2                          0                          0
##   Sequence.coverage.BZ21.... Sequence.coverage.BZ6....
## 1                          0                      22.4
## 2                          0                       0.0
##   Sequence.coverage.BZ7.... Sequence.coverage.BZ8.... Sequence.coverage.BZ9....
## 1                         0                         0                      20.9
## 2                         0                         0                       0.0
##   Intensity Intensity.BZ10 Intensity.BZ11 Intensity.BZ12 Intensity.BZ13
## 1 349190000       25728000       61873000       45660000       62547000
## 2  11625000        1719400        5244400         656040        1152500
##   Intensity.BZ14 Intensity.BZ15 Intensity.BZ16 Intensity.BZ17 Intensity.BZ18
## 1       44739000       48837000              0              0              0
## 2        1543900        1308900              0              0              0
##   Intensity.BZ19 Intensity.BZ20 Intensity.BZ21 Intensity.BZ6 Intensity.BZ7
## 1              0              0              0      28596000             0
## 2              0              0              0             0             0
##   Intensity.BZ8 Intensity.BZ9 LFQ.intensity.BZ10 LFQ.intensity.BZ11
## 1             0      31209000            2859200            3946900
## 2             0             0                  0                  0
##   LFQ.intensity.BZ12 LFQ.intensity.BZ13 LFQ.intensity.BZ14 LFQ.intensity.BZ15
## 1            3650600            2956000            2687800            3409300
## 2                  0                  0                  0                  0
##   LFQ.intensity.BZ16 LFQ.intensity.BZ17 LFQ.intensity.BZ18 LFQ.intensity.BZ19
## 1                  0                  0                  0                  0
## 2                  0                  0                  0                  0
##   LFQ.intensity.BZ20 LFQ.intensity.BZ21 LFQ.intensity.BZ6 LFQ.intensity.BZ7
## 1                  0                  0           4635600                 0
## 2                  0                  0                 0                 0
##   LFQ.intensity.BZ8 LFQ.intensity.BZ9 MS.MS.count.BZ10 MS.MS.count.BZ11
## 1                 0           3041400               30               56
## 2                 0                 0                4                8
##   MS.MS.count.BZ12 MS.MS.count.BZ13 MS.MS.count.BZ14 MS.MS.count.BZ15
## 1               35               38               32               36
## 2                1                3                2                4
##   MS.MS.count.BZ16 MS.MS.count.BZ17 MS.MS.count.BZ18 MS.MS.count.BZ19
## 1                0                1                0                0
## 2                0                0                0                0
##   MS.MS.count.BZ20 MS.MS.count.BZ21 MS.MS.count.BZ6 MS.MS.count.BZ7
## 1                0                0              27               0
## 2                0                0               0               0
##   MS.MS.count.BZ8 MS.MS.count.BZ9 MS.MS.count
## 1               0              35         290
## 2               0               0          22
##


                                                              Peptide.sequences
## 1 AANAAENDFSVSQAEMSSR;AVSEEQQPALK;DVDDDGEEKELMER;EVLEALGEVMVSR;EVLEALGEVMVSRPR;
    FAALDNEEEDKEEEIIK;GAVIVVSHDAR;GFNLPYQDAR;IGFFNQQYAEQLR;ILAGLGFDPEMQNRPTQK;
    KAEQGSEEEGEGEEEEEEGGESK;KNQDEESQEAPELLK;KTFFEELAVEDK;LQGQLEQGDDTAAER;LSVPTSDEEDEVPAPKPR;
    LTPTHGEMR;MEETPTEYLQR;NLDFGIDMDSR;NQDEESQEAPELLK;QAMLENASDIK;RLQGQLEQGDDTAAER;STLLLLLTGK;
    TFFEELAVEDK
## 2


                           EVPIPEHIDIYHLTR;FHWEQDQIAHMK;IPPPVIMVQNVSFK
##   Only.identified.by.site Reverse Potential.contaminant  id
## 1                                                       351
## 2                                                       431
##
     Peptide.IDs
## 1
    22;321;476;733;734;756;900;942;1281;1324;1453;1540;1574;1869;1915;1931;2002;2149;2194;2287;2450;2805;2910

## 2
     736;816;1373
##
          Peptide.is.razor
## 1 True;True;True;True;True;True;True;True;True;True;True;True;True;True;True;True;True;True;
    True;True;True;True;True
## 2
            True;True;True
##
                             Mod..peptide.IDs
## 1
    26;344;508;780;781;806;958;1002;1361;1406;1407;1539;1631;1665;1975;2024;2040;2124;2125;2334;2335;2387;2485;24

## 2
                                   783;870;1458
##
```

```
                                            Evidence.IDs
## 1
    96;97;1598;1599;1600;1601;1602;1603;1604;1605;2327;2328;2329;2330;3919;3920;3921;3922;3923;3924;3925;3926;3927

## 2



    3938;3939;3940;3941;3942;3943;4348;4349;4350;7842
##



                                                                  MS.MS.IDs
## 1
    258;259;3256;3257;3258;3259;3260;3261;3262;3263;3264;3265;3266;3267;3268;3269;3270;3271;3272;3273;3274;3275;32

## 2



    7902;7903;7904;7905;7906;7907;7908;7909;7910;7911;7912;7913;7914;7915;7916;7917;8772;8773;8774;8775;8776;1673

##
                                          Best.MS.MS
## 1
    259;3267;4698;7872;7889;8006;9978;10736;15218;16174;17567;18479;18902;23137;23804;23881;24510;28127;29072;303

## 2
                                     7915;8773;16733
##    Oxidation..M..site.IDs Oxidation..M..site.positions Taxonomy.IDs  name       ID
## 1         208;209;210;211              297;444;650;703        -1;-1 ABCF1 Q8NE71
## 2                                                             -1;-1 ABCF2 Q9UG63
```

2. Create a SummarizedExperiment

We used the following design to create a SummarizedExperiment.

```
#Load design matrix
design <- read.delim("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/0_Data/
    Design_matrix.txt")
design
```

```
##                 label       condition       rbp replicate crosslinking
## 1   LFQ.intensity.BZ6 HNRNPA2B1_noUV HNRNPA2B1         1         noUV
## 2   LFQ.intensity.BZ7 HNRNPA2B1_high HNRNPA2B1         1         high
## 3   LFQ.intensity.BZ8 HNRNPA2B1_high HNRNPA2B1         2         high
## 4   LFQ.intensity.BZ9  HNRNPA2B1_low HNRNPA2B1         1          low
## 5  LFQ.intensity.BZ10  HNRNPA2B1_low HNRNPA2B1         2          low
## 6  LFQ.intensity.BZ11    HNRNPU_noUV    HNRNPU         1         noUV
## 7  LFQ.intensity.BZ12    HNRNPU_high    HNRNPU         1         high
## 8  LFQ.intensity.BZ13    HNRNPU_high    HNRNPU         2         high
## 9  LFQ.intensity.BZ14     HNRNPU_low    HNRNPU         1          low
## 10 LFQ.intensity.BZ15     HNRNPU_low    HNRNPU         2          low
## 11 LFQ.intensity.BZ16    HNRNPC_noUV    HNRNPC         1         noUV
## 12 LFQ.intensity.BZ17    HNRNPC_high    HNRNPC         1         high
## 13 LFQ.intensity.BZ18    HNRNPC_high    HNRNPC         2         high
## 14 LFQ.intensity.BZ19     HNRNPC_low    HNRNPC         1          low
## 15 LFQ.intensity.BZ20     HNRNPC_low    HNRNPC         2          low
```

```
#Create a SummarizedExperiment
se <- make_se(unique_pg, columns = ecols, expdesign = design)

#Subset the SummarizedExperiment according to the proteins
HNRNPC_se <- se[,se$rbp == "HNRNPC" ]
HNRNPU_se <- se[,se$rbp == "HNRNPU" ]
HNRNPA2B1_se <- se[,se$rbp == "HNRNPA2B1" ]

#Remove noUV samples for comparing the two RNase doses
HNRNPC_se_flt2 <- HNRNPC_se[,HNRNPC_se$crosslinking != "noUV" ]
HNRNPU_se_flt2 <- HNRNPU_se[,HNRNPU_se$crosslinking != "noUV" ]
HNRNPA2B1_se_flt2 <- HNRNPA2B1_se[,HNRNPA2B1_se$crosslinking != "noUV" ]
```

3. Perform enrichment analysis using DEP

Here, we performed enrichment analysis by comparing the RNase high vs RNase low using the DEP2 R package (PMID: 37624922).

```
set.seed(3)
#Function to calculate the differential expression
DEP_analysis <- function(se, a) {
  filt <- filter_se(se,thr = 0, filter_formula = ~ Reverse != '+' & Potential.contaminant !="+" &
     Peptides > 1 & Unique.peptides > 0)
  norm <- normalize_vsn(filt)
  imputed <- DEP2::impute(norm,fun = "QRILC")
  diff <- test_diff(imputed, type = "control", control = a, fdr.type = "BH")
  dep <- add_rejections(diff, alpha = 1, lfc = 0)
  results <- get_results(dep)
  return(list(se = se, filt = filt,
              norm = norm, imputed = imputed, diff = diff, dep = dep,
                results = results))
}

#Run DEP analysis
HNRNPC_se_DE <- DEP_analysis(se =  HNRNPC_se_flt2,a = "HNRNPC_low")
HNRNPU_se_DE <- DEP_analysis(se =  HNRNPU_se_flt2,a = "HNRNPU_low")
HNRNPA2B1_se_DE <- DEP_analysis(se =  HNRNPA2B1_se_flt2,a = "HNRNPA2B1_low")

head(HNRNPC_se_DE$results)
```

```
##     name        ID HNRNPC_high_vs_HNRNPC_low_p.val
## 1    DCD    P81605                      0.53394372
## 2   DDX1 Q92499-3                      0.06815325
## 3 DDX17    Q92841                      0.37545198
## 4 DDX3X O00571-2                      0.10961081
## 5   DDX5    P17844                      0.16397844
## 6   DHX9    Q08211                      0.18833307
##   HNRNPC_high_vs_HNRNPC_low_p.adj significant
## 1                           0.571        TRUE
## 2                           0.103        TRUE
## 3                           0.421        TRUE
## 4                           0.158        TRUE
## 5                           0.210        TRUE
## 6                           0.234        TRUE
##   HNRNPC_high_vs_HNRNPC_low_significant HNRNPC_high_vs_HNRNPC_low_ratio
## 1                                  TRUE                           1.230
## 2                                  TRUE                          -3.370
## 3                                  TRUE                           0.220
## 4                                  TRUE                           0.640
## 5                                  TRUE                           0.608
## 6                                  TRUE                          -0.512
##   HNRNPC_high_centered HNRNPC_low_centered
## 1                0.617              -0.617
## 2               -1.690               1.690
## 3                0.110              -0.110
## 4                0.320              -0.320
## 5                0.304              -0.304
## 6               -0.256               0.256
```

```
#Save the LFQ intensities
write.table(as.data.frame(HNRNPC_se_DE$dep@assays@data@listData), file = "~/Documents/Postdoc/PD_
    Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/2_DEP/HNRNPC_se_DE_LFQ_intensity.txt", row.names
    = TRUE, sep = "\t", quote = F)
write.table(as.data.frame(HNRNPU_se_DE$dep@assays@data@listData), file = "~/Documents/Postdoc/PD_
    Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/2_DEP/HNRNPU_se_DE_LFQ_intensity.txt", row.names
    = TRUE, sep = "\t", quote = F)
write.table(as.data.frame(HNRNPA2B1_se_DE$dep@assays@data@listData), file = "~/Documents/Postdoc/
    PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/2_DEP/HNRNPA2B1_se_DE_LFQ_intensity.txt", row.
    names = TRUE, sep = "\t", quote = F)

write.table(HNRNPC_se_DE$results, file = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_
    HighLow_293T/2_DEP/HNRNPC_se_norm_QRILC_res_LFQ_intensity.txt", row.names = FALSE, sep = "\t"
    , quote = F)
write.table(HNRNPU_se_DE$results, file = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_
    HighLow_293T/2_DEP/HNRNPU_se_norm_QRILC_res_LFQ_intensity.txt", row.names = FALSE, sep = "\t"
    , quote = F)
write.table(HNRNPA2B1_se_DE$results, file = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/
    RNase_HighLow_293T/2_DEP/HNRNPA2B1_se_norm_QRILC_res_LFQ_intensity.txt", row.names = FALSE,
    sep = "\t", quote = F)
```

Significant reduced proteins were selected as proteins having an FDR < 0.05 and a logFC vs RNase low (0.02U/µL) dose < 0.

4. Perform FDR analysis using Clipper

We used Clipper (PMID:34635147) function to compare noUV (1 replicate) and UVC samples (2 replicates for each RNase dose).

```
#Impute data with noUV
HNRNPC_se_flt <- HNRNPC_se[rownames(HNRNPC_se_DE$filt)]
write.table(as.data.frame(HNRNPC_se@assays@data@listData), file = "~/Documents/Postdoc/PD_
     Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/2_DEP/HNRNPC_se_LFQ_intensity_raw.txt", row.names
     = TRUE, sep = "\t", quote = F)
HNRNPC_se_flt <- DEP2::impute(HNRNPC_se_flt,fun = "QRILC")
HNRNPU_se_flt <- HNRNPU_se[rownames(HNRNPU_se_DE$filt)]
write.table(as.data.frame(HNRNPU_se@assays@data@listData), file = "~/Documents/Postdoc/PD_
     Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/2_DEP/HNRNPU_se_LFQ_intensity_raw.txt", row.names
     = TRUE, sep = "\t", quote = F)
HNRNPU_se_flt <- DEP2::impute(HNRNPU_se_flt,fun = "QRILC")
HNRNPA2B1_se_flt <- HNRNPA2B1_se[rownames(HNRNPA2B1_se_DE$filt)]
write.table(as.data.frame(HNRNPA2B1_se@assays@data@listData), file = "~/Documents/Postdoc/PD_
     Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/2_DEP/HNRNPA2B1_se_LFQ_intensity_raw.txt", row.
     names = TRUE, sep = "\t", quote = F)
HNRNPA2B1_se_flt <- DEP2::impute(HNRNPA2B1_se_flt,fun = "QRILC")

#Prepare data for clipper
HNRNPC_data <- as.data.frame(HNRNPC_se_flt@assays@data@listData)
HNRNPU_data <- as.data.frame(HNRNPU_se_flt@assays@data@listData)
HNRNPA2B1_data <- as.data.frame(HNRNPA2B1_se_flt@assays@data@listData)

#Run Clipper on low and high samples
HNRNPC_clipper_high = Clipper(score.exp = as.matrix(HNRNPC_data[,c(2,3)]), score.back = as.matrix
     (HNRNPC_data[,-c(2,3,4,5)]), FDR = 0.1, analysis = "e")
HNRNPC_clipper_low = Clipper(score.exp = as.matrix(HNRNPC_data[,c(4,5)]), score.back = as.matrix(
     HNRNPC_data[,-c(2,3,4,5)]), FDR = 0.1, analysis = "e")

HNRNPU_clipper_high = Clipper(score.exp = as.matrix(HNRNPU_data[,c(2,3)]), score.back = as.matrix
     (HNRNPU_data[,-c(2,3,4,5)]), FDR = 0.1, analysis = "e")
HNRNPU_clipper_low = Clipper(score.exp = as.matrix(HNRNPU_data[,c(4,5)]), score.back = as.matrix(
     HNRNPU_data[,-c(2,3,4,5)]), FDR = 0.1, analysis = "e")

HNRNPA2B1_clipper_high = Clipper(score.exp = as.matrix(HNRNPA2B1_data[,c(2,3)]), score.back = as.
     matrix(HNRNPA2B1_data[,-c(2,3,4,5)]), FDR = 0.1, analysis = "e")
HNRNPA2B1_clipper_low = Clipper(score.exp = as.matrix(HNRNPA2B1_data[,c(4,5)]), score.back = as.
     matrix(HNRNPA2B1_data[,-c(2,3,4,5)]), FDR = 0.1, analysis = "e")

#Attach results to data
HNRNPC_data$FDR_high <- HNRNPC_clipper_high$q
HNRNPU_data$FDR_high <- HNRNPU_clipper_high$q
HNRNPA2B1_data$FDR_high <- HNRNPA2B1_clipper_high$q

HNRNPC_data$FDR_low <- HNRNPC_clipper_low$q
HNRNPU_data$FDR_low <- HNRNPU_clipper_low$q
HNRNPA2B1_data$FDR_low <- HNRNPA2B1_clipper_low$q

#Calculate logFC
HNRNPC_data$logFC_high <- rowMeans(HNRNPC_data[,c(2,3)])-HNRNPC_data$HNRNPC_noUV_1
HNRNPC_data$logFC_low <- rowMeans(HNRNPC_data[,c(4,5)])-HNRNPC_data$HNRNPC_noUV_1

HNRNPU_data$logFC_high <- rowMeans(HNRNPU_data[,c(2,3)])-HNRNPU_data$HNRNPU_noUV_1
HNRNPU_data$logFC_low <- rowMeans(HNRNPU_data[,c(4,5)])-HNRNPU_data$HNRNPU_noUV_1

HNRNPA2B1_data$logFC_high <- rowMeans(HNRNPA2B1_data[,c(2,3)])-HNRNPA2B1_data$HNRNPA2B1_noUV_1
HNRNPA2B1_data$logFC_low <- rowMeans(HNRNPA2B1_data[,c(4,5)])-HNRNPA2B1_data$HNRNPA2B1_noUV_1
```

```
head(HNRNPC_data)
```

```
##        HNRNPC_noUV_1 HNRNPC_high_1 HNRNPC_high_2 HNRNPC_low_1 HNRNPC_low_2
## DCD         23.22081      23.00590      22.68121     24.15777     18.66839
## DDX1        20.53089      18.51677      17.78069     21.02425     21.10534
## DDX17       12.26102      23.08323      22.96714     22.99967     23.18721
## DDX3X       15.39217      19.04855      19.43683     19.86245     20.17217
## DDX5        13.12702      21.05768      20.41967     20.55309     20.28444
## DHX9        15.48094      21.39477      20.72251     21.82287     21.89360
##          FDR_high    FDR_low logFC_high  logFC_low
## DCD   1.00000000 0.02222222 -0.3772564 -1.8077345
## DDX1  1.00000000 0.02222222 -2.3821578  0.5339063
## DDX17 0.02439024 0.02222222 10.7641615 10.8324139
## DDX3X 0.02439024 0.02222222  3.8505197  4.6251393
## DDX5  0.02439024 0.02222222  7.6116546  7.2917438
## DHX9  0.02439024 0.02222222  5.5777032  6.3772984
```

```
#Save the clipper results
write.table(HNRNPC_data, file = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_
    293T/2_DEP/HNRNPC_clipper_results.txt", row.names = TRUE, sep = "\t", quote = F)
write.table(HNRNPU_data, file = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_
    293T/2_DEP/HNRNPU_clipper_results.txt", row.names = TRUE, sep = "\t", quote = F)
write.table(HNRNPA2B1_data, file = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow
    _293T/2_DEP/HNRNPA2B1_clipper_results.txt", row.names = TRUE, sep = "\t", quote = F)
```

5. Visualization of the results

For the visualization, we first filter the DEP results for UVC-enriched proteins (a.k.a RDAPs). Only proteins with an FDR < 0.1 and a FC vs noUV > 3 in at least one RNase condition were categorized as RDAPs.

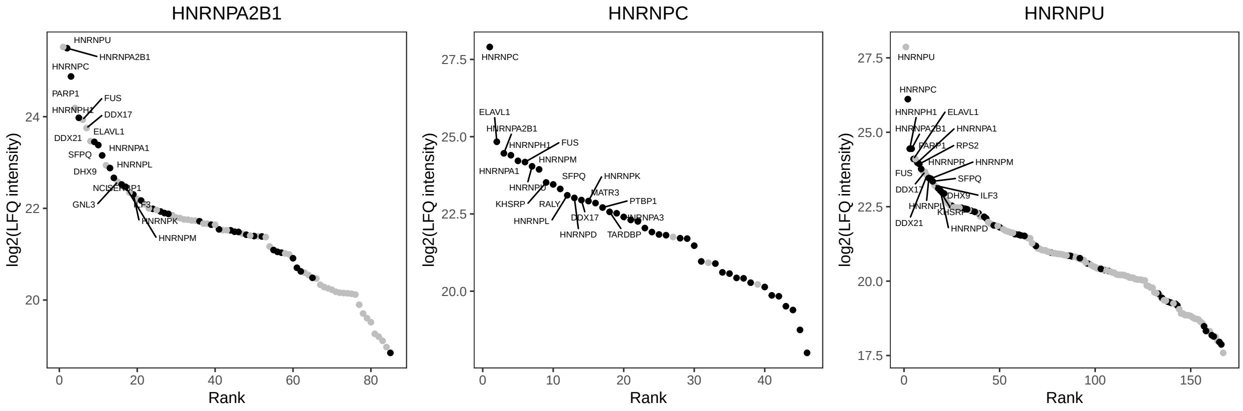
```
#Store DEP results
HNRNPC <- HNRNPC_se_DE$results
HNRNPU <- HNRNPU_se_DE$results
HNRNPA2B1 <- HNRNPA2B1_se_DE$results

#Get the significant UVC-enriched proteins
HNRNPC_data_sub <- subset(HNRNPC_data, (HNRNPC_data$logFC_low > log2(3) & HNRNPC_data$FDR_low <
    0.1) | (HNRNPC_data$logFC_high > log2(3) & HNRNPC_data$FDR_high < 0.1))
HNRNPU_data_sub <- subset(HNRNPU_data, (HNRNPU_data$logFC_low > log2(3) & HNRNPU_data$FDR_low <
    0.1) | (HNRNPU_data$logFC_high > log2(3) & HNRNPU_data$FDR_high < 0.1))
HNRNPA2B1_data_sub <- subset(HNRNPA2B1_data, (HNRNPA2B1_data$logFC_low > log2(3) & HNRNPA2B1_data
    $FDR_low < 0.1) | (HNRNPA2B1_data$logFC_high > log2(3) & HNRNPA2B1_data$FDR_high < 0.1))

#Keep only UVC-enriched proteins
HNRNPC <- subset(HNRNPC, HNRNPC$name %in% rownames(HNRNPC_data_sub))
HNRNPU <- subset(HNRNPU, HNRNPU$name %in% rownames(HNRNPU_data_sub))
HNRNPA2B1 <- subset(HNRNPA2B1, HNRNPA2B1$name %in% rownames(HNRNPA2B1_data_sub))

#Determine the significant proteins
HNRNPC$sign <- HNRNPC$HNRNPC_high_vs_HNRNPC_low_p.adj < 0.05
HNRNPU$sign <- HNRNPU$HNRNPU_high_vs_HNRNPU_low_p.adj < 0.05
HNRNPA2B1$sign <- HNRNPA2B1$HNRNPA2B1_high_vs_HNRNPA2B1_low_p.adj < 0.05

head(HNRNPC, n = 2)
```

```
##     name          ID HNRNPC_high_vs_HNRNPC_low_p.val
## 3 DDX17      Q92841                       0.3754520
## 4 DDX3X  O00571-2                         0.1096108
##   HNRNPC_high_vs_HNRNPC_low_p.adj significant
## 3                           0.421        TRUE
## 4                           0.158        TRUE
##   HNRNPC_high_vs_HNRNPC_low_significant HNRNPC_high_vs_HNRNPC_low_ratio
## 3                                  TRUE                            0.22
## 4                                  TRUE                            0.64
##   HNRNPC_high_centered HNRNPC_low_centered  sign
## 3                 0.11               -0.11 FALSE
## 4                 0.32               -0.32 FALSE
```

Rank plot of intensities

As a first visualization, we generated rank plots based on the imputed LFQ intensities for the RNAse low samples and highlighted the top 20 UVC-enriched proteins with the highest intensity for all the tested RBPs. Black dots: RDAPs.

```
#Get the ranking of the proteins
HNRNPC_norm <- as.data.frame(HNRNPC_se_DE$dep@assays@data@listData)
HNRNPC_norm$name <- rownames(HNRNPC_norm)
HNRNPC_norm$low_avg <- rowMeans(HNRNPC_norm[,3:4])
HNRNPC_norm$UVC <- HNRNPC_norm$name %in% HNRNPC$name

HNRNPU_norm <- as.data.frame(HNRNPU_se_DE$dep@assays@data@listData)
HNRNPU_norm$name <- rownames(HNRNPU_norm)
HNRNPU_norm$low_avg <- rowMeans(HNRNPU_norm[,3:4])
HNRNPU_norm$UVC <- HNRNPU_norm$name %in% HNRNPU$name

HNRNPA2B1_norm <- as.data.frame(HNRNPA2B1_se_DE$dep@assays@data@listData)
HNRNPA2B1_norm$name <- rownames(HNRNPA2B1_norm)
HNRNPA2B1_norm$low_avg <- rowMeans(HNRNPA2B1_norm[,3:4])
HNRNPA2B1_norm$UVC <- HNRNPA2B1_norm$name %in% HNRNPA2B1$name

HNRNPC_norm$Rank = rank(-HNRNPC_norm$low_avg)
HNRNPU_norm$Rank = rank(-HNRNPU_norm$low_avg)
HNRNPA2B1_norm$Rank = rank(-HNRNPA2B1_norm$low_avg)

 options(ggrepel.max.overlaps = Inf)

HNRNPC_plot <- ggplot(HNRNPC_norm, aes(x=Rank, y=low_avg, label = name)) +
  geom_point(aes(col = UVC)) +
  scale_color_manual(values = c(‘TRUE‘ = "black", ‘FALSE‘ = "grey")) +
  ggrepel::geom_text_repel(data = subset(HNRNPC_norm, Rank < 20), size = 2) +
    theme_bw() +
  ggtitle("HNRNPC") +
  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)) +
  xlab("Rank") +
  ylab("log2(LFQ intensity)")

HNRNPU_plot <- ggplot(HNRNPU_norm, aes(x=Rank, y=low_avg, label = name)) +
  geom_point(aes(col = UVC)) +
  scale_color_manual(values = c(‘TRUE‘ = "black", ‘FALSE‘ = "grey")) +
  ggrepel::geom_text_repel(data = subset(HNRNPU_norm, Rank < 20), size = 2) +
    theme_bw() +
  ggtitle("HNRNPU") +
  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)) +
  xlab(”Rank”) +
  ylab(”log2(LFQ intensity)”)

HNRNPA2B1_plot <- ggplot(HNRNPA2B1_norm, aes(x=Rank, y=low_avg, label = name)) +
  geom_point(aes(col = UVC)) +
  scale_color_manual(values = c(‘TRUE‘ = ”black”, ‘FALSE‘ = ”grey”)) +
  ggrepel::geom_text_repel(data = subset(HNRNPA2B1_norm, Rank < 20), size = 2) +
    theme_bw() +
  ggtitle(”HNRNPA2B1”) +
  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)) +
  xlab(”Rank”) +
  ylab(”log2(LFQ intensity)”)


ggarrange(HNRNPA2B1_plot, HNRNPC_plot, HNRNPU_plot, ncol = 3)
```

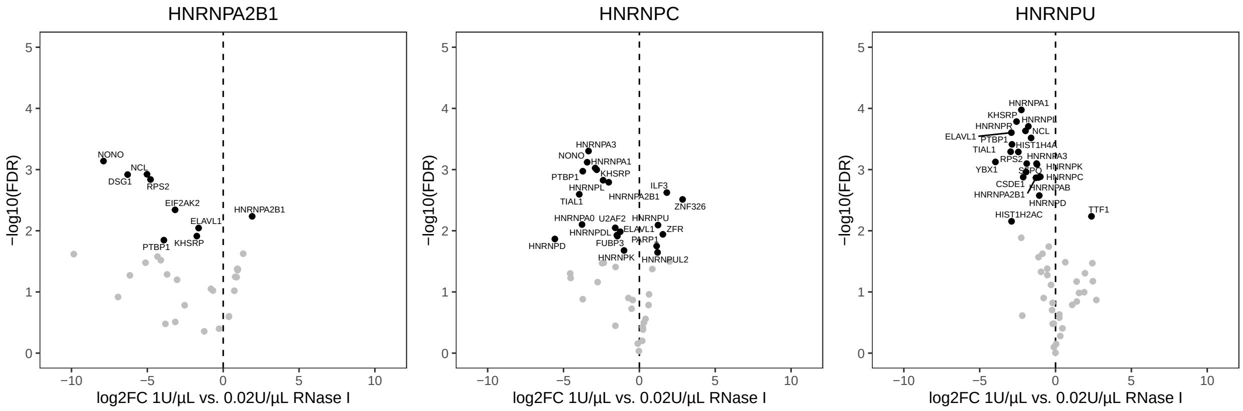


```
#Save the plot as pdf
pdf(”~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/3_DEP_visualization/Rank_
    plot.pdf”, height = 4, width = 12)
ggarrange(HNRNPA2B1_plot, HNRNPC_plot, HNRNPU_plot, ncol = 3)
dev.off()
```

Volcano plots

Next, we showed the DE results as a typical volcano plot by having the fold change on the x axis and the FDR on the y axis. Black dots: significant proteins with FDR < 0.05 between high and low RNase doses.

```
#Create a volcano plot
HNRNPC.ggplot <- ggplot(data=HNRNPC, aes(x=HNRNPC_high_vs_HNRNPC_low_ratio, y=-log10(HNRNPC_high_
    vs_HNRNPC_low_p.val))) + geom_vline(xintercept = 0, linetype = ”dashed”) +
  geom_point(aes(col = sign)) +
  scale_color_manual(values = c(‘TRUE‘ = ”black”, ‘FALSE‘ = ”grey”)) +
    ggrepel::geom_text_repel(data = filter(HNRNPC, sign), aes(label = name), size = 2, box.
    padding = unit(0.1, ”lines”), point.padding = unit(0.1, ”lines”), segment.size = 0.5,max.
    overlaps = Inf) +
  theme_bw() +
  ggtitle("HNRNPC") +
  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)) +
  xlab("log2FC 1U/µL vs. 0.02U/µL RNase I") +
  ylab("-log10(FDR)") +
  xlim(-11, 11) +
  ylim(0, 5)

HNRNPU.ggplot <- ggplot(data=HNRNPU, aes(x=HNRNPU_high_vs_HNRNPU_low_ratio, y=-log10(HNRNPU_high_
    vs_HNRNPU_low_p.val))) + geom_vline(xintercept = 0, linetype = "dashed") +
  geom_point(aes(col = sign)) +
  scale_color_manual(values = c(`TRUE` = "black", `FALSE` = "grey")) +
    ggrepel::geom_text_repel(data = filter(HNRNPU, sign), aes(label = name), size = 2, box.
    padding = unit(0.1, "lines"), point.padding = unit(0.1, "lines"), segment.size = 0.5,max.
    overlaps = Inf) +
  theme_bw() +
  ggtitle("HNRNPU") +
  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)) +
  xlab("log2FC 1U/µL vs. 0.02U/µL RNase I") +
  ylab("-log10(FDR)") +
  xlim(-11, 11) +
  ylim(0, 5)

HNRNPA2B1.ggplot <- ggplot(data=HNRNPA2B1, aes(x=HNRNPA2B1_high_vs_HNRNPA2B1_low_ratio, y=-log10(
    HNRNPA2B1_high_vs_HNRNPA2B1_low_p.val))) + geom_vline(xintercept = 0, linetype = "dashed") +
  geom_point(aes(col = sign)) +
  scale_color_manual(values = c(`TRUE` = "black", `FALSE` = "grey")) +
  ggrepel::geom_text_repel(data = filter(HNRNPA2B1, sign), aes(label = name), size = 2, box.
    padding = unit(0.1, "lines"), point.padding = unit(0.1, "lines"), segment.size = 0.5,max.
    overlaps = Inf) +
  theme_bw() +
  ggtitle("HNRNPA2B1") +
  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)) +
  xlab("log2FC 1U/µL vs. 0.02U/µL RNase I") +
  ylab("-log10(FDR)") +
  xlim(-11, 11) +
  ylim(0, 5)

ggarrange(HNRNPA2B1.ggplot, HNRNPC.ggplot, HNRNPU.ggplot, ncol = 3)
```



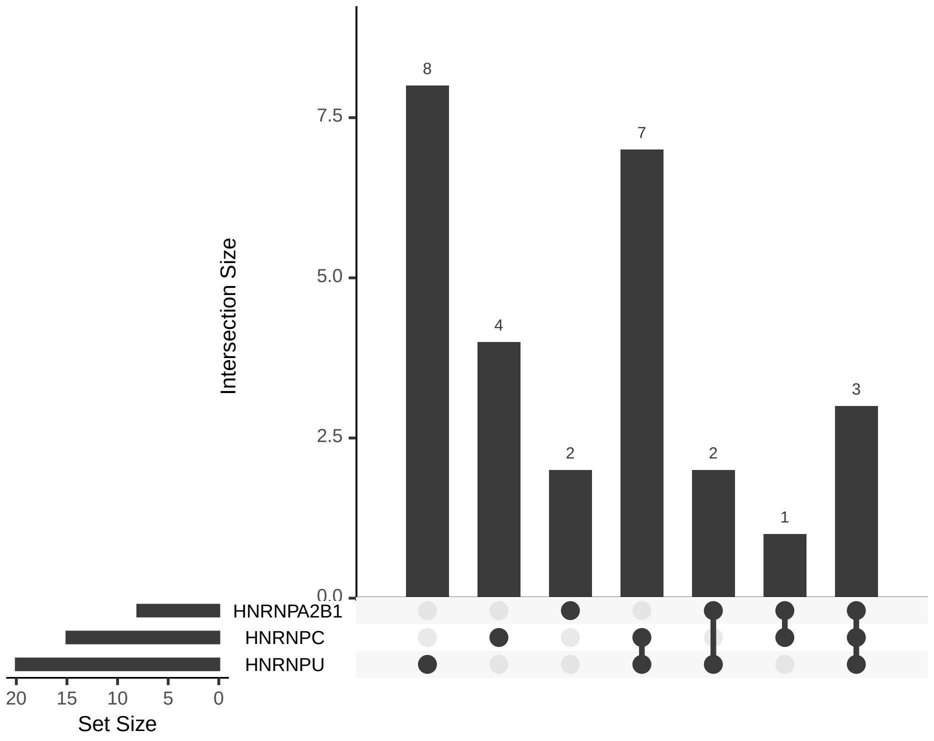
```
#Save the plot as pdf
pdf("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/3_DEP_visualization/
     Volcano_plot.pdf", height = 4, width = 12)
ggarrange(HNRNPA2B1.ggplot, HNRNPC.ggplot, HNRNPU.ggplot, ncol = 3)
dev.off()
```

Upset plot

Here, we compared the overlap of the RDAPs with significant reduction after 1U/µL between the RBP tested using upset plot.

```
#Subset significant proteins
HNRNPC_flt <- subset(HNRNPC, HNRNPC_high_vs_HNRNPC_low_p.adj < 0.05 & HNRNPC_high_vs_HNRNPC_low_
     ratio < 0)
HNRNPA2B1_flt <- subset(HNRNPA2B1, HNRNPA2B1_high_vs_HNRNPA2B1_low_p.adj < 0.05 & HNRNPA2B1_high_
     vs_HNRNPA2B1_low_ratio < 0)
HNRNPU_flt <- subset(HNRNPU, HNRNPU_high_vs_HNRNPU_low_p.adj < 0.05 & HNRNPU_high_vs_HNRNPU_low_
     ratio < 0)

#Make upset plot of detected proteins in noUV samples
lt.tsk = list(HNRNPC = HNRNPC_flt$name,
              HNRNPU = HNRNPU_flt$name,
              HNRNPA2B1 = HNRNPA2B1_flt$name)

upsetPlot <- upset(fromList(lt.tsk),
      sets = c("HNRNPU","HNRNPC","HNRNPA2B1"),
      mb.ratio = c(0.8, 0.2),
      number.angles = 0,
      text.scale = 1,
      point.size = 3,
      line.size = 1,
      keep.order = TRUE,
      # empty.intersections = "on"
      # queries = list(list(query = intersects, params = list("tss.total"), color = "red", active
     = T))
)
```

```
upsetPlot
```



```
#Save the upsetplot as pdf
pdf("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/3_DEP_visualization/
    Upsetplot_signprot.pdf", width = 3.5, height = 5)
upsetPlot
dev.off()
```

Arc diagram

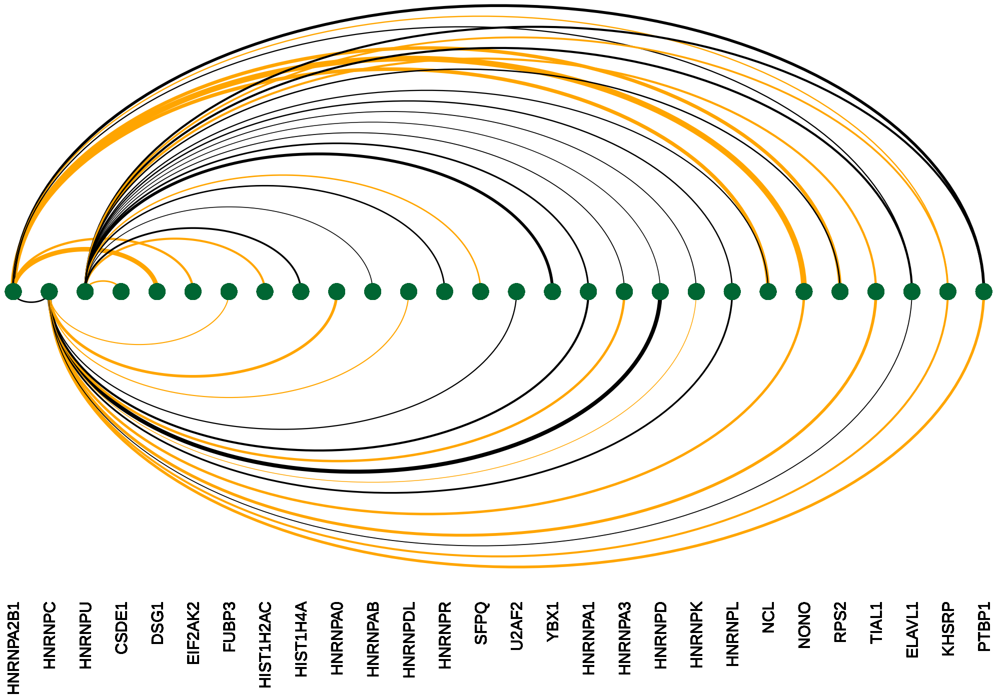
We then used these information to generated a binary table of the interaction that we will use for generating the arc diagram.

```
#Function to create the binary table
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)
}

#Binary table with colnames:
sign.proteins <- fromList(lt.tsk)
write.table(sign.proteins, file = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_
    293T/3_DEP_visualization/Upsetplot_signprot.txt", row.names = TRUE, sep = "\t", quote = F)

head(sign.proteins)
```

```
##              HNRNPC HNRNPU HNRNPA2B1
## ELAVL1            1      1         1
## FUBP3             1      0         0
## HNRNPA0           1      0         0
## HNRNPA1           1      1         0
## HNRNPA2B1         1      1         0
## HNRNPA3           1      1         0
```

```
#Prepare data
data_int <- data.frame(to = c(HNRNPC_flt$name, HNRNPU_flt$name, HNRNPA2B1_flt$name),
                       ratio = c(HNRNPC_flt$HNRNPC_high_vs_HNRNPC_low_ratio, HNRNPU_flt$HNRNPU_
    high_vs_HNRNPU_low_ratio, HNRNPA2B1_flt$HNRNPA2B1_high_vs_HNRNPA2B1_low_ratio))
data_int$from <- c(rep("HNRNPC", length(HNRNPC_flt$name)), rep("HNRNPU", length(HNRNPU_flt$name))
    , rep("HNRNPA2B1", length(HNRNPA2B1_flt$name)))
connect <- data_int[,c(3,1:2)]
connect <- connect %>% group_by(from) %>% arrange(ratio, .by_group=TRUE)

mygraph <- graph_from_data_frame( connect, directed = FALSE )

#Number of connection per RBP
c( as.character(connect$from), as.character(connect$to)) %>%
  as.tibble() %>%
  group_by(value) %>%
  summarize(n=n()) -> coauth
colnames(coauth) <- c("name", "n")

#Add grouping
coauth2 <- coauth[order(coauth$n, decreasing = TRUE),]
coauth2$grp <- c( rep(1, 3), rep( 4 , length(coauth2$n[coauth2$n == 3])), rep( 3 , length(coauth2
    $n[coauth2$n == 2])), rep( 2 , length(coauth2$n[coauth2$n == 1])))
coauth3 <- coauth2[order(coauth2$grp, decreasing = FALSE),]

#Generate a arcplot compatible object
star_edges = get.edgelist(mygraph)

#Load BioGRID interactions
biogrid <- read.delim("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/0_Data/
    BIOGRID/RBP_biogrid_interactors.txt",sep="\t", header=TRUE)
biogrid <- data.frame("a" = biogrid$Official.Symbol.Interactor.A, "b" = biogrid$Official.Symbol.
    Interactor.B)

#Compare the signifcant proteins to BioGrid annotation
hnC <- biogrid == "HNRNPC"
hnC <- data.frame("a"=biogrid$a,"b"=biogrid$b,"c"=paste(hnC[,1],hnC[,2], sep="_"))
hnC.1 <- subset(hnC, c == "TRUE_FALSE")
hnC.2 <- subset(hnC, c == "FALSE_TRUE")
hnC <- rbind(hnC.1,hnC.2)
hnC.a <- HNRNPC_flt$name %in% hnC$a
hnC.b <- HNRNPC_flt$name %in% hnC$b
hnC.int <- data.frame("gene" = HNRNPC_flt$name,"biogrid" = paste(hnC.a,hnC.b, sep = "_"))
hnC.int$rbp <- "HNRNPC"

hna2b1 <- biogrid == "HNRNPA2B1"
hna2b1 <- data.frame("a"=biogrid$a,"b"=biogrid$b,"c"=paste(hna2b1[,1],hna2b1[,2], sep="_"))
hna2b1.1 <- subset(hna2b1, c == "TRUE_FALSE")
hna2b1.2 <- subset(hna2b1, c == "FALSE_TRUE")
hna2b1 <- rbind(hna2b1.1,hna2b1.2)
hna2b1.a <- HNRNPA2B1_flt$name %in% hna2b1$a
hna2b1.b <- HNRNPA2B1_flt$name %in% hna2b1$b
hna2b1.int <- data.frame("gene" = HNRNPA2B1_flt$name,"biogrid" = paste(hna2b1.a,hna2b1.b, sep = "
    _"))
```

```
hna2b1.int$rbp <- "HNRNPA2B1"

hnU <- biogrid == "HNRNPU"
hnU <- data.frame("a"=biogrid$a,"b"=biogrid$b,"c"=paste(hnU[,1],hnU[,2], sep="_"))
hnU.1 <- subset(hnU, c == "TRUE_FALSE")
hnU.2 <- subset(hnU, c == "FALSE_TRUE")
hnU <- rbind(hnU.1,hnU.2)
hnU.a <- HNRNPU_flt$name %in% hnU$a
hnU.b <- HNRNPU_flt$name %in% hnU$b
hnU.int <- data.frame("gene" = HNRNPU_flt$name,"biogrid" = paste(hnU.a,hnU.b, sep = "_"))
hnU.int$rbp <- "HNRNPU"

#Combine the BioGRID results and prepare data for arcplot
all <- rbind(hnC.int,hna2b1.int,hnU.int)
write.table(all, file = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/3_DEP
    _visualization/RNaseHL_BioGRID_interactions.txt", sep = "\t", row.names = FALSE, quote = F)

all$int <- paste(all$gene, all$rbp, sep = "_")
edges <- as.data.frame(star_edges)
edges$int <- paste(edges$V2, edges$V1, sep = "_")
all$int[grep("HNRNPA2B1_HNRNPC", all$int)] <- "HNRNPC_HNRNPA2B1"

idx <- match(all$int, edges$int)

all <- all[match(edges$int, all$int),]
all$color <- ifelse(all$biogrid == "FALSE_FALSE", "orange", "black")

#Do the arcplot
arcplot(star_edges, show.nodes = TRUE, show.labels = TRUE, ordering=coauth3$name[c(3,2,1,4:length
    (coauth3$name))],
        lwd.arcs=(connect$ratio*-1)/1.5, col.arcs=all$color,
        col.nodes = "#006633", cex.nodes=2, line=-0.5,col.labels = "black", ylim = c(-0.5,0.5),
    above = c(1:length(HNRNPA2B1_flt$name),(length(HNRNPA2B1_flt$name)+length(HNRNPC_flt$name)+1)
    :length(star_edges[,1])))
```



Nodes: significant proteins; edges: significant reduced association with bait; edge width: logFC after DEP analysis; orange edge: novel putative associations.

```
#Save the arc plot as PDF
pdf("~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/3_DEP_visualization/
    Arcplot_signprot.pdf", width = 10, height = 5)
arcplot(star_edges, show.nodes = TRUE, show.labels = TRUE, ordering=coauth3$name[c(1,3,2,4:length
    (coauth3$name))],
        lwd.arcs=(connect$ratio*-1)/1.5, col.arcs=all$color,
        col.nodes = "#006633", cex.nodes=2, line=-0.5,col.labels = "black", ylim = c(-0.5,0.5),
    above = c(1:length(HNRNPA2B1_flt$name),(length(HNRNPA2B1_flt$name)+length(HNRNPC_flt$name)+1)
    :length(star_edges[,1])))
dev.off()
```

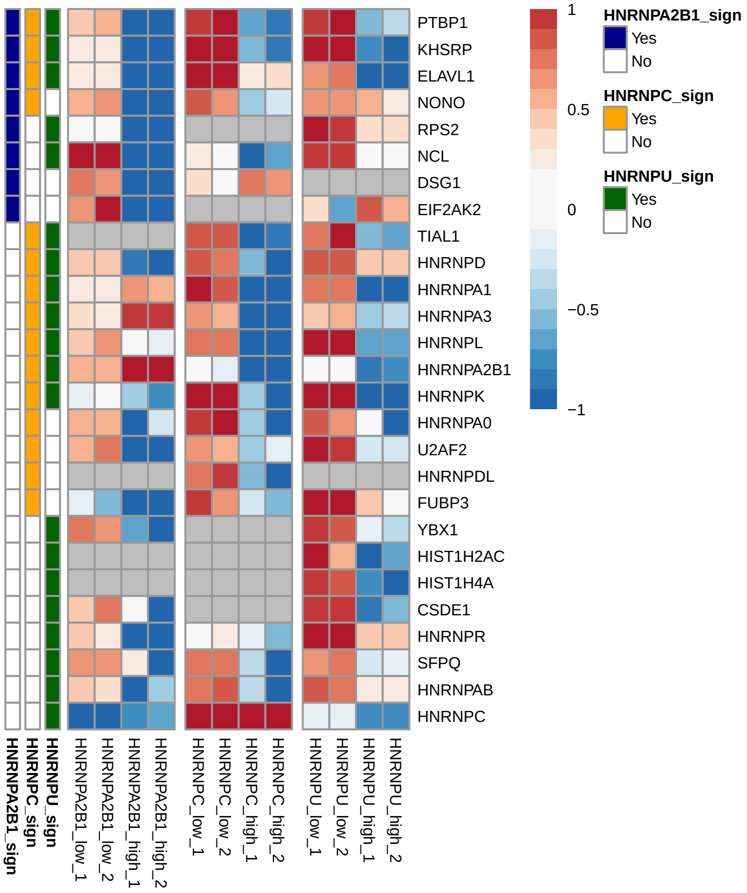
Heatmap

We also generated heatmap of imputed LFQ intensities for the RDAPs significantly reduced in at least one RBP tested.

```
#Load results from imputed LFQ intensities used in DEP analysis
HNRNPC_LFQ <- as.data.frame(HNRNPC_se_DE$dep@assays@data@listData)
HNRNPU_LFQ <- as.data.frame(HNRNPU_se_DE$dep@assays@data@listData)
HNRNPA2B1_LFQ <- as.data.frame(HNRNPA2B1_se_DE$dep@assays@data@listData)

head(HNRNPC_LFQ)
```

```
##         HNRNPC_high_1 HNRNPC_high_2 HNRNPC_low_1 HNRNPC_low_2
## DCD           23.10995      22.86497      24.10267      19.40497
## DDX1          19.23464      15.86573      20.96915      20.87263
## DDX17         23.18728      23.15090      22.94457      22.95450
## DDX3X         19.15259      19.62059      18.60218      18.89130
## DDX5          21.16173      20.60343      20.49799      20.05173
## DHX9          21.49882      20.90627      21.76777      21.66089
```

```
#Get the significant proteins
sign_prot <- unique(c(HNRNPC_flt$name, HNRNPU_flt$name, HNRNPA2B1_flt$name))

rbp_fc <- merge(HNRNPC_flt[,c(1,7)], HNRNPU_flt[,c(1,7)], by = "name", all.x =TRUE, all.y = TRUE)
rbp_fc <- merge(rbp_fc, HNRNPA2B1_flt[,c(1,7)], by = "name", all.x =TRUE, all.y = TRUE)
rbp_fc <- rbp_fc %>% replace(is.na(.), 0)
rbp_fc$avglogfc <- rowMeans(rbp_fc[,2:4])


#Merge the LFQ intensities of RBP tested
all.LFQ <- merge(HNRNPA2B1_LFQ[,c(3,4,1,2)], HNRNPC_LFQ[,c(3,4,1,2)], by = "row.names", all =
     TRUE)
rownames(all.LFQ) <- all.LFQ$Row.names
all.LFQ <- merge(all.LFQ[-1], HNRNPU_LFQ[,c(3,4,1,2)], by = "row.names", all = TRUE)
rownames(all.LFQ) <- all.LFQ$Row.names
all.LFQ <- all.LFQ[-1]
all.LFQ <- all.LFQ[sign_prot,]

#Color scale to be used in the pheatmap
my.breaks <- c(seq(-1, -0.01, by=0.1), seq(0.1, 1, by=0.1))
my.colors <- c(colorRampPalette(colors = c("#2166AC", "#4393C3", "#92C5DE", "#D1E5F0", "#F7F7F7")
     )(length(my.breaks)/2), colorRampPalette(colors = c("#F7F7F7", "#FDDBC7", "#F4A582", "#D6604D
     ", "#B2182B"))(length(my.breaks)/2))

#Annotation about significance
annotation <- as.data.frame(sign.proteins)
annotation$name <- rownames(annotation)
annotation <- merge(annotation, rbp_fc[,c(1,5)], by = "name")
rownames(annotation) <- annotation$name
annotation <- annotation[,c(3,2,4,5)]
annotation <- annotation %>% arrange(-HNRNPA2B1, -HNRNPC, -HNRNPU, avglogfc)
annotation[annotation == 1] <- "Yes"
annotation[annotation == 0] <- "No"
annotation$HNRNPU <- as.factor(annotation$HNRNPU)
annotation$HNRNPC <- as.factor(annotation$HNRNPC)
annotation$HNRNPA2B1 <- as.factor(annotation$HNRNPA2B1)
annotation <- annotation[,-c(4)]
colnames(annotation) <- c("HNRNPU_sign", "HNRNPC_sign", "HNRNPA2B1_sign")

ann_colors = list(HNRNPU_sign = c("Yes"="darkgreen", "No"="white"), HNRNPC_sign = c("Yes"="orange
     ", "No"="white"), HNRNPA2B1_sign = c("Yes"="darkblue", "No"="white"))

all <- all.LFQ[match(rownames(annotation), rownames(all.LFQ)),]

#Generate the heatmap
pheatmap <- pheatmap(
  mat                = all,
  annotation_row = annotation,
  annotation_colors = ann_colors,
  cellheight=10,
  cellwidth = 10,
  na_col = "grey",
  color = my.colors,
  breaks = my.breaks,
  show_colnames      = TRUE,
  show_rownames      = TRUE,
  drop_levels        = TRUE,
  fontsize           = 5.5,
  cluster_rows       = FALSE,
  cluster_cols          = FALSE,
  scale                 = ”row”,
  gaps_col = c(4,8)
)
```



```
#Save the heatmap
pheatmap <- pheatmap(
  mat                    = all,
  annotation_row = annotation,
  annotation_colors = ann_colors,
  cellheight=5,
  cellwidth = 5,
  na_col = "grey",
  color = my.colors,
  breaks = my.breaks,
  show_colnames      = TRUE,
  show_rownames      = TRUE,
  drop_levels        = TRUE,
  fontsize           = 5.5,
  cluster_rows       = FALSE,
  cluster_cols       = FALSE,
  scale              = "row",
  gaps_col = c(4,8),
  filename = "~/Documents/Postdoc/PD_Projects/3_irCLIP-RNP/MS/RNase_HighLow_293T/3_DEP_
    visualization/all_heatmap_LFQ.pdf",
  width = 5,
  height = 5
)
```

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] grid      stats4    stats     graphics  grDevices utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] SuperExactTest_1.1.0        eulerr_7.0.1
##  [3] data.table_1.15.2           DESeq2_1.38.3
##  [5] pheatmap_1.0.12             arcdiagram_0.1.12
##  [7] UpSetR_1.4.0                colormap_0.1.4
##  [9] ggraph_2.2.1                igraph_2.0.3
## [11] hrbrthemes_0.8.7            patchwork_1.2.0
## [13] viridis_0.6.5               viridisLite_0.4.2
## [15] Clipper_0.0.0.9000          ggpubr_0.6.0
## [17] lubridate_1.9.3             forcats_1.0.0
## [19] stringr_1.5.1               dplyr_1.1.4
## [21] purrr_1.0.2                 readr_2.1.5
## [23] tidyr_1.3.1                 tibble_3.2.1
## [25] ggplot2_3.5.0               tidyverse_2.0.0
## [27] DEP2_0.4.8.24               R6_2.5.1
## [29] limma_3.54.2                MSnbase_2.24.2
## [31] ProtGenerics_1.30.0         mzR_2.32.0
## [33] Rcpp_1.0.12                 MsCoreUtils_1.10.0
## [35] SummarizedExperiment_1.28.0 Biobase_2.58.0
## [37] GenomicRanges_1.50.2        GenomeInfoDb_1.34.9
## [39] IRanges_2.32.0              S4Vectors_0.36.2
## [41] BiocGenerics_0.44.0         MatrixGenerics_1.10.0
## [43] matrixStats_1.2.0           formatR_1.14
##
## loaded via a namespace (and not attached):
##   [1] missForest_1.5              bit64_4.0.5
##   [3] knitr_1.45                  DelayedArray_0.24.0
##   [5] KEGGREST_1.38.0             RCurl_1.98-1.14
##   [7] AnnotationFilter_1.22.0     doParallel_1.0.17
##   [9] generics_0.1.3              preprocessCore_1.60.2
##  [11] cowplot_1.1.3               RSQLite_2.3.5
##  [13] proxy_0.4-27                bit_4.0.5
##  [15] tzdb_0.4.0                  httpuv_1.6.14
##  [17] assertthat_0.2.1            TCseq_1.22.6
##  [19] xfun_0.42                   hms_1.1.3
##  [21] evaluate_0.23               promises_1.2.1
##  [23] fansi_1.0.6                 DBI_1.2.2
##  [25] geneplotter_1.76.0          ellipsis_0.3.2
##  [27] RSpectra_0.16-1             QFeatures_1.8.0
##  [29] backports_1.4.1             fontLiberation_0.1.0
##  [31] V8_4.4.2                    annotate_1.76.0
##  [33] fontBitstreamVera_0.1.1     vctrs_0.6.5
##  [35] imputeLCMD_2.1              abind_1.4-5
##  [37] cachem_1.0.8                withr_3.0.0
##  [39] ggforce_0.4.2               itertools_0.1-3
##  [41] GenomicAlignments_1.34.1    fdrtool_1.2.17
##  [43] MultiAssayExperiment_1.24.0 cluster_2.1.6
##  [45] lazyeval_0.2.2              crayon_1.5.2
##  [47] crul_1.4.0                  labeling_0.4.3
##  [49] glmnet_4.1-8                edgeR_3.40.2
##  [51] pkgconfig_2.0.3             tweenr_2.0.3
##  [53] rlang_1.1.3                 lifecycle_1.0.4
##  [55] sandwich_3.1-0              downloader_0.4
##  [57] fontquiver_0.2.1            httpcode_0.3.0
##  [59] affyio_1.68.0               extrafontdb_1.0
##  [61] randomForest_4.7-1.1        polyclip_1.10-6
##  [63] rngtools_1.5.2              Matrix_1.6-5
##  [65] carData_3.0-5               zoo_1.8-12
##  [67] GlobalOptions_0.1.2         png_0.1-8
##  [69] rjson_0.2.21                bitops_1.0-7
##  [71] Biostrings_2.66.0           blob_1.2.4
##  [73] doRNG_1.8.6                 shape_1.4.6.1
##  [75] rstatix_0.7.2               tmvtnorm_1.6
##  [77] ggsignif_0.6.4              scales_1.3.0
##  [79] memoise_2.0.1               magrittr_2.0.3
##  [81] plyr_1.8.9                  zlibbioc_1.44.0
##  [83] compiler_4.2.1              RColorBrewer_1.1-3
##  [85] pcaMethods_1.90.0           clue_0.3-65
##  [87] Rsamtools_2.14.0            cli_3.6.2
##  [89] affy_1.76.0                 XVector_0.38.0
##  [91] MASS_7.3-60.0.1             tidyselect_1.2.1
##  [93] vsn_3.66.0                  stringi_1.8.3
##  [95] highr_0.10                  yaml_2.3.8
##  [97] norm_1.0-11.1               askpass_1.2.0
##  [99] locfit_1.5-9.9              MALDIquant_1.22.2
## [101] ggrepel_0.9.5               tools_4.2.1
## [103] timechange_0.3.0            parallel_4.2.1
## [105] circlize_0.4.16             rstudioapi_0.15.0
## [107] foreach_1.5.2               gridExtra_2.3
## [109] farver_2.1.1                mzID_1.36.0
## [111] Rtsne_0.17                  digest_0.6.35
## [113] BiocManager_1.30.22         shiny_1.8.0
## [115] gfonts_0.2.0                car_3.1-2
## [117] broom_1.0.5                 later_1.3.2
## [119] ncdf4_1.22                  httr_1.4.7
## [121] gdtools_0.3.5               AnnotationDbi_1.60.2
## [123] ComplexHeatmap_2.14.0       colorspace_2.1-0
## [125] XML_3.99-0.16.1             reticulate_1.35.0
## [127] umap_0.2.10.0               splines_4.2.1
## [129] graphlayouts_1.1.1          gmm_1.8
## [131] systemfonts_1.0.5           xtable_1.8-4
## [133] jsonlite_1.8.8              tidygraph_1.3.1
## [135] pillar_1.9.0                htmltools_0.5.7
## [137] mime_0.12                   glue_1.7.0
## [139] fastmap_1.1.1               BiocParallel_1.32.6
## [141] class_7.3-22                codetools_0.2-19
## [143] mvtnorm_1.2-4               utf8_1.2.4
## [145] lattice_0.22-5              curl_5.2.1
## [147] openssl_2.1.1               Rttf2pt1_1.3.12
## [149] survival_3.5-8              rmarkdown_2.26
## [151] munsell_0.5.0               e1071_1.7-14
## [153] GetoptLong_1.0.5            GenomeInfoDbData_1.2.9
## [155] iterators_1.0.14            impute_1.72.3
## [157] reshape2_1.4.4              gtable_0.3.4
## [159] extrafont_0.19
```