

# Application Qian 2001 method on Selevsek 2015 data (Z-normalize by genes)

*s\_morimoto*

2016/07/01

This source code is built under R ver.3.1.2

You are using R version 3.2.2 (2015-08-14)

The transcriptome data are taken in *Lee,2011*, described in *Selevsek,2015*. The data has 6 time points(T1~T6), but T6(240[min]) is twice of T5(120[min]) so discarded from analysis.

The proteome data has 6 time points. This data has T2=15[min] in which timepoint was not taken data in *Lee,2011*. To establish consistency to transcriptome data, deleted T2(15[min])data, and rename T3 to T2, T4 to T3, and so on.

SETUP parameters

```
rm()

dataDirectry <- "/Users/mos/Dropbox/Draft_201603Morimoto/Analysis/Data"

dataPath_trc <- "Gene2011_testResult_final.xlsx" #
dataPath_prt <- "ProtSWATH_testResult_final.xlsx" #

sheet = 1
header=TRUE

## filenames of the OUTPUT FILES

outputDirectry <- "/Users/mos/Dropbox/Draft_201603Morimoto/Analysis/Data"

dataOutput_trc.prefix <- "Gene2011_testResult_ana_20160706" #
dataOutput_prt.prefix <- "ProtSWATH_testResult_ana_20160706" #

file_dataGQ <- 'dataGQ_20160706.csv'

file_save.image <- 'Selevsek_Qian_20160706.RData'
```

LOAD PACKAGES

```
## Loading required package: plyr
## Loading required package: dplyr
##
## Attaching package: 'dplyr'
##
##      'package:plyr'      :
##
##      arrange, count, desc, failwith, id, mutate, rename, summarise,
##      summarize
##
```

```
##      'package:stats'      :
##
##      filter, lag
##
##      'package:base'      :
##
##      intersect, setdiff, setequal, union
##
## Loading required package: tidy
## Loading required package: xlsx
## Loading required package: rJava
## Loading required package: xlsxjars
## Loading required package: readxl
## Loading required package: ggplot2
## Loading required package: reshape2
## Loading required package: pander
## Loading required package: stringr
```

## LOAD DATA

```
raw_data.trc <- read_excel(
  sprintf(fmt = '%s/%s',
    dataDirectory ,
    dataPath_trc
  ),
  sheet,
  col_names = header,
  col_types = NULL,
  na = "", skip = 0
)[ ,2:4]

raw_data.prt <- read_excel(
  sprintf(fmt = '%s/%s',
    dataDirectory ,
    dataPath_prt
  ),
  sheet,
  col_names = header,
  col_types = NULL,
  na = "", skip = 0
)[ ,2:4]

data.trc <- data.frame(Protein=unique(raw_data.trc$Protein),
  log2FC=0,
  Label='T1-T1'
) %>%
full_join(raw_data.trc) %>%
mutate(
  id=as.character(Protein),
  var=as.character(Label),
  val=as.numeric(log2FC),
  dtname='trc'
) %>%
mutate(
```

```

var =
  ifelse(Label=='T1-T1','0min',
    ifelse(Label=='T2-T1','30min',
      ifelse(Label=='T3-T1','60min',
        ifelse(Label=='T4-T1','90min',
          ifelse(Label=='T5-T1','120min',
            ifelse(Label=='T6-T1','240min',NA)
          )
        )
      )
    )
  )
) %>%
dplyr::select(id,var,val,dtname)

```

```
## Joining by: c("Protein", "log2FC", "Label")
```

```
## Warning in outer_join_impl(x, y, by$x, by$y): joining character vector and
## factor, coercing into character vector
```

```
## Warning in outer_join_impl(x, y, by$x, by$y): joining character vector and
## factor, coercing into character vector
```

```

data.prt <- data.frame(Protein=unique(raw_data.prt$Protein),
  log2FC=0,
  Label='T1-T1'
) %>%
full_join(raw_data.prt) %>%
mutate(
  id=as.character(Protein),
  val=as.numeric(log2FC),
  dtname='prt'
) %>%
mutate(
  var =
    ifelse(Label=='T1-T1','0min',
      ifelse(Label=='T2-T1','15min',
        ifelse(Label=='T3-T1','30min',
          ifelse(Label=='T4-T1','60min',
            ifelse(Label=='T5-T1','90min',
              ifelse(Label=='T6-T1','120min',NA)
            )
          )
        )
      )
    )
  )
) %>%
dplyr::select(id,var,val,dtname)

```

```
## Joining by: c("Protein", "log2FC", "Label")
```

```
## Warning in outer_join_impl(x, y, by$x, by$y): joining character vector and
## factor, coercing into character vector
```

```
## Warning in outer_join_impl(x, y, by$x, by$y): joining character vector and
## factor, coercing into character vector
```

```
# This chunk is added 2016/7/1
```

```
data.trc <- data.trc %>%
  mutate(val = 2^val)

data.prt <- data.prt %>%
  mutate(val = 2^val)
```

```
write.xlsx(
  data.trc,
  sprintf(fmt = '%s/%s.xlsx',
    outputDirectry,
    dataOutput_trc.prefix
  )
)
```

```
write.xlsx(
  data.prt,
  sprintf(fmt = '%s/%s.xlsx',
    outputDirectry,
    dataOutput_prt.prefix
  )
)
```