PhosMap

Dongdong Zhan

2019-06-24

Introduction

PhosMap is a comprehensive R package for analyzing quantitative phosphoproteomics data, which provides mutltiple functions for users as follow: 1. clustering: principal component analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE); 2. differential expression analysis; 3. time course analysis; 4. kinase activity prediction to find activated/deactivated kinases from the kinase-substrate database 5. phosphorylation motif enrichment analysis to provide clues for finding candidate kinases that are not present in the database; 6. and data visualization.

Loading data

To test the efficacy of PhosMap and help users get started quickly, we collected a dataset including 39 phosphoproteomics and 32 proteomics raw files deposited in the ProteomeXchange Consortium (Ressa, et al., 2018).The partial key intermediate results were provided for uers to master PhosMap. We have embeded intermediate results from demo into PhosMap for help users get started.

```r
# load PhosMap
library("PhosMap")
# load intermediate results from https://github.com/ecnuzdd/PhosMap_datasets
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/BRAFi.RData"
load_data <- load_data_with_ftp(ftp_url, 'Rdata')
```

```
## First loading data from FTP sever, it may take a few minutes.
```

```
## Downloading data from ftp://111.198.139.72:4000/pub/PhosMap_datasets/BRAFi.RData.
```

```
## Completing the RData load.
```

```r
temp_file <- tempfile()
writeBin(load_data, temp_file)
load(temp_file)
```

Object description

- background_df A data frame for motif enrichment analysis as background
- combined_df_with_mapped_gene_symbol Get a data frame mapped GI number to Gene Symbol
- data_frame_normalization_with_control_no_pair A data frame containning phosphoproteomics data normalized by proteomics data.
- foreground_df A data frame for motif enrichment analysis as foreground.
- fuzzy_input_df A data frame for time course analysis as input.
- group A factor for experiment group information.
- merge_df_with_phospho_peptides A merged phosphoproteomics data frame based on peptides files

(unique ID).
- motif_group_m_ratio_df_mat A matrix for motif profile.
- phospho_data_filtering_STY_and_normalization A phosphoproteomics data frame after normalization and filtering.
- profiling_data_normalized A proteomics data frame after normalization and filtering.
- summary_df_of_unique_proteins_with_sites A data frame that phosphorylation sites had been mapping to protein sequence and eliminated redundancy.

Data pre-processing

An intact data pre-processing procedure of phosphoproteomics data covered three main steps: merging input files after quality control, mapping phosphorylation sites (p-sites) to the corresponding protein sequence and data normalization.

Merging input files after quality control

‘Phosphoproteomics data’ and ‘The phosphoproteomics experimental design’ are required as input. For p-sites detected by Mascot, PhosMap could provide confidence probability of p-sites extracted from Mascot xml file. For p-sites detected by other software, a two column table including their corresponding sequences and confidence probability was indispensable. Then quality control at peptide and site levels for each experiment was performed.

```r
BASE_DIR <- getwd() # working directory
BASE_DIR <- normalizePath(BASE_DIR)
phosphorylation_exp_design_info_file_path <- normalizePath(file.path(BASE_DIR,
                                                                     'phosphorylation_exp_design_info.txt'))
phosphorylation_peptide_dir <- normalizePath(file.path(BASE_DIR, 'phosphorylation_peptide_txt'))

if(FALSE){
  # if you have xml files from mascot results, you can run the cmd to parser them to text files.
  mascot_xml_dir <- normalizePath(file.path(BASE_DIR, 'mascot_xml'))
  mascot_txt_dir <- normalizePath(file.path(BASE_DIR, 'mascot_txt'))
  extract_psites_score(phosphorylation_exp_design_info_file_path, mascot_xml_dir, mascot_txt_dir)

  # Based on above-mentioned text files from Mascot results,
  # the following cmd can generate CSV files of phosphorylation sites with confidence score.
  psites_score_dir <- normalizePath(file.path(BASE_DIR, 'psites_score_txt'))
  generate_psites_score_file(mascot_txt_dir, phosphorylation_peptide_dir, psites_score_dir)
}
# Merge phosphoproteomics data based on peptides files (unique ID).
# If qc = TRUE, considering confidence score of phosphorylation sites.
# A merged phosphoproteomics data frame based on peptides files (unique ID).
merge_df_with_phospho_peptides <- pre_process_filter_psites(
  phosphorylation_peptide_dir,
  psites_score_dir,
  phosphorylation_exp_design_info_file_path,
  qc = TRUE,
  min_score = 20,
  min_FDR = 0.01
)
```

Mapping phosphorylation sites (p-sites) to the corresponding protein sequence

- Mapping protein gi to gene symbol and outputing expression profile matrix with gene symnol.

```
# Get a data frame mapped GI number to Gene Symbol.
# system.time({
#   combinated_df_with_mapped_gene_symbol = get_combined_data_frame(
#     merge_df_with_phospho_peptides, species = 'human', id_type = 'RefSeq_Protein_GI'
#   )
# })
head(combined_df_with_mapped_gene_symbol)
```

```
##                Sequence          ID     Modification GeneSymbol Exp027012
## 1 AAAAAATAPPsPGPAQPGPR gi|39930517 Phospho (ST)(11)       SAMD1 191101700
## 2        AAALSLSTLAsPK  gi|4758248 Phospho (ST)(11)       EFNB1  19612020
## 3        AAALsLSTLASPK  gi|4758248  Phospho (ST)(5)       EFNB1 225996500
## 4        AAALSLsTLASPK  gi|4758248  Phospho (ST)(7)       EFNB1  25986220
## 5        AAAtPESQEPQAK gi|13491174  Phospho (ST)(4)     MARCKSL1 543292900
## 6 AADPPAENSsAPEAEQGGAE gi|34098946 Phospho (ST)(10)        YBX1 385447600
##    Exp027013  Exp027014  Exp027015  Exp027016  Exp027017  Exp027018
## 1   72397330          0   32192620  231611600  124170700   28650550
## 2   17807950   19357340   79808370   84246970   56323030   48652040
## 3  148918100   77889940  360054600  226245100  189362500  203163300
## 4   12633960          0          0          0          0          0
## 5 1942228000 1524829000 1160647000 2877282000 1013506000 1597019000
## 6  624299300  579280200  364341700  649720200  415048000  890674400
##   Exp027019  Exp027020  Exp027021  Exp027022 Exp027023 Exp027024
## 1  31780240  123238200   30951010          0 159176400         0
## 2  33331610          0   26782070   59331840  81300420  11794550
## 3 159109000  112891900  117936700  130344300 136386600  40931780
## 4         0          0          0          0         0         0
## 5 578654800 2225267000 1101911000 2347455000 800039800  32572150
## 6 803288700 1327916000  637003800 1049919000 917176600  96717140
##    Exp027025  Exp027026  Exp027027  Exp027028  Exp027029  Exp027030
## 1          0          0   36968750  345538800  167434500  232864900
## 2   25698570   48053290   11260400   49733950   17026620   33251190
## 3   40848750   84373550   18245820  449307600   27093610  205768900
## 4          0          0          0          0          0    7733698
## 5 1403377000 2010127000 1768473000 2510754000 2389508000 2335473000
## 6  408221700  880458700  551895600 1435267000  626601400  653992300
##    Exp027031  Exp027032  Exp027033 Exp027034 Exp027035  Exp027036
## 1          0  341993300  171803500         0  38057890  109961100
## 2   68674110   82809850   20488140  66844210  94660320   26547410
## 3  140357500  203682300   71224800 170784100 153310800   41458280
## 4          0    5456458          0         0         0          0
## 5 2901491000 2740149000 1173034000 315909400 341922700 1432846000
## 6  693000900  766916700  568389600 639079000 754503200 1056575000
##    Exp027037  Exp027038  Exp027039  Exp027040  Exp027041  Exp027042
## 1  167525000  165389700   51471910  143915200  356015900  208103900
## 2          0          0   44516010  107605300   33192480          0
## 3   70873710   99639770  132810800   78469830   92153170  344426900
## 4          0          0    9818745          0          0          0
## 5 2104620000 2328424000 1673055000 2601683000 2413220000 1619502000
## 6  807748200  978603400  685710400  844008900  898072800  449702300
##   Exp027043  Exp027044 Exp027045 Exp027046 Exp027047 Exp027048  Exp027049
## 1 101188100          0  54273770  63307160  83332280 162916400  120589000
## 2         0          0         0         0         0  22553820          0
## 3 134677700   68013420  27048770  52237850  65281570 116332000          0
## 4         0          0         0         0         0         0          0
## 5 101506600 1102288000 865370300 172955100 192015000 687942500 1670224000
## 6 267850800  806592800 494795000         0 315606800 295383200          0
##    Exp027050
## 1  271443600
## 2          0
## 3          0
## 4          0
## 5 1428628000
## 6          0
```

- Constructing the data frame with unique phosphorylation site for each protein sequence.

```
# Assign psites to protein sequence.
# Unique ID: protein_gi + phosphorylation site in protein sequence.
# system.time({
#   summary_df_of_unique_proteins_with_sites = get_summary_with_unique_sites(
#     combinated_df_with_mapped_gene_symbol, species = 'human', fasta_type = 'refseq'
#   )
# })
head(summary_df_of_unique_proteins_with_sites)
```

```
##                   AA_in_protein AA_in_peptide            Sequence
## gi|39930517_s161          s161           s11 AAAAAATAPPsPGPAQPGPR
## gi|4758248_s281           s281            s5        AAALsLSTLASPK
## gi|4758248_s283           s283            s7        AAALSLsTLASPK
## gi|4758248_s287           s287           s11        AAALSLSTLAsPK
## gi|4758248_s292           s292            s3   GGsGTAGTEPSDIIIPLR
## gi|4758248_t284           t284            t8        AAALSLStLASPK
##                          ID     Modification GeneSymbol Exp027012
## gi|39930517_s161 gi|39930517 Phospho (ST)(11)      SAMD1 191101700
## gi|4758248_s281   gi|4758248  Phospho (ST)(5)      EFNB1 225996500
## gi|4758248_s283   gi|4758248  Phospho (ST)(7)      EFNB1  25986220
## gi|4758248_s287   gi|4758248 Phospho (ST)(11)      EFNB1  19612020
## gi|4758248_s292   gi|4758248  Phospho (ST)(3)      EFNB1         0
## gi|4758248_t284   gi|4758248  Phospho (ST)(8)      EFNB1         0
##                  Exp027013 Exp027014 Exp027015 Exp027016 Exp027017
## gi|39930517_s161  72397330         0  32192620 231611600 124170700
## gi|4758248_s281  148918100  77889940 360054600 226245100 189362500
## gi|4758248_s283   12633960         0         0         0         0
## gi|4758248_s287   17807950  19357340  79808370  84246970  56323030
## gi|4758248_s292          0         0         0         0         0
## gi|4758248_t284          0  12164570         0  34897000         0
##                  Exp027018 Exp027019 Exp027020 Exp027021 Exp027022
## gi|39930517_s161  28650550  31780240 123238200  30951010         0
## gi|4758248_s281  203163300 159109000 112891900 117936700 130344300
## gi|4758248_s283          0         0         0         0         0
## gi|4758248_s287   48652040  33331610         0  26782070  59331840
## gi|4758248_s292          0         0         0         0         0
## gi|4758248_t284          0         0         0         0         0
##                  Exp027023 Exp027024 Exp027025 Exp027026 Exp027027
## gi|39930517_s161 159176400         0         0         0  36968750
## gi|4758248_s281  136386600  40931780  40848750  84373550  18245820
## gi|4758248_s283          0         0         0         0         0
## gi|4758248_s287   81300420  11794550  25698570  48053290  11260400
## gi|4758248_s292          0         0         0         0         0
## gi|4758248_t284    4121336         0         0  53214350         0
##                  Exp027028 Exp027029 Exp027030 Exp027031 Exp027032
## gi|39930517_s161 345538800 167434500 232864900         0 341993300
## gi|4758248_s281  449307600  27093610 205768900 140357500 203682300
## gi|4758248_s283          0         0   7733698         0   5456458
## gi|4758248_s287   49733950  17026620  33251190  68674110  82809850
## gi|4758248_s292          0         0         0         0         0
## gi|4758248_t284   12024320         0         0         0         0
##                  Exp027033 Exp027034 Exp027035 Exp027036 Exp027037
## gi|39930517_s161 171803500         0  38057890 109961100 167525000
## gi|4758248_s281   71224800 170784100 153310800  41458280  70873710
## gi|4758248_s283          0         0         0         0         0
## gi|4758248_s287   20488140  66844210  94660320  26547410         0
## gi|4758248_s292          0         0         0         0         0
## gi|4758248_t284          0         0         0   6888869         0
##                  Exp027038 Exp027039 Exp027040 Exp027041 Exp027042
## gi|39930517_s161 165389700  51471910 143915200 356015900 208103900
## gi|4758248_s281   99639770 132810800  78469830  92153170 344426900
## gi|4758248_s283          0   9818745         0         0         0
## gi|4758248_s287          0  44516010 107605300  33192480         0
## gi|4758248_s292          0         0   7382834         0         0
## gi|4758248_t284          0  40079320         0  35790700  21361960
##                  Exp027043 Exp027044 Exp027045 Exp027046 Exp027047
## gi|39930517_s161 101188100         0  54273770  63307160  83332280
## gi|4758248_s281  134677700  68013420  27048770  52237850  65281570
## gi|4758248_s283          0         0         0         0         0
## gi|4758248_s287          0         0         0         0         0
## gi|4758248_s292          0         0         0         0         0
## gi|4758248_t284    8331117         0         0         0         0
##                  Exp027048 Exp027049 Exp027050
## gi|39930517_s161 162916400 120589000 271443600
## gi|4758248_s281  116332000         0         0
## gi|4758248_s283          0         0         0
## gi|4758248_s287   22553820         0         0
## gi|4758248_s292          0         0         0
## gi|4758248_t284          0         0         0
```

Data normalization

PhosMap provides two kinds of normalizations.

1. PhosMap allowed for a total sum scaling normalization.

```
# Imputation with the next order of magnitude of the minimum except for zero.
# Filtering data only including phosphorylation site.
phospho_data_filtering_STY_and_normalization_list <- get_normalized_data_of_psites(
  summary_df_of_unique_proteins_with_sites,
  phosphorylation_exp_design_info_file_path,
  topN = NA, mod_types = c('S', 'T', 'Y')
)
phospho_data_filtering_STY <-
  phospho_data_filtering_STY_and_normalization_list$ptypes_area_df_with_id
phospho_data_filtering_STY_and_normalization <-
  phospho_data_filtering_STY_and_normalization_list$ptypes_fot5_df_with_id
head(phospho_data_filtering_STY_and_normalization)
```

2. If having matched proteomics data with phosphoproteomics, PhosMap allowed for normalizing phosphoproteomics data based on proteomics data.

```
# Based on phospho_data_filtering_STY_and_normalization
ID <- paste(phospho_data_filtering_STY_and_normalization$GeneSymbol,
          phospho_data_filtering_STY_and_normalization$AA_in_protein,
          sep = '_')
Value <- phospho_data_filtering_STY_and_normalization[,-seq(1,6)]
phospho_data <- data.frame(ID, Value)
phospho_data_rownames <- paste(phospho_data_filtering_STY_and_normalization$ID,
                               phospho_data_filtering_STY_and_normalization$GeneSymbol,
                               phospho_data_filtering_STY_and_normalization$AA_in_protein,
                               sep = '_')
rownames(phospho_data) <- phospho_data_rownames
# Further normalize phosphoproteomics data based on proteomics data
# The configurations of function see help document.
data_frame_normalization_with_control_no_pair <- normalize_phos_data_to_profiling(
  phospho_data, profiling_data_normalized,
  phosphorylation_exp_design_info_file_path,
  profiling_exp_design_info_file_path,
  control_label = '0',
  pair_flag = FALSE
)
head(data_frame_normalization_with_control_no_pair)
```

Data analysis

PhosMap incorporated four analysis modules, including clustering and differential expression analysis, time course analysis, kinase-substrate enrichment analysis to find activated/deactivated kinases and motif enrichment analysis.

Clustering and differential expression analysis

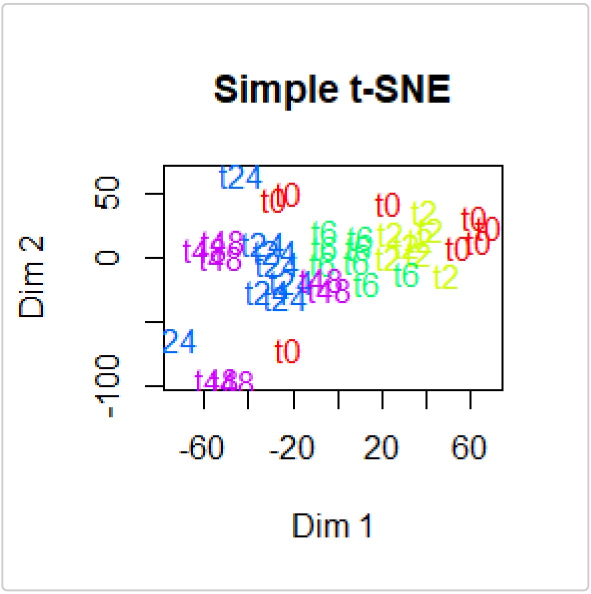
- In PhosMap, Clustering methods allowed for t-SNE and PCA

t-SNE

```
# Clustering: t-SNE or PCA
expr_data_frame <- data_frame_normalization_with_control_no_pair
# t-SNE using all experiments
visualization_with_simple_tsne(expr_data_frame, group)
```

```
## Loading required namespace: Rtsne
```

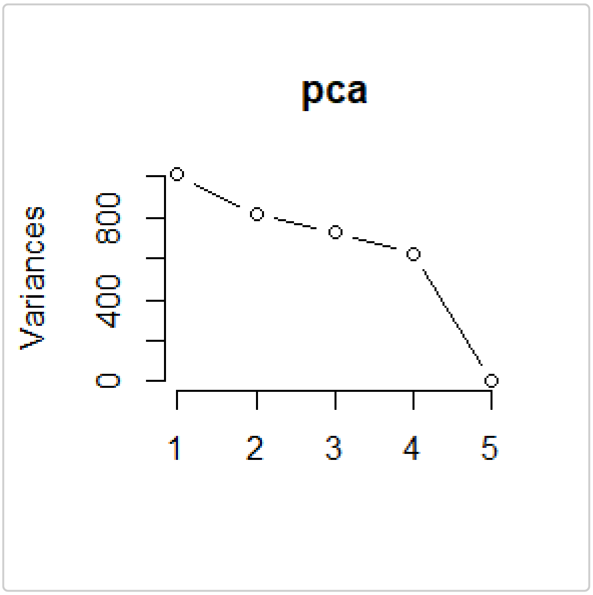
```
## Performing PCA
## Read the 39 x 39 data matrix successfully!
## OpenMP is working. 1 threads.
## Using no_dims = 2, perplexity = 10.000000, and theta = 0.500000
## Computing input similarities...
## Building tree...
## Done in 0.00 seconds (sparsity = 0.902038)!
## Learning embedding...
## Iteration 50: error is 55.494115 (50 iterations in 0.01 seconds)
## Iteration 100: error is 56.304273 (50 iterations in 0.01 seconds)
## Iteration 150: error is 60.784762 (50 iterations in 0.01 seconds)
## Iteration 200: error is 56.973069 (50 iterations in 0.00 seconds)
## Iteration 250: error is 60.232398 (50 iterations in 0.01 seconds)
## Iteration 300: error is 1.697451 (50 iterations in 0.01 seconds)
## Iteration 350: error is 1.123384 (50 iterations in 0.01 seconds)
## Iteration 400: error is 0.957553 (50 iterations in 0.00 seconds)
## Iteration 450: error is 0.807439 (50 iterations in 0.01 seconds)
## Iteration 500: error is 0.637739 (50 iterations in 0.01 seconds)
## Fitting performed in 0.08 seconds.
```

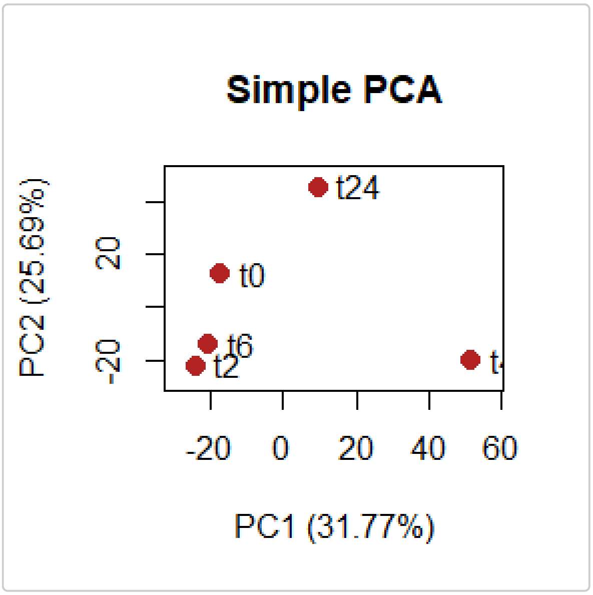


clustering example: t-SNE

PCA

```r
expr_ID <- as.vector(expr_data_frame[,1])
expr_Valule <- expr_data_frame[,-1]
expr_Valule_mean <- NULL
expr_Valule_row <- nrow(expr_Valule)
for(i in 1:expr_Valule_row){
  x <- as.vector(unlist(expr_Valule[i,]))
  x_m <- tapply(x, group, mean)
  expr_Valule_mean <- rbind(expr_Valule_mean, x_m)
}
group_levels = levels(group)
colnames(expr_Valule_mean) <- group_levels
expr_df <- data.frame(expr_ID, expr_Valule_mean)
# PCA using mean value in group for comparison with original literature
visualization_with_simple_pca(expr_df)
```





- In PhosMap, differential expression analysis methods allowed for limma, SAM and ANOVA Data preparation (t2 VS t0)

```
# Differently expressed Proteins/Genes analysis
# t2 vs t0
expr_data_frame <- data_frame_normalization_with_control_no_pair[,1:17]
# phosphoproteomics data normalized by proteomics data
# select phosphorylation sites with greater variation
expr_data_frame_var <- apply(expr_data_frame, 1, function(x){
  var(x[-1])
})
index_of_kept <- which(expr_data_frame_var>1)
expr_data_frame <- expr_data_frame[index_of_kept,]

# group information (t0 vs t2)
deps_group_levels <- c('t0', 't2')
deps_group <- factor(as.vector(group)[1:16], levels = deps_group_levels)
```

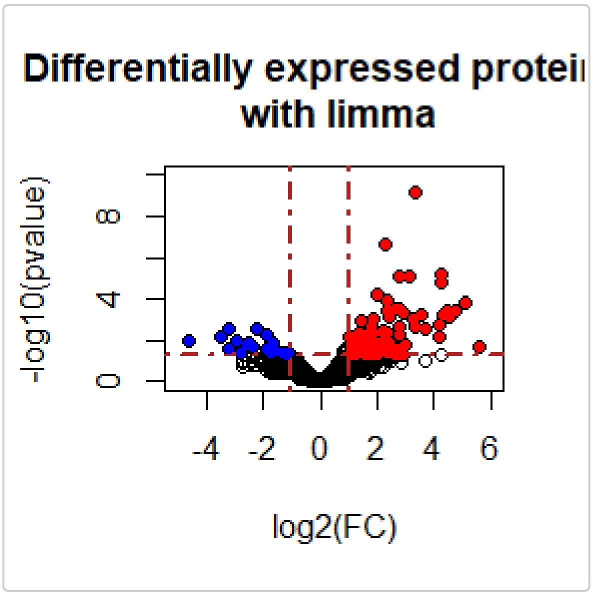
limma

```
# (1) limma
limma_results_df <- analysis_deps_limma(expr_data_frame, deps_group, deps_group_levels,
                                        log2_label = FALSE, adjust_method = 'none')
```

```
## Loading required namespace: limma
```

```
##
##  The matrix of experiment design.   groupt0 groupt2
## 1         1       0
## 2         1       0
## 3         1       0
## 4         1       0
## 5         1       0
## 6         1       0
## 7         1       0
## 8         1       0
## 9         0       1
## 10       0       1
## 11       0       1
## 12       0       1
## 13       0       1
## 14       0       1
## 15       0       1
## 16       0       1
## attr(,"assign")
## [1] 1 1
## attr(,"contrasts")
## attr(,"contrasts")$group
## [1] "contr.treatment"
##
##
##  The combination of pairwise comparison(s).
##  t2-t0
##
##  The matrix of comparison statement, compare other groups with control.      Contrasts
## Levels t2-t0
##     t0    -1
##     t2     1
```

```
limma_results_df$ID <- apply(limma_results_df, 1, function(x){
  x = strsplit(x, '_')[[1]]
  paste(x[2], x[3], sep = '_')
})
visualization_deps_with_scatter(limma_results_df, minFC = 2, minPvalue = 0.05,
                                main = 'Differentially expressed proteins  \n with limma',
                                show_text = FALSE, min_up_text = 70, min_down_text = 70)
```



Differential expression analysis: limma

SAM

```
# (2) SAM
sam_results_list <- analysis_deps_sam(expr_data_frame, deps_group, log2_label = FALSE, nperms = 100,
                                      rand = NULL, minFDR = 0.05, samr_plot = TRUE)
sam_results <- rbind(sam_results_list$genes_up_df, sam_results_list$genes_down_df)
```

ANOVA

```r
# (3) annova
anova_result_df <- analysis_deps_anova(expr_data_frame, deps_group, log2_label = FALSE,
                                       return_padjust = TRUE, adjust_method = 'BH')
visualization_deps_with_scatter(anova_result_df, minFC = 2, minPvalue = 0.05,
                                main = 'Differentially expressed proteins \n with anova',
                                show_text = FALSE, min_up_text = 15, min_down_text = 15)
```

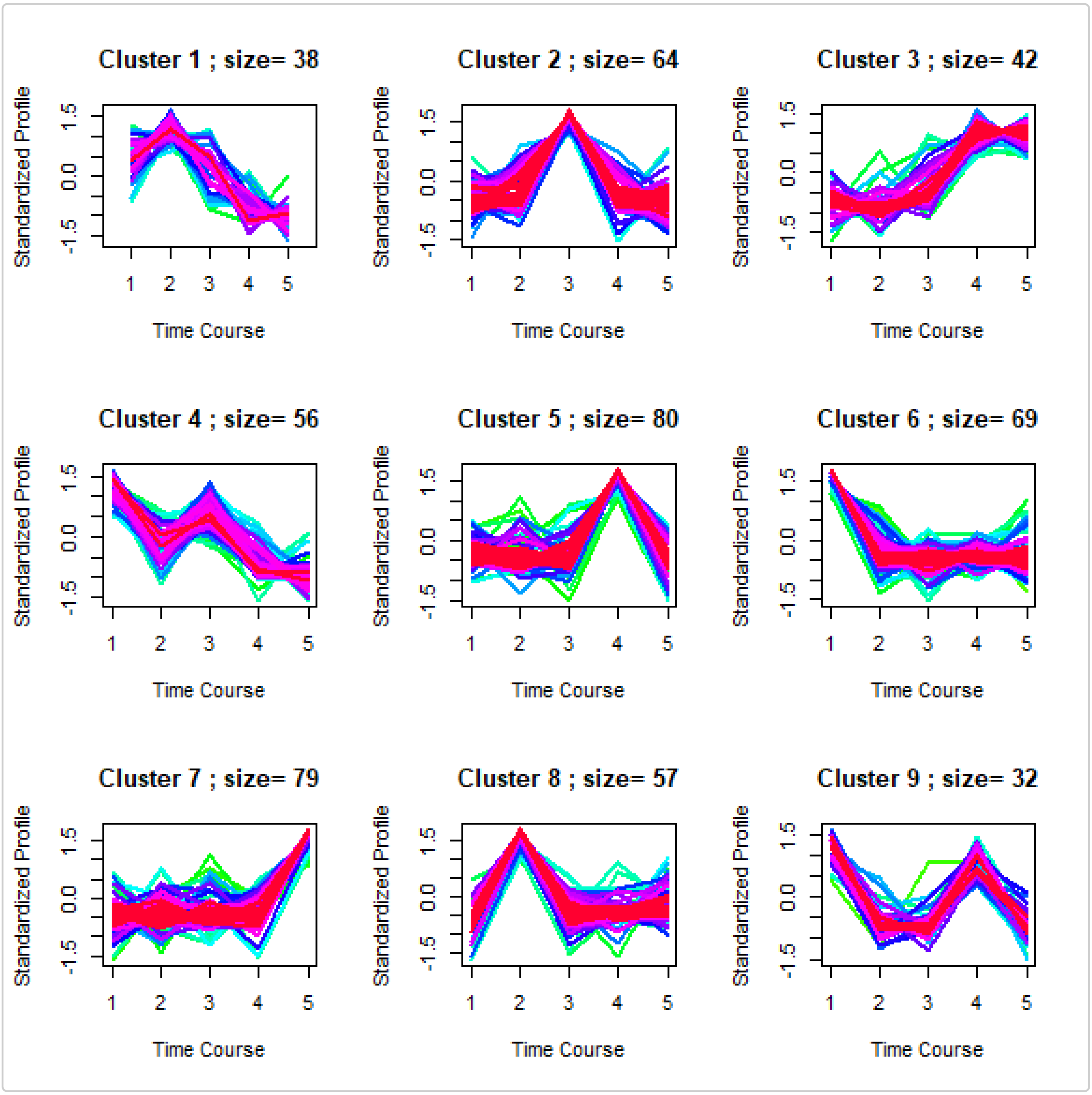
Time course analysis

Fuzzy clustering was applied to time course analysis for discovering patterns associated with time points in PhosMap.The corresponding line chart combined with membership for each cluster was also drawn.

```r
group_levels <- levels(group)
# fuzzy c-means clustering
set.seed(1000)
fuzzy_clustObj <- visualization_fuzzycluster(
    fuzzy_input_df, group, group_levels,
    k_cluster=9, iteration = 100,
    mfrow = c(3,3), min_mem = 0.1,
    plot = TRUE
)
```

```
## Loading required namespace: e1071
```

```
## Loading required namespace: ClueR
```



Time course analysis example

```
# clusters information
clusterS_info <- fuzzy_clustObj$cluster
clusterS_names <- names(clusterS_info)
clusters_df <- data.frame(clusterS_names, clusterS_info)
# write.csv(clusters_df, 'clusters_df.csv', row.names = TRUE)
```

Kinase activity prediction to find activated/deactivated kinases

In PhosMap, three kinase activity prediction methods were included: KSEA, multiple linear regression (MLR) and Mean Value.

Data preparation

```
# For early and late response
# early -> clusterS_info==1
# late -> clusterS_info==2
cluster_flag <- 'early'
cluster_symbol <- clusterS_names[clusterS_info==1]
expr_data_frame <- data_frame_normalization_with_control_no_pair
```

```
index_of_cluster <- match(cluster_symbol, expr_data_frame$ID)
cluster_df <- expr_data_frame[index_of_cluster,]
```

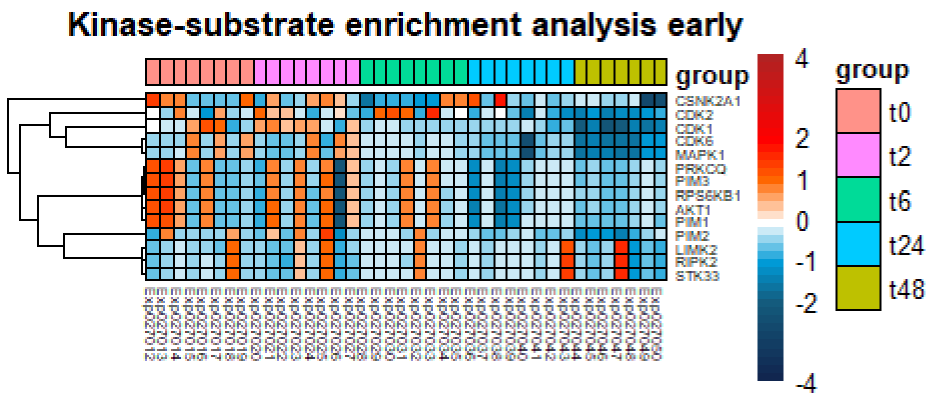
KSEA

```
# Perform KSEA
summary_df_list_from_ksea_cluster <- get_summary_from_ksea(cluster_df, species = 'human',
                                                           log2_label = FALSE, ratio_cutoff = 3)
```

```
##
##  Starting KSEA
##  completing:  1 / 39
##  completed:  1 / 39
##  completing:  2 / 39
##  completed:  2 / 39
##  completing:  3 / 39
##  completed:  3 / 39
##  completing:  4 / 39
##  completed:  4 / 39
##  completing:  5 / 39
##  completed:  5 / 39
##  completing:  6 / 39
##  completed:  6 / 39
##  completing:  7 / 39
##  completed:  7 / 39
##  completing:  8 / 39
##  completed:  8 / 39
##  completing:  9 / 39
##  completed:  9 / 39
##  completing:  10 / 39
##  completed:  10 / 39
##  completing:  11 / 39
##  completed:  11 / 39
##  completing:  12 / 39
##  completed:  12 / 39
##  completing:  13 / 39
##  completed:  13 / 39
##  completing:  14 / 39
##  completed:  14 / 39
##  completing:  15 / 39
##  completed:  15 / 39
##  completing:  16 / 39
##  completed:  16 / 39
##  completing:  17 / 39
##  completed:  17 / 39
##  completing:  18 / 39
##  completed:  18 / 39
##  completing:  19 / 39
##  completed:  19 / 39
##  completing:  20 / 39
##  completed:  20 / 39
##  completing:  21 / 39
##  completed:  21 / 39
##  completing:  22 / 39
##  completed:  22 / 39
##  completing:  23 / 39
##  completed:  23 / 39
##  completing:  24 / 39
##  completed:  24 / 39
##  completing:  25 / 39
##  completed:  25 / 39
##  completing:  26 / 39
##  completed:  26 / 39
##  completing:  27 / 39
##  completed:  27 / 39
##  completing:  28 / 39
##  completed:  28 / 39
##  completing:  29 / 39
##  completed:  29 / 39
##  completing:  30 / 39
##  completed:  30 / 39
##  completing:  31 / 39
##  completed:  31 / 39
##  completing:  32 / 39
##  completed:  32 / 39
##  completing:  33 / 39
##  completed:  33 / 39
##  completing:  34 / 39
##  completed:  34 / 39
##  completing:  35 / 39
##  completed:  35 / 39
##  completing:  36 / 39
##  completed:  36 / 39
##  completing:  37 / 39
##  completed:  37 / 39
##  completing:  38 / 39
##  completed:  38 / 39
##  completing:  39 / 39
##  completed:  39 / 39
##  Ending KSEA
##  Extracting information data frame derived from KSEA
##  ********** Regulation direction from KSEA **********
##  ********** Pvalue from KSEA **********
##  ********** Activity from KSEA **********
##  ********** Kinase_site_substrate quantification matrix after KSEA **********
##
##  KSEA OK! ^_^
```

```
# Activity of regulons for regulation
ksea_regulons_activity_df_cluster <- summary_df_list_from_ksea_cluster$ksea_regulons_activity_df
ksea_id_cluster <- as.vector(ksea_regulons_activity_df_cluster[,1])
ksea_value_cluster <- ksea_regulons_activity_df_cluster[,-1]
if(FALSE){
  # Pvalue of regulons for regulation
  ksea_regulons_pvalue_cluster <- summary_df_list_from_ksea_cluster$ksea_regulons_pvalue_df
  # Activity of regulons for regulation
  ksea_regulons_activity_cluster <- summary_df_list_from_ksea_cluster$ksea_regulons_activity_df
  # Expression ratio of regulons for regulation
  ksea_kinase_site_substrate_original_ratio_cluster <-
    summary_df_list_from_ksea_cluster$ksea_kinase_site_substrate_original_ratio_df
}
```

```r
# plot pheatmap
if(TRUE){
  # annotation setting
  annotation_col <- data.frame(
    group = group
  )
  rownames(annotation_col) <- colnames(ksea_value_cluster)

  # breaks and colors setting
  breaks_1 <- seq(-4, -2, 0.2)
  colors_1 <- colorRampPalette(c('#11264f', '#145b7d'))(length(breaks_1)-1)

  breaks_2 <- seq(-2, -1, 0.2)
  colors_2 <- colorRampPalette(c('#145b7d', '#009ad6'))(length(breaks_2))

  breaks_3 <- seq(-1, 1, 0.2)
  colors_3 <- colorRampPalette(c('#009ad6', 'white', '#FF6600'))(length(breaks_3))

  breaks_4 <- seq(1, 2, 0.2)
  colors_4 <- colorRampPalette(c('#FF6600', 'red'))(length(breaks_4))

  breaks_5 <- seq(2, 4, 0.2)
  colors_5 <- colorRampPalette(c('red', 'firebrick'))(length(breaks_5))

  breaks <- c(breaks_1, breaks_2, breaks_3, breaks_4, breaks_5)
  breaks <- breaks[which(!duplicated(breaks))]
  color <- c(colors_1, colors_2, colors_3, colors_4, colors_5)
  color <- color[which(!duplicated(color))]
  library(pheatmap)
  ph <- pheatmap(
    ksea_value_cluster,
    scale = 'none',
    annotation_col = annotation_col,
    clustering_distance_rows = 'euclidean',
    fontsize_row = 5,
    # cutree_rows = 1,
    show_rownames = TRUE,
    fontsize_col = 5,
    # cutree_cols = 1,
    cluster_cols = FALSE,
    border_color = 'black',
    cellwidth = 5, cellheight = 5,
    breaks = breaks,
    color = color,
    legend_breaks = c(-4, -2, -1, 0, 1, 2, 4),
    legend_labels = c(-4, -2, -1, 0, 1, 2, 4),
    main = paste('Kinase-substrate enrichment analysis', cluster_flag, sep = ' ')
  )
}
```



KSEA method

MLR

```r
# get kinase activity matrix with multiple linear regression (mlr) method
kinase_activity_df_mlr <- get_ka_by_mean_or_mlr(cluster_df, species = 'human',
                                                log2_label = TRUE, method = 'mlr')
```

Mean value

```r
# get kinase activity matrix with mean value method
kinase_activity_df_mean <- get_ka_by_mean_or_mlr(cluster_df, species = 'human',
                                                 log2_label = TRUE, method = 'mean')
```

Motif enrichment analysis (MEA)

PhosMap allowed for performing MEA on user defined phosphopeptides lists.

Data preparation

```r
# *** foreground ***
foreground_data <- phospho_data_filtering_STY_and_normalization # pre-processed data
foreground_sequence <- as.vector(foreground_data$Sequence)
GI <- as.vector(foreground_data$GI)
```

```
Sequence <- as.vector(foreground_data$Sequence)
AA_in_protein <- as.vector(foreground_data$AA_in_protein)

# *** required parameters ***
fixed_length <- 15
species <- 'human'
motifx_pvalue <- 0.01

# get foreground data frame
# foreground_df = get_aligned_seq_for_mea(ID, Sequence, AA_in_protein, fixed_length,
#                                         species = 'human', fasta_type = 'refseq')
# get background data frame
# background_df = get_global_background_df(species = 'human', fasta_type = 'refseq')
```

Motif enrichment analysis

```
# construct foreground and background
# To facilitate testing the module, select an appropriate number of items at random.
foreground <- as.vector(foreground_df$aligned_seq)
foreground <- foreground[sample(length(foreground), 1000)]
background <- as.vector(background_df$Aligned_Seq)
background <- background[sample(length(background), 10000)]



motifs_list <- mea_based_on_background(foreground, AA_in_protein, background, motifx_pvalue)
```

```
## Start executing motifx and find motif pattern.
## Foreground sequences: 1000.
## Background sequences: 10000.
## Phosphorylation: [STY] exists in foreground.
## Motifx pvalue cutoff: 0.01.
## Motifx analysis OK! ^_^
## $S
##                motif     score foreground_matches foreground_size
## 1   .......SPS....R 37.664847                  7             820
## 2   ....R..SPT..... 37.282931                  7             813
## 3   ..K...SSP...... 36.564902                 11             806
## 4   ......TSPS..... 35.837518                  5             795
## 5   ....K..SP....E. 35.615086                  6             790
## 6   .R.....SP.....R 35.707569                  5             784
## 7   ....P..SPT..... 35.721156                  6             779
## 8   .......SP...S.E 35.783941                  6             773
## 9   .S.A...SP...... 34.973447                  5             767
## 10  ...E...SP.K.... 34.860592                  5             762
## 11  .....P.SP.....A 34.925900                  5             757
## 12  G..R...SP...... 34.859468                  5             752
## 13  ...LS..SP...... 34.738924                  7             747
## 14  .....P.SPT..... 34.718201                  4             740
## 15  .....S.SP.....Q 34.837585                  7             736
## 16  A..K...SP...... 34.632052                  4             729
## 17  ..S...KSP...... 34.655472                  6             725
## 18  ....R..SP.P.... 34.560902                  7             719
## 19  ...A...SP..T... 34.378505                  4             712
## 20  .......SP.S...R 34.418713                  5             708
## 21  P.....TSP...... 34.499510                  3             703
## 22  ...R...SPE..... 34.216657                  4             700
## 23  ......SSP.G.... 34.289137                  6             696
## 24  E......SP...S.. 34.339147                  4             690
## 25  ...N...SP.....K 34.356690                  4             686
## 26  ..E....SP.....P 34.251677                  4             682
## 27  K..E...SP...... 34.165072                  6             678
## 28  ..V....SPT..... 34.090857                  3             672
## 29  .......SP...... 16.000000                190             669
## 30  ...MR..S....... 32.000000                  2             479
## 31  ....RS.S.A..... 48.000000                  6             477
## 32  ....R..S......W 32.000000                  2             471
## 33  ....RSKS....... 45.496708                  5             469
## 34  ....R.WS....... 32.000000                  1             464
## 35  ....RR.SF...... 36.446295                  4             463
## 36  ..L.RA.S....... 36.508762                  7             459
## 37  ....RS.S...V... 37.129222                  4             452
## 38  ....RT.S..S.... 36.291496                  4             448
## 39  ....RS.S.....Q. 35.958837                  4             444
## 40  .K..R..S....S.. 35.648764                  3             440
## 41  V...R..S..S.... 35.084045                  2             437
## 42  ...TR..S..S.... 34.448679                  2             435
## 43  ....RR.SQ...... 34.160552                  5             433
## 44  ..A....S.DE.... 40.964230                  4             428
## 45  ....RS.S......N 32.649943                  2             424
## 46  L......SD.E.... 38.885971                  3             422
## 47  ....R..SA...A.. 31.760446                  5             419
## 48  ......GS.DE.... 36.252140                  4             414
## 49  ......DSE.E.... 33.351385                  5             410
## 50  ..L.R.QS....... 29.847178                  3             405
## 51  P....SGS....... 30.607105                  4             402
## 52  ....R..S....... 10.639735                 57             398
## 53  ..R....SD.E.... 32.149045                  2             341
## 54  ..N....SD.E.... 30.234177                  2             339
## 55  .....D.S.EE.... 29.154015                  3             337
## 56  ..R..S.S..N.... 28.713005                  2             334
## 57  ..R....SE.E.... 28.192383                  3             332
## 58  .R.....S.DE.... 27.229618                  2             329
## 59  A...GS.S....... 26.590274                  3             327
## 60  K......S.D...S. 25.922776                  2             324
## 61  C......SD.E.... 26.531174                  1             322
## 62  ..K.LS.S....... 25.635262                  3             321
## 63  .R.....SD.E.... 25.784727                  1             318
## 64  .......SGEE.... 25.454843                  2             317
## 65  .R...S.S.....G. 24.900026                  3             315
## 66  .....R.S.SP.... 25.576561                  3             312
## 67  ..H....S.D..N.. 24.291578                  2             309
## 68  ..EE...S..E.... 25.068986                  2             307
## 69  T.R..S.S....... 24.169979                  1             305
## 70  .K....KS.S..... 25.068519                  3             304
## 71  Q...S..S.D..... 23.907708                  3             301
## 72  ...D..ES.E..... 24.031658                  2             298
## 73  .......SE.E.Q.. 24.128510                  3             296
## 74  YR.S...S....... 24.863189                  2             293
## 75  .......SDS..Q.. 23.671932                  3             291
## 76  ..ES...SA...... 24.288622                  3             288
## 77  .......S.DE.P.. 23.179855                  2             285
## 78  ....S.DS..E.... 24.055972                  3             283
## 79  ..V....S.S..D.. 23.041296                2             280
## 80  .......S..G....  4.616931               36             278
## 81  .S.....S.ED.... 23.555099                3             242
## 82  R....S.S.....I. 23.592586                1             239
## 83  K....S.S......P 23.162563                3             238
## 84  ....K..S.E...E. 22.457194                3             235
## 85  ...AS..S.V..... 23.264145                2             232
## 86  .......S.EE..A. 22.542927                2             230
## 87  ....Q.SS..S.... 23.200258                3             228
## 88  ..R.A..S.S..... 22.603529                3             225
## 89  ...RS..SD...... 22.515248                3             222
## 90  ...AS..S..E.... 22.098104                2             219
## 91  H....S.S.....I. 21.930574                1             217
## 92  .....EDS.D..... 21.744894                3             216
## 93  .N...S.S..F.... 21.582617                2             213
## 94  ..M....SD.E.... 21.782518                1             211
## 95  ...H...SL.S.... 21.401080                2             210
## 96  .......S..E....  3.277582               32             208
## 97  .......S..S....  3.546496               40             176
## 98  ...K.W.S....... 19.115805                1             136
## 99  ....K..S......W 18.844808                1             135
## 100 .R.K...S......P 22.197498                2             134
## 101 QS....DS....... 20.950690                2             132
## 102 .KS....S.S..... 21.369730                3             130
## 103 ....K..SLT..... 21.335552                2             127
## 104 QR.S...S....... 21.099034                1             125
## 105 K.HK...S....... 20.811626                1             124
## 106 K....S.S.....I. 20.843039                1             123
## 107 ....KR.S.....S. 21.740975                2             122
## 108 .V....GS..D.... 20.948026                2             120
## 109 ....Q..SSP..... 21.937718                2             118
## 110 R...S..S.P..... 20.896746                2             116
## 111 ...S...S.A..S.. 20.640245                2             114
## 112 .......S.S.....  2.345906               21             112
## 113 .......S.....K.  2.271358               12              91
## 114 .......S.DI...I 21.039996                2              79
## 115 E......SSP..... 21.349231                1              77
## 116 ....K..S..M.... 18.387706                1              76
## 117 ...S...S....AG. 20.583493                2              75
## 118 ...R...SS.....S 21.074320                2              73
## 119 .T....ES.....S. 20.433519                2              71
## 120 .......S...S...  2.149138               14              69
## 121 .......S......E  2.051214                9              55
## 122 .....S.S.......  2.566943               11              46
## 123 .......SDN..... 18.155014                2              35
##     background_matches background_size fold_increase
## 1                    0            5092           Inf
## 2                    0            5092           Inf
## 3                    0            5092           Inf
## 4                    0            5092           Inf
## 5                    0            5092           Inf
## 6                    0            5092           Inf
## 7                    0            5092           Inf
## 8                    0            5092           Inf
## 9                    0            5092           Inf
## 10                   0            5092           Inf
## 11                   0            5092           Inf
## 12                     0            5092          Inf
## 13                     0            5092          Inf
## 14                     0            5092          Inf
## 15                     0            5092          Inf
## 16                     0            5092          Inf
## 17                     0            5092          Inf
## 18                     0            5092          Inf
## 19                     0            5092          Inf
## 20                     0            5092          Inf
## 21                     0            5092          Inf
## 22                     0            5092          Inf
## 23                     0            5092          Inf
## 24                     0            5092          Inf
## 25                     0            5092          Inf
## 26                     0            5092          Inf
## 27                     0            5092          Inf
## 28                     0            5092          Inf
## 29                   336            5092     4.304043
## 30                     0            4756          Inf
## 31                     0            4756          Inf
## 32                     0            4756          Inf
## 33                     0            4756          Inf
## 34                     0            4756          Inf
## 35                     0            4756          Inf
## 36                     0            4756          Inf
## 37                     0            4756          Inf
## 38                     0            4756          Inf
## 39                     0            4756          Inf
## 40                     0            4756          Inf
## 41                     0            4756          Inf
## 42                     0            4756          Inf
## 43                     0            4756          Inf
## 44                     0            4756          Inf
## 45                     0            4756          Inf
## 46                     0            4756          Inf
## 47                     0            4756          Inf
## 48                     0            4756          Inf
## 49                     0            4756          Inf
## 50                     0            4756          Inf
## 51                     0            4756          Inf
## 52                   254            4756     2.681637
## 53                     0            4502          Inf
## 54                     0            4502          Inf
## 55                     0            4502          Inf
## 56                     0            4502          Inf
## 57                     0            4502          Inf
## 58                     0            4502          Inf
## 59                     0            4502          Inf
## 60                     0            4502          Inf
## 61                     0            4502          Inf
## 62                     0            4502          Inf
## 63                     0            4502          Inf
## 64                     0            4502          Inf
## 65                     0            4502          Inf
## 66                     0            4502          Inf
## 67                     0            4502          Inf
## 68                     0            4502          Inf
## 69                      0            4502           Inf
## 70                      0            4502           Inf
## 71                      0            4502           Inf
## 72                      0            4502           Inf
## 73                      0            4502           Inf
## 74                      0            4502           Inf
## 75                      0            4502           Inf
## 76                      0            4502           Inf
## 77                      0            4502           Inf
## 78                      0            4502           Inf
## 79                      0            4502           Inf
## 80                    276            4502      2.112293
## 81                      0            4226           Inf
## 82                      0            4226           Inf
## 83                      0            4226           Inf
## 84                      0            4226           Inf
## 85                      0            4226           Inf
## 86                      0            4226           Inf
## 87                      0            4226           Inf
## 88                      0            4226           Inf
## 89                      0            4226           Inf
## 90                      0            4226           Inf
## 91                      0            4226           Inf
## 92                      0            4226           Inf
## 93                      0            4226           Inf
## 94                      0            4226           Inf
## 95                      0            4226           Inf
## 96                    348            4226      1.868258
## 97                    501            3878      1.759209
## 98                      0            3377           Inf
## 99                      0            3377           Inf
## 100                     0            3377           Inf
## 101                     0            3377           Inf
## 102                     0            3377           Inf
## 103                     0            3377           Inf
## 104                     0            3377           Inf
## 105                     0            3377           Inf
## 106                     0            3377           Inf
## 107                     0            3377           Inf
## 108                     0            3377           Inf
## 109                     0            3377           Inf
## 110                     0            3377           Inf
## 111                     0            3377           Inf
## 112                   343            3377      1.846028
## 113                   169            3034      2.367384
## 114                     0            2865           Inf
## 115                     0            2865           Inf
## 116                     0            2865           Inf
## 117                     0            2865           Inf
## 118                     0            2865           Inf
## 119                     0            2865           Inf
## 120                   280            2865      2.076087
## 121                   165            2585      2.563636
## 122                   224            2420      2.583463
## 123                     0            2196           Inf
##
## $T
##              motif     score foreground_matches foreground_size
## 1  D....P.TP...... 36.676650                  3             178
## 2  ..A.G..TP...... 36.251038                  4             175
## 3  ..RA...TP...... 35.905342                  3             171
## 4  E....A.TP...... 35.896627                  3             168
## 5  .......TPP..R.. 36.099525                  4             165
## 6  ..V....TP.K.... 36.116506                  3             161
## 7  ....KP.TP...... 34.921233                  3             158
## 8  A......TP...N.. 34.437500                  2             155
## 9  S......TP....R. 34.416099                  4             153
## 10 .....EKTP...... 34.615805                  5             149
## 11 E......TP.K.... 31.621434                  2             144
## 12 .......TP..SR.. 30.375660                  3             142
## 13 ..R..S.T.D..... 32.075889                  3             139
## 14 ....G..TPP..... 28.623742                  3             136
## 15 A....SDT....... 29.743913                  3             133
## 16 .......TP......  9.258404                 33             130
## 17 ..R..SAT....... 27.376021                  2              97
## 18 ...R.S.T.....E. 25.158896                  2              95
## 19 IS...S.T....... 23.941215                  2              93
## 20 ...N.S.T.P..... 23.075485                  2              91
## 21 ...R.S.T.S..... 23.052091                  2              89
## 22 .R.SS..T....... 23.955142                  2              87
## 23 G..Q...T..S.... 23.835249                  2              85
## 24 ......ST..L...D 23.364351                  3              83
## 25 R...S..T.S..... 22.590859                  2              80
## 26 .D.....T.P....R 22.392211                  2              78
## 27 .....SDT....G.. 21.769318                  2              76
## 28 ......ST..SE... 22.902286                  3              74
## 29 .......TE.E.D.. 22.121802                  2              71
## 30 .......T.SP...K 21.798585                  3              69
## 31 ..SI...T...S... 22.174881                  2              66
## 32 ..EK...T.P..... 21.653617                  2              64
## 33 ....SP.T.W..... 22.325458                  2              62
## 34 .......T.S.SD.. 20.945402                  2              60
## 35 ...H..GT....... 18.484326                  1              58
## 36 ......ET.....DG 20.833959                  2              57
## 37 ..I...DT..E.... 21.627701                  2              55
## 38 A.....GTA...... 20.710249                  1              53
## 39 .....S.T.......  2.176177                 11              52
## 40 ....S.ST.....M. 20.995448                  2              41
## 41 A.....ST....D.. 20.315962                  1              39
## 42 .......T.M..P.. 18.050606                  1              38
## 43 .N.F.K.T....... 20.790071                  2              37
## 44 ...SS.GT....... 21.044632                  2              35
##    background_matches background_size fold_increase
## 1                   0            3280           Inf
## 2                   0            3280           Inf
## 3                   0            3280           Inf
## 4                   0            3280           Inf
## 5                   0            3280           Inf
## 6                   0            3280           Inf
## 7                   0            3280           Inf
## 8                   0            3280           Inf
## 9                   0            3280           Inf
## 10                  0            3280           Inf
## 11                  0            3280           Inf
## 12                   0            3280          Inf
## 13                   0            3280          Inf
## 14                   0            3280          Inf
## 15                   0            3280          Inf
## 16                 249            3280     3.343837
## 17                   0            3031          Inf
## 18                   0            3031          Inf
## 19                   0            3031          Inf
## 20                   0            3031          Inf
## 21                   0            3031          Inf
## 22                   0            3031          Inf
## 23                   0            3031          Inf
## 24                   0            3031          Inf
## 25                   0            3031          Inf
## 26                   0            3031          Inf
## 27                   0            3031          Inf
## 28                   0            3031          Inf
## 29                   0            3031          Inf
## 30                   0            3031          Inf
## 31                   0            3031          Inf
## 32                   0            3031          Inf
## 33                   0            3031          Inf
## 34                   0            3031          Inf
## 35                   0            3031          Inf
## 36                   0            3031          Inf
## 37                   0            3031          Inf
## 38                   0            3031          Inf
## 39                 277            3031     2.314704
## 40                   0            2754          Inf
## 41                   0            2754          Inf
## 42                   0            2754          Inf
## 43                   0            2754          Inf
## 44                   0            2754          Inf
```

```
# Find sequences in foreground that are mapped to specific motif
foreground_sequences_mapped_to_motifs <- get_foreground_seq_to_motifs(motifs_list, foreground)
```

```
##
##  Find sequences in foreground that are mapped to specific motif.
```

```
# Find data frame in foreground that are mapped to specific motif
foreground_df_mapped_to_motifs <- get_foreground_df_to_motifs(foreground_sequences_mapped_to_motifs,
                                                              foreground, foreground_df)
```

```
##
##  Find data frame in foreground that are mapped to specific motif.
```

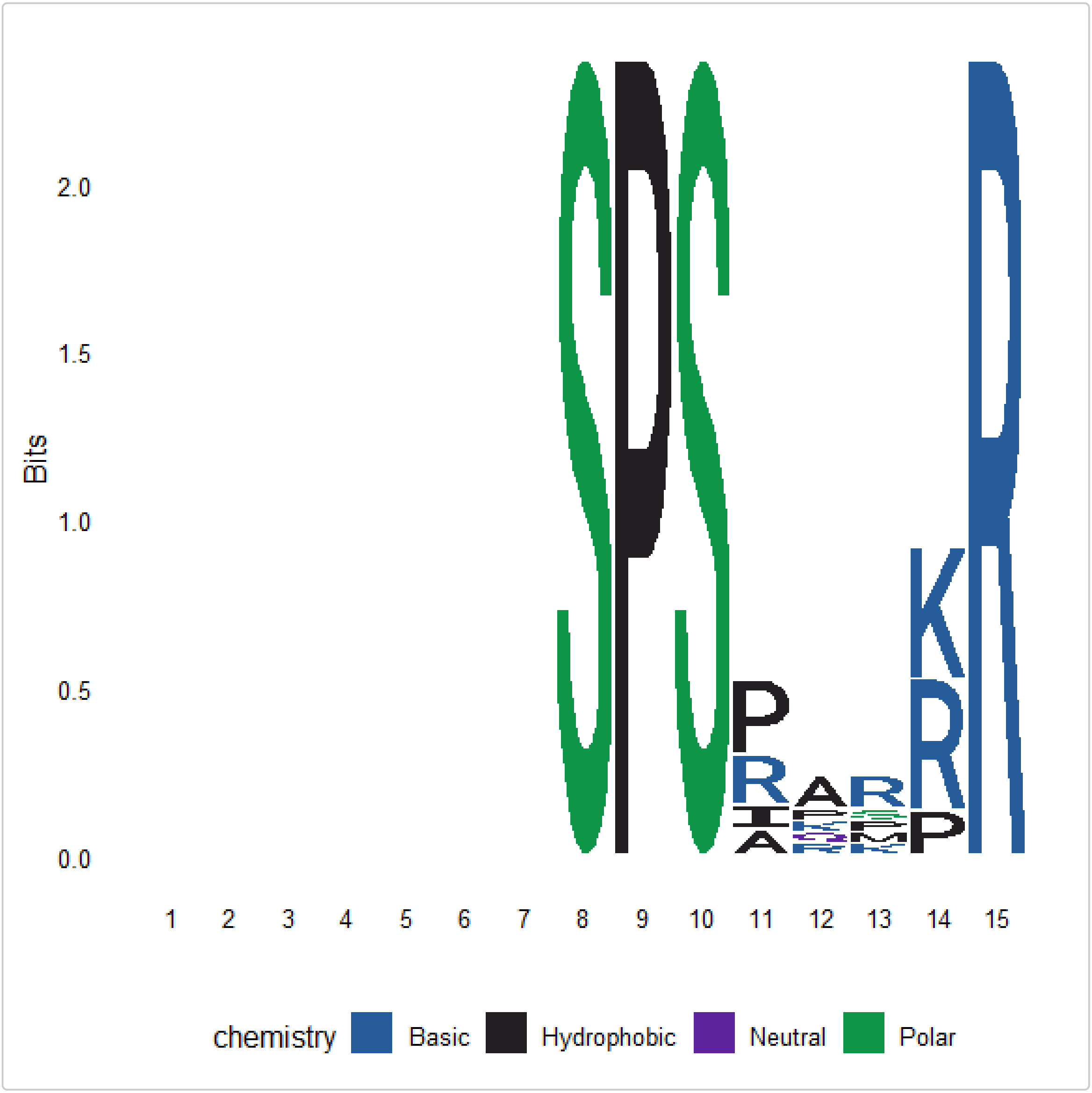
Plot motif logo

```
# The data can be used for ploting logo of sepcific motif: foreground_sequences_mapped_to_motifs
# ploting logo: Q......S.......
require(ggseqlogo)
```

```
## Loading required package: ggseqlogo
```

```
## Registered S3 methods overwritten by 'ggplot2':
##   method         from 
##   [.quosures     rlang
##   c.quosures     rlang
##   print.quosures rlang
```

```
ggseqlogo(foreground_sequences_mapped_to_motifs[[1]])
```

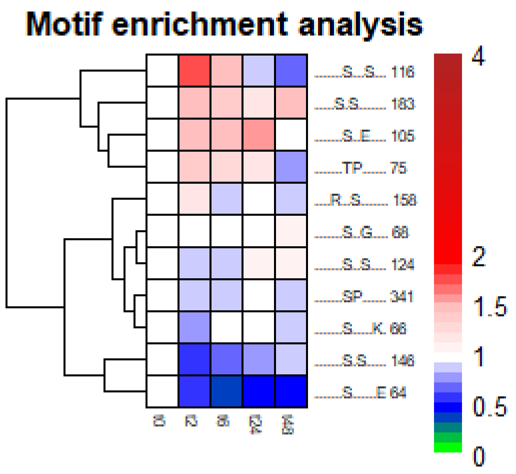


Plot motif logo

```
if(TRUE){
  # batch plot and count peptides for each motif
  foreground_sequences_mapped_to_motifs_count <- length(foreground_sequences_mapped_to_motifs)
  motifs <- names(foreground_sequences_mapped_to_motifs)
  peptides_count <- NULL
  for(i in seq_len(foreground_sequences_mapped_to_motifs_count)){
    l_i <- foreground_sequences_mapped_to_motifs[[i]]
    peptides_count <- c(peptides_count, length(l_i))
  }
  motifs_peptides_count_df <- data.frame(motifs, peptides_count)
  # quantile(peptides_count, seq(0,1,0.05))
  if(FALSE){
    plot_seqlogo(BASE_DIR, foreground_sequences_mapped_to_motifs, plot_min_seqs = 25)
  }

}
```

Assign quantitative values of peptides to their motif

```
# Select motifs at least having 50 peptides
# Assign quantitative values of peptides to their motif
foreground_value <- foreground_data[,-c(seq(1,6))]
min_seqs <- 50
index_of_motifs <- which(peptides_count>=min_seqs)
motif_group_m_ratio_df <- NULL
for(i in index_of_motifs){
  motif <- motifs[i]
  aligned_peptides <- foreground_sequences_mapped_to_motifs[[i]]
  index_of_match <- match(aligned_peptides, foreground_df$aligned_seq)
  motif_value <- foreground_value[index_of_match,]
  motif_value_colsum <- colSums(motif_value)
  motif_group_m <- tapply(motif_value_colsum, group, mean)
  motif_group_m_ratio <- motif_group_m/motif_group_m[1]
  motif_group_m_ratio_df <- rbind(motif_group_m_ratio_df, motif_group_m_ratio)
}
motifs_subset <- motifs[index_of_motifs]
peptides_count_subset <- peptides_count[index_of_motifs]
rownames(motif_group_m_ratio_df) <- paste(motifs_subset, peptides_count_subset)

# The matrix is import inot pheatmap
motif_group_m_ratio_df_mat <- as.matrix(motif_group_m_ratio_df)



# plot pheatmap
if(TRUE){
  library(pheatmap)
  # breaks and colors setting
  breaks_1 <- seq(0, 0.5, 0.1)
  colors_1 <- colorRampPalette(c('green', 'blue'))(length(breaks_1)-1)

  breaks_3 <- seq(0.5, 1.5, 0.1)
  colors_3 <- colorRampPalette(c('blue', 'white', '#FFBFBF'))(length(breaks_3))

  breaks_4 <- seq(1.5, 2, 0.1)
  colors_4 <- colorRampPalette(c('#FFBFBF', 'red'))(length(breaks_4))

  breaks_5 <- seq(2, 4, 0.1)
  colors_5 <- colorRampPalette(c('red','firebrick'))(length(breaks_5))

  breaks <- c(breaks_1, breaks_3, breaks_4, breaks_5)
  breaks <- breaks[which(!duplicated(breaks))]
  colors <- c(colors_1, colors_3, colors_4, colors_5)
  colors <- colors[which(!duplicated(colors))]

  length(breaks)
  length(which(!duplicated(colors)))
  ph <- pheatmap(
    motif_group_m_ratio_df_mat,
    scale = 'none',
    # annotation_col = annotation_col,
    clustering_distance_cols = 'euclidean',
    fontsize_row = 6, cutree_rows = 1, show_rownames = TRUE, cluster_rows = TRUE,
    fontsize_col = 6, cutree_cols = 1, show_colnames = TRUE, cluster_cols = FALSE,
    border_color = 'black',
    # color = colors,
    cellwidth = 12, cellheight = 12,
    breaks = breaks,
    color = colors,
    legend_breaks = c(0, 0.5, 1, 1.5, 2, 4),
    legend_labels = c(0, 0.5, 1, 1.5, 2, 4),
    main = 'Motif enrichment analysis'
  )
}
```



KSEA method

Formatting output

```
# formatting output
formatted_output_df <- formatted_output_mef_results(foreground_sequences_mapped_to_motifs)
```

```
## Output formatted sequences in foreground that are mapped to specific motif.
```

```
# write file
# write.table(formatted_output_df, 'formatted_output_df.txt', row.names = FALSE,
#             col.names = FALSE, sep = '\t')
```

Session Info

```
sessionInfo()
```

```
## R version 3.6.0 (2019-04-26)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 17134)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=Chinese (Simplified)_China.936
## [2] LC_CTYPE=Chinese (Simplified)_China.936
## [3] LC_MONETARY=Chinese (Simplified)_China.936
## [4] LC_NUMERIC=C
## [5] LC_TIME=Chinese (Simplified)_China.936
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] ggseqlogo_0.1   pheatmap_1.0.12 PhosMap_0.99.33
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.1         compiler_3.6.0     pillar_1.4.1
##  [4] RColorBrewer_1.1-2 highr_0.8          plyr_1.8.4
##  [7] bitops_1.0-6       class_7.3-15       tools_3.6.0
## [10] digest_0.6.19      evaluate_0.14      Rtsne_0.15
## [13] tibble_2.1.3       gtable_0.3.0       pkgconfig_2.0.2
## [16] rlang_0.3.4        yaml_2.2.0         parallel_3.6.0
## [19] xfun_0.7           e1071_1.7-2        dplyr_0.8.1
## [22] stringr_1.4.0      knitr_1.23         tidyselect_0.2.5
## [25] grid_3.6.0         glue_1.3.1         impute_1.58.0
## [28] R6_2.4.0           rmarkdown_1.13     limma_3.40.2
## [31] purrr_0.3.2        ggplot2_3.1.1      ClueR_1.4
## [34] magrittr_1.5       scales_1.0.0       htmltools_0.3.6
## [37] assertthat_0.2.1   colorspace_1.4-1   labeling_0.3
## [40] stringi_1.4.3      RCurl_1.95-4.12    lazyeval_0.2.2
## [43] munsell_0.5.0      crayon_1.3.4
```