qPLEXdata

(Analysis of qPLEX-RIME data and Total proteome data)

Kamal Kishore

November 17, 2026

Contents

1 Description **1**

2 Usage **1**

3 Experiment 1: ER interactome in MCF7 cells **2**

4 Experiment 2: Crosslinking comparison **3**

5 Experiment 3: ER complex upon OHT treatment **3**

6 Experiment 4: Total proteome upon OHT treatment **7**

7 Experiment 5: ER interactome in PDX tumours **8**

8 Experiment 6: ER interactome in human breast cancer tumours **9**

9 Experiment 7: NCOA3 interactome in MCF7 cells **10**

10 Experiment 8: CBP interactome in MCF7 cells **10**

11 Experiment 9: POLR2A (RNA polymerase II) interactome in MCF7 cells **11**

12 Session Information **12**

1 Description

The report outlines the steps for the statistical analysis of the datasets generated from the application of the qPLEX-RIME approach in breast cancer cells and clinical tumour material (PDX and human breast cancer tumours). In addition, it also contains statistical analysis of total proteome data in breast cancer cells.

2 Usage

```
library(dplyr)
suppressWarnings(library(qPLEXanalyzer))
suppressWarnings(library(qPLEXdata))
data(human_anno)
```

3 Experiment 1: ER interactome in MCF7 cells

In this experiment we have used the qPLEX-RIME approach to identify ER-associated proteins. We performed replicate ER RIME pull-downs in five independent biological replicates and an equal number of matched IgG mock samples was included. As IgG samples represent the low background intensity, their intensity distribution profile is different from ER samples. Hence, normalizing the two together would have resulted in over-correction of the IgG intensity resulting in inaccurate computation of differences between the two groups. Therefore, the peptide intensities were normalized by median scaling within each group separately. The normalized peptide intensities were aggregated (by summing) to protein intensities. Thereafter, differential protein expression was performed using *limma* based analysis.

```
## load data
data(exp1_specificity)

## create MSnSet object
MSnset_data <- convertToMSnset(exp1_specificity$intensities,
                               metadata=exp1_specificity$metadata,
                               indExpData=c(6:15),Sequences=1,Accessions=5)

## Normalization
MSnset_norm <- groupScaling(MSnset_data, median)

## Warning:  There was 1 warning in `summarize()`.
## i In argument:  `across(everything(), scalingFunction, na.rm = TRUE)`.
## Caused by warning:
## !  The `...` argument of `across()` is deprecated as of dplyr 1.1.0.
## Supply arguments directly to `.fns` through an anonymous function
## instead.
##
##   # Previously
##   across(a:b, mean, na.rm = TRUE)
##
##   # Now
##   across(a:b, \(x) mean(x, na.rm = TRUE))
## i The deprecated feature was likely used in the qPLEXanalyzer package.
##   Please report the issue at
##   <https://github.com/crukci-bioinformatics/qPLEXanalyzer/issues>.

## Summation of peptide to protein intensity
MSnset_Pnorm <- summarizeIntensities(MSnset_norm, sum, human_anno)

## Differential analysis
contrasts <- c(ER_vs_IgG = "ER - IgG")
diffstats <- computeDiffStats(MSnSetObj=MSnset_Pnorm, contrasts=contrasts)
diffexp <- getContrastResults(diffstats=diffstats, contrast=names(contrasts))
diffexp <- diffexp[which(diffexp$adj.P.Val < 0.01 & diffexp$log2FC >1),]
```

4 Experiment 2: Crosslinking comparison

An ER qPLEX-RIME experiment was performed to compare two different ways of cell crosslinking. MCF7 cells were crosslinked with DSG/formaldehyde (double) or with formaldehyde alone (single). Four biological replicates were obtained for each condition along with two IgG samples pooled from replicates of each group. One of the outlier sample was removed from the analysis. The peptide intensities were normalized using median scaling within the group, treating ER pull downs as one group and IgG pull downs as another. The normalized peptide intensities were aggregated (by summing) to protein intensities. Thereafter, differential protein expression was performed using *limma* based analysis.

```
## load data
data(exp2_Xlink)

## create MSnSet object
MSnset_data <- convertToMSnset(exp2_Xlink$intensities,
                               metadata=exp2_Xlink$metadata,
                               indExpData=c(7:16),Sequences=2,Accessions=6)
exprs(MSnset_data) <- exprs(MSnset_data)+0.01
MSnset_data <- MSnset_data[,-5]

## Normalization
MSnset_norm <- groupScaling(MSnset_data, median)

## Summation of peptide to protein intensity
MSnset_Pnorm <- summarizeIntensities(MSnset_norm, sum, human_anno)

## Differential analysis
contrasts <- c(DSG.FA_vs_FA = "DSG.FA - FA")
diffstats <- computeDiffStats(MSnSetObj=MSnset_Pnorm, contrasts=contrasts)
diffexp <- getContrastResults(diffstats=diffstats, contrast=names(contrasts),
                              controlGroup = "IgG")
diffexp <- diffexp[which(diffexp$adj.P.Val < 0.05 & abs(diffexp$log2FC) > 0.5),]
```

5 Experiment 3: ER complex upon OHT treatment

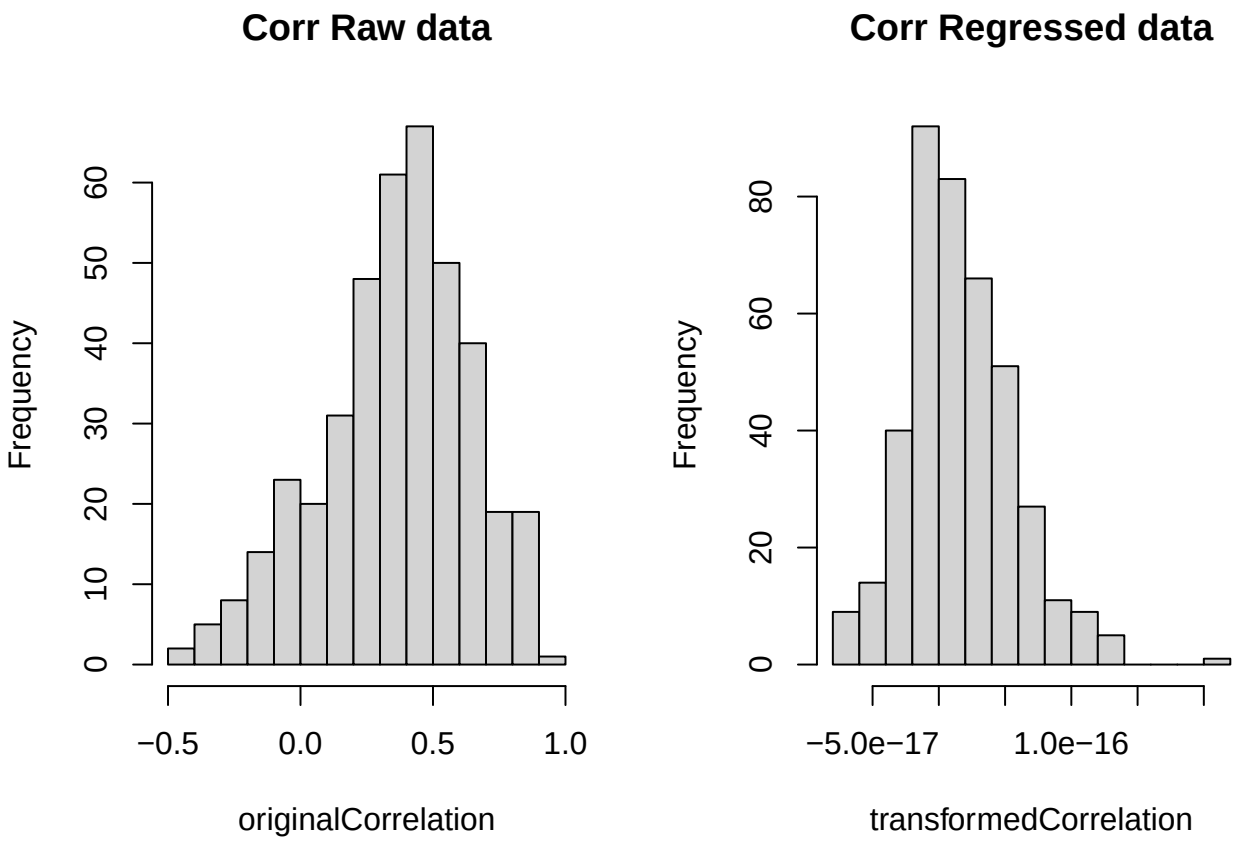
Three ER qPLEX-RIME experiments were performed to investigate the dynamics of the ER complex assembly upon 4-hydroxytamoxifen (OHT) treatment at 2h, 6h and 24h or at 24h post-treatment with the vehicle alone (ethanol). Two biological replicates of each condition were included in each experiment to finally consider a total of six replicates per time point. Additionally, MCF7 cells were treated with OHT or ethanol and cross-linked at 24h post-treatment in each experiment to be used for mock IgG pull-downs and to enable discrimination of non-specific binding.

The peptide intensities in each 10plex experiment were aggregated (by summing) to protein intensities. Thereafter, protein intensity in each sample was divided by its average intensity across all the samples and log2 transformed. The scaled protein intensities from each 10plex experiment were then combined and only proteins identified in all the experiments were kept for further analysis. To filter non-specific proteins, a *limma* based differential analysis was performed comparing ER and IgG pull-downs. This step filtered out non-specific binding. Further, a subset of dataset was created excluding IgG pull-down samples.

A linear regression method was applied on proteins (of this dataset) that were identified across all three replicate experiments to correct for the ER dependency. Finally, *limma* based statistical was applied on the regressed intensities for the identification of differentially bound proteins.

```
## load data
data(exp3_OHT_ESR1)

## create MSnSet object
MSnset_data1 <- convertToMSnset(exp3_OHT_ESR1$intensities_qPLEX1,
                                metadata=exp3_OHT_ESR1$metadata_qPLEX1,
                                indExpData=c(7:16),Sequences=2,Accessions=6)
pData(MSnset_data1)$Run <- 1
MSnset_data2 <- convertToMSnset(exp3_OHT_ESR1$intensities_qPLEX2,
                                metadata=exp3_OHT_ESR1$metadata_qPLEX2,
                                indExpData=c(7:16),Sequences=2,Accessions=6)
pData(MSnset_data2)$Run <- 2
MSnset_data3 <- convertToMSnset(exp3_OHT_ESR1$intensities_qPLEX3,
                                metadata=exp3_OHT_ESR1$metadata_qPLEX3,
                                indExpData=c(7:16),Sequences=2,Accessions=6)
pData(MSnset_data3)$Run <- 3

## Summation of peptide to protein intensity
MSnset_P1 <- summarizeIntensities(MSnset_data1, sum, human_anno)
MSnset_P2 <- summarizeIntensities(MSnset_data2, sum, human_anno)
MSnset_P3 <- summarizeIntensities(MSnset_data3, sum, human_anno)

## Normalization
MSnset_P1 <- rowScaling(MSnset_P1,mean)
MSnset_P2 <- rowScaling(MSnset_P2,mean)
MSnset_P3 <- rowScaling(MSnset_P3,mean)

###### Compute common unique peptides
features1 <- fData(MSnset_data1)
features1 <- as.data.frame(features1[, c("Sequences",
                                         "Accessions")],
                           stringsAsFactors = FALSE)
features2 <- fData(MSnset_data2)
features2 <- as.data.frame(features2[, c("Sequences",
                                           "Accessions")],
                           stringsAsFactors = FALSE)
features3 <- fData(MSnset_data3)
features3 <- as.data.frame(features3[, c("Sequences",
                                           "Accessions")],
                           stringsAsFactors = FALSE)
features <- rbind(features1,features2,features3)
features <- unique(features)
features$Sequences <- as.character(features$Sequences)
features$Accessions <- as.character(features$Accessions)
counts <- features %>% count(Accessions)
colnames(counts)[2] <- "Count"
```

```
##### create combine MSnSet object

MSnset_P1 <- updateFvarLabels(MSnset_P1)
MSnset_P2 <- updateFvarLabels(MSnset_P2)
MSnset_P3 <- updateFvarLabels(MSnset_P3)

MSnset_P1 <- updateSampleNames(MSnset_P1)
MSnset_P2 <- updateSampleNames(MSnset_P2)
MSnset_P3 <- updateSampleNames(MSnset_P3)

suppressWarnings(MSnset_comb <- combine(MSnset_P1, MSnset_P2, MSnset_P3))
tokeep <- which(complete.cases(fData(MSnset_comb))==TRUE)
MSnset_comb <- MSnset_comb[tokeep,]
sampleNames(MSnset_comb) <- pData(MSnset_comb)$SampleName

pData(MSnset_comb)$BioRep <- c(rep(1,4),rep(2,4),c(1,2),rep(3,4),rep(4,4),c(3,4),
                           rep(5,4),rep(6,4),c(5,6))
fData(MSnset_comb) <- fData(MSnset_comb)[,c(1:4)]
colnames(fData(MSnset_comb)) <- c("Accessions","Gene","Description",
                                  "GeneSymbol")
ind <- match(fData(MSnset_comb)$Accessions, counts$Accessions)
fData(MSnset_comb)$Count <- counts$Count[ind]

### create separate MSnSet for IgG comparision
pheno <- pData(MSnset_comb)
pheno$SampleGroup <- c(rep(c(rep("Exp",8),rep("IgG",2)),3))
pheno$SampleGroup <- factor(pheno$SampleGroup)
MSnset_IgG <- MSnset_comb
pData(MSnset_IgG) <- pheno

### Differential analysis to find ER specific interactors
contrasts <- c(
  Exp_vs_IgG = "Exp - IgG"
)

diffstats <- computeDiffStats(MSnSetObj=MSnset_IgG, contrasts=contrasts,
                              transform = FALSE)
results <- getContrastResults(diffstats=diffstats, contrast=names(contrasts),
                              transform = FALSE)

### create subset of protein filtering non-specific IgG
ind <- which(results$adj.P.Val < 0.01 & results$log2FC > 1)
diff_IgG <- results[ind,]
ind <- match(diff_IgG$Accessions, fData(MSnset_comb)$Accessions)
MSnset_subset <- MSnset_comb[ind]
IgG_ind <- which(pData(MSnset_subset)$SampleGroup == "IgG")

### perform regression analysis on dataset
MSnset_reg <- regressIntensity(MSnset_subset, controlInd=IgG_ind, ProteinId="P03372")
```




```
### Differential analysis
contrasts <- c(
  tam.2h_vs_vehicle = "tam.2h - vehicle",
  tam.6h_vs_vehicle = "tam.6h - vehicle",
  tam.24h_vs_vehicle = "tam.24h - vehicle"
)

suppressWarnings(diffstats <- computeDiffStats(MSnSetObj=MSnset_reg, contrasts=contrasts,
                              transform = FALSE))

diffexp1 <- getContrastResults(diffstats=diffstats, contrast=names(contrasts)[1],
                              transform = FALSE)
diffexp1 <- diffexp1[which(diffexp1$adj.P.Val < 0.05 & abs(diffexp1$log2FC) > 0.5),]
diffexp2 <- getContrastResults(diffstats=diffstats, contrast=names(contrasts)[2],
                              transform = FALSE)
diffexp2 <- diffexp2[which(diffexp2$adj.P.Val < 0.05 & abs(diffexp2$log2FC) > 0.5),]
diffexp3 <- getContrastResults(diffstats=diffstats, contrast=names(contrasts)[3],
                              transform = FALSE)
diffexp3 <- diffexp3[which(diffexp3$adj.P.Val < 0.05 & abs(diffexp3$log2FC) > 0.5),]
```

6 Experiment 4: Total proteome upon OHT treatment

We performed two 10plex-TMT time-course experiments to study the effect of OHT on total protein levels. MCF7 cells were treated with OHT for 2h, 6h, 24h or for 24h with the vehicle alone (ethanol) and a total number of four biological replicates was obtained. The peptide intensities in each 10plex experiment were aggregated (by summing) to protein intensities and only proteins that were identified in both experiments were used for further analysis. The protein intensities were then normalized using median scaling to account for sample loading variability. Thereafter, differential protein expression was performed using *limma* based analysis.

```
## load data
data(exp4_OHT_FP)

## create MSnSet object
MSnset_data1 <- convertToMSnset(exp4_OHT_FP$FP_1,
                                metadata=exp4_OHT_FP$metadata_FP1,
                                indExpData=c(7:14),Sequences=2,Accessions=6)
pData(MSnset_data1)$Run <- 1
MSnset_data2 <- convertToMSnset(exp4_OHT_FP$FP_2,
                                metadata=exp4_OHT_FP$metadata_FP2,
                                indExpData=c(7:14),Sequences=2,Accessions=6)
pData(MSnset_data2)$Run <- 2

## Summation of peptide to protein intensity
MSnset_P1 <- summarizeIntensities(MSnset_data1, sum, human_anno)
MSnset_P2 <- summarizeIntensities(MSnset_data2, sum, human_anno)


### Computing common unique peptides
features1 <- fData(MSnset_data1)
features1 <- as.data.frame(features1[, c("Sequences","Accessions")],
                           stringsAsFactors = FALSE)
features2 <- fData(MSnset_data2)
features2 <- as.data.frame(features2[, c("Sequences","Accessions")],
                           stringsAsFactors = FALSE)
features <- rbind(features1,features2)
features <- unique(features)
features$Sequences <- as.character(features$Sequences)
features$Accessions <- as.character(features$Accessions)
counts <- features %>% count(Accessions)
colnames(counts)[2] <- "Count"

##### create combine MSnSet object

MSnset_P1 <- updateFvarLabels(MSnset_P1)
MSnset_P2 <- updateFvarLabels(MSnset_P2)
MSnset_P1 <- updateSampleNames(MSnset_P1)
MSnset_P2 <- updateSampleNames(MSnset_P2)

suppressWarnings(MSnset_comb <- combine(MSnset_P1, MSnset_P2))
```

```
tokeep <- which(complete.cases(fData(MSnset_comb))==TRUE)
MSnset_comb <- MSnset_comb[tokeep,]
sampleNames(MSnset_comb) <- pData(MSnset_comb)$SampleName
fData(MSnset_comb) <- fData(MSnset_comb)[,c(1:4)]
colnames(fData(MSnset_comb)) <- c("Accessions","Gene","Description",
                                  "GeneSymbol")
ind <- match(fData(MSnset_comb)$Accessions,counts$Accessions)
fData(MSnset_comb)$Count <- counts$Count[ind]

## Normalization
MSnset_Pnorm <- normalizeScaling(MSnset_comb, median)

## Differential analysis
contrasts <- c(
  tam.2h_vs_vehicle = "tam.2h - vehicle",
  tam.6h_vs_vehicle = "tam.6h - vehicle",
  tam.24h_vs_vehicle = "tam.24h - vehicle"
)
batchEffect <- c("Run", "BioRep")

diffstats <- computeDiffStats(MSnset_Pnorm, contrasts=contrasts,
                              batchEffect=batchEffect)

diffexp1 <- getContrastResults(diffstats=diffstats, contrast=names(contrasts)[1])
diffexp1 <- diffexp1[which(diffexp1$adj.P.Val < 0.05 & abs(diffexp1$log2FC) > 0.5),]
diffexp2 <- getContrastResults(diffstats=diffstats, contrast=names(contrasts)[2])
diffexp2 <- diffexp2[which(diffexp2$adj.P.Val < 0.05 & abs(diffexp2$log2FC) > 0.5),]
diffexp3 <- getContrastResults(diffstats=diffstats, contrast=names(contrasts)[3])
diffexp3 <- diffexp3[which(diffexp3$adj.P.Val < 0.05 & abs(diffexp3$log2FC) > 0.5),]
```

7 Experiment 5: ER interactome in PDX tumours

An ER qPLEX-RIME experiment was performed using three independent ER+ human Patient Derived Xenograft (PDX) tumours. Cryosections of each tumour were double-crosslinked and each tumour was split in two parts that were used for ER and IgG RIME pull-down assays. One of the tumours was split in three different parts to be used as ER or IgG qPLEX-RIME in order to assess technical variability. The peptide intensities were normalized by median scaling within each group separately. The normalized peptide intensities were aggregated (by summing) to protein intensities. Thereafter, differential protein expression was performed using *limma* based analysis.

```
## load data
data(exp5_PDX)

## create MSnSet object
MSnset_data <- convertToMSnset(exp5_PDX$intensities, metadata=exp5_PDX$metadata,
                               indExpData=c(7:16), Sequences=2,Accessions=6)

## Exclude outlier and techical replicate samples
```

```
MSnset_data <- MSnset_data[,-c(7:10)]

## Normalization
MSnset_norm <- groupScaling(MSnset_data, median)

## Summation of peptide to protein intensity
MSnset_Pnorm <- summarizeIntensities(MSnset_norm, sum, human_anno)

## Differential analysis
contrasts <- c(PDX_vs_IgG = "PDX - IgG")

diffstats <- computeDiffStats(MSnset_Pnorm, contrasts=contrasts)

## Warning in duplicateCorrelation(intensities, design = design, block = pData(MSnSetObj)$TechRep
Blocks all of size 1:  setting intrablock correlation to zero.

diffexp <- getContrastResults(diffstats=diffstats, contrast=names(contrasts))
diffexp <- diffexp[which(diffexp$adj.P.Val < 0.05 & diffexp$log2FC > 1),]
```

8 Experiment 6: ER interactome in human breast cancer tumours

An ER qPLEX-RIME experiment was performed using five independent ER-positive human breast cancer tumours. Cryosections of each tumour were double-crosslinked and each tumour was split in two parts that were used for ER and IgG RIME pull-down assays. The peptide intensities were normalized by median scaling within each group separately. The normalized peptide intensities were aggregated (by summing) to protein intensities followed by differential protein expression using *limma* based analysis.

```
## load data
data(exp6_ER)

## create MSnSet object
MSnset_data <- convertToMSnset(exp6_ER$intensities, metadata=exp6_ER$metadata,
                               indExpData=c(6:15), Sequences=2, Accessions=5,
                               rmMissing=FALSE)
exprs(MSnset_data)[is.na(exprs(MSnset_data))] <- 0
exprs(MSnset_data) <- exprs(MSnset_data)+0.01

## Normalization
MSnset_norm <- groupScaling(MSnset_data, median, groupingColumn="SampleGroup")

## Summation of peptide to protein intensity
MSnset_Pnorm <- summarizeIntensities(MSnset_norm, sum, human_anno)

## Differential analysis
contrasts <- c(ER_vs_IgG = "ER - IgG")
diffstats <- computeDiffStats(MSnset_Pnorm, contrasts=contrasts)
diffexp <- getContrastResults(diffstats=diffstats, contrast=names(contrasts))
diffexp <- diffexp[which(diffexp$adj.P.Val < 0.01 & diffexp$log2FC >1),]
```

9 Experiment 7: NCOA3 interactome in MCF7 cells

In this experiment we have used the qPLEX-RIME method to identify and characterize NCOA3 (SRC-3) associated proteins. We performed NCOA3 RIME pull-downs in five independent biological replicates and in five matched IgG mock samples. The peptide intensities were normalized by median scaling within each group separately. The normalized peptide intensities were aggregated (by summing) to protein intensities followed by differential protein expression using *limma* based analysis.

```
## load data
data(exp7_NCOA3)

## create MSnSet object
MSnset_data <- convertToMSnset(exp7_NCOA3$intensities, metadata=exp7_NCOA3$metadata,
                               indExpData=c(7:16), Sequences=2, Accessions=6,
                               rmMissing=FALSE)
exprs(MSnset_data)[is.na(exprs(MSnset_data))] <- 0
exprs(MSnset_data) <- exprs(MSnset_data)+0.01

## Normalization
MSnset_norm <- groupScaling(MSnset_data, median, groupingColumn="SampleGroup")

## Summation of peptide to protein intensity
MSnset_Pnorm <- summarizeIntensities(MSnset_norm, sum, human_anno)

## Differential analysis
contrasts <- c(NCOA3_vs_IgG = "NCOA3 - IgG")
diffstats <- computeDiffStats(MSnset_Pnorm, contrasts=contrasts)
diffexp <- getContrastResults(diffstats=diffstats, contrast=names(contrasts))
diffexp <- diffexp[which(diffexp$adj.P.Val < 0.01 & diffexp$log2FC >1),]
```

10 Experiment 8: CBP interactome in MCF7 cells

A qPLEX-RIME experiment was designed for the characterization of the CBP (CREB-binding protein) interactome. Five independent biological replicates of CBP RIME pull-downs and five IgG RIME pull-downs were prepared for this experiment. The peptide intensities were normalized by median scaling within each group separately. The normalized peptide intensities were aggregated (by summing) to protein intensities followed by differential protein expression using *limma* based analysis.

```
## load data
data(exp8_CBP)

## create MSnSet object
MSnset_data <- convertToMSnset(exp8_CBP$intensities, metadata=exp8_CBP$metadata,
                               indExpData=c(7:16), Sequences=2, Accessions=6,
                               rmMissing=FALSE)
exprs(MSnset_data)[is.na(exprs(MSnset_data))] <- 0
exprs(MSnset_data) <- exprs(MSnset_data)+0.01

## Normalization
```

```
MSnset_norm <- groupScaling(MSnset_data, median, groupingColumn="SampleGroup")

## Summation of peptide to protein intensity
MSnset_Pnorm <- summarizeIntensities(MSnset_norm, sum, human_anno)

## Differential analysis
contrasts <- c(CREBBP_vs_IgG = "CREBBP - IgG")
diffstats <- computeDiffStats(MSnset_Pnorm, contrasts=contrasts)
diffexp <- getContrastResults(diffstats=diffstats, contrast=names(contrasts))
diffexp <- diffexp[which(diffexp$adj.P.Val < 0.01 & diffexp$log2FC >1),]
```

11 Experiment 9: POLR2A (RNA polymerase II) interactome in MCF7 cells

The qPLEX-RIME method was applied for the characterization of the largest and catalytic component of RNA polymerase II (RPB1). Particularly, the phosphorylated form at Serine 5 in the C-terminal domain (CTD) was used as the bait protein. Five biological replicates of RNA polymerase II RIME pull-downs and five IgG pull-downs were included for the identification and characterization of RNA polymerase II-associated proteins. The peptide intensities were normalized by median scaling within each group separately. The normalized peptide intensities were aggregated (by summing) to protein intensities followed by differential protein expression using *limma* based analysis.

```
## load data
data(exp9_PolII)

## create MSnSet object
MSnset_data <- convertToMSnset(exp9_PolII$intensities, metadata=exp9_PolII$metadata,
                               indExpData=c(7:16), Sequences=2, Accessions=6,
                               rmMissing=FALSE)
exprs(MSnset_data)[is.na(exprs(MSnset_data))] <- 0
exprs(MSnset_data) <- exprs(MSnset_data)+0.01

## Normalization
MSnset_norm <- groupScaling(MSnset_data, median, groupingColumn="SampleGroup")

## Summation of peptide to protein intensity
MSnset_Pnorm <- summarizeIntensities(MSnset_norm, sum, human_anno)

## Differential analysis
contrasts <- c(POLR2A_vs_IgG = "POLR2A - IgG")
diffstats <- computeDiffStats(MSnset_Pnorm, contrasts=contrasts)
diffexp <- getContrastResults(diffstats=diffstats, contrast=names(contrasts))
diffexp <- diffexp[which(diffexp$adj.P.Val < 0.01 & diffexp$log2FC >1),]
```

12 Session Information

```
sessionInfo()

## R version 4.6.2 (2026-10-29)
## Platform: x86_64-pc-linux-gnu
## Running under: Ubuntu 26.04.1 LTS
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
## LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblasp-r0.3.32.so;  LAPACK version 3.
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## time zone: Etc/UTC
## tzcode source: system (glibc)
##
## attached base packages:
## [1] stats4    stats     graphics  grDevices utils     datasets
## [7] methods   base
##
## other attached packages:
##  [1] qPLEXdata_1.30.1     qPLEXanalyzer_1.30.0 MSnbase_2.38.1
##  [4] ProtGenerics_1.44.0  S4Vectors_0.50.3     mzR_2.46.0
##  [7] Rcpp_1.1.2           Biobase_2.72.0       BiocGenerics_0.58.1
## [10] generics_0.1.4       dplyr_1.2.1
##
## loaded via a namespace (and not attached):
##  [1] rlang_1.3.0                 magrittr_2.0.5
##  [3] clue_0.3-68                 otel_0.3.0
##  [5] matrixStats_1.5.0           compiler_4.6.2
##  [7] PTMods_1.0.0                vctrs_0.7.3
##  [9] reshape2_1.4.5              stringr_1.6.0
## [11] pkgconfig_2.0.3             MetaboCoreUtils_1.20.1
## [13] crayon_1.5.3                XVector_0.52.0
## [15] tzdb_0.5.0                  preprocessCore_1.74.0
## [17] purrr_1.2.2                 xfun_0.62
## [19] MultiAssayExperiment_1.38.0 highr_0.12
## [21] DelayedArray_0.38.2         BiocParallel_1.46.0
## [23] parallel_4.6.2              cluster_2.1.8.3
## [25] R6_2.6.1                    stringi_1.8.9
## [27] RColorBrewer_1.1-3          limma_3.68.5
## [29] GenomicRanges_1.64.0        Seqinfo_1.2.0
```

```
## [31] assertthat_0.2.1            SummarizedExperiment_1.42.0
## [33] iterators_1.0.14            knitr_1.52
## [35] readr_2.2.0                 IRanges_2.46.0
## [37] splines_4.6.2               Matrix_1.7-6
## [39] igraph_2.3.4                tidyselect_1.2.1
## [41] abind_1.4-8                 doParallel_1.0.17
## [43] codetools_0.2-20            affy_1.90.0
## [45] lattice_0.23-1              tibble_3.3.1
## [47] plyr_1.8.9                  withr_3.0.3
## [49] S7_0.2.2                    evaluate_1.0.5
## [51] Spectra_1.22.2              Biostrings_2.80.2
## [53] pillar_1.11.1               affyio_1.82.0
## [55] BiocManager_1.30.27         MatrixGenerics_1.24.0
## [57] foreach_1.5.2               MALDIquant_1.22.3
## [59] ncdf4_1.24                  hms_1.1.4
## [61] ggplot2_4.0.4               scales_1.4.0
## [63] glue_1.8.1                  lazyeval_0.2.3
## [65] maketools_1.3.2             tools_4.6.2
## [67] mzID_1.50.0                 sys_3.4.3
## [69] data.table_1.18.6.1         QFeatures_1.22.0
## [71] vsn_3.80.0                  buildtools_1.0.0
## [73] fs_2.1.0                    XML_3.99-0.25
## [75] grid_4.6.2                  impute_1.86.0
## [77] tidyr_1.3.2                 MsCoreUtils_1.24.0
## [79] PSMatch_1.16.0              cli_3.6.6
## [81] S4Arrays_1.12.1             ggdendro_0.2.0
## [83] AnnotationFilter_1.36.0     pcaMethods_2.4.0
## [85] gtable_0.3.6                digest_0.6.39
## [87] SparseArray_1.12.3          farver_2.1.2
## [89] lifecycle_1.0.5             statmod_1.5.2
## [91] MASS_7.3-66
```