Metabolic_Distribution

Emmi Mueller

March 8, 2019

#Require vegan, fitdistrplus, actuar, and BiocManager with flowFCS

```
rm(list = ls())
getwd()
```

```
## [1] "C:/Users/emmim/GitHub/MetabolicDistribution"
```

```
package.list <- c('BiocManager', 'vegan', 'fitdistrplus', 'ggplot2', 'car', 'here', 'ggcyto', 'actuar',

for (package in package.list) {
  if (!require(package, character.only=T, quietly=T)) {
    install.packages(package, dependencies = TRUE)
    library(package, character.only=T)
  } }
```

```
## This is vegan 2.5-7
```

```
## here() starts at C:/Users/emmim/GitHub/MetabolicDistribution
```

```
## As part of improvements to flowWorkspace, some behavior of
## GatingSet objects has changed. For details, please read the section
## titled "The cytoframe and cytoset classes" in the package vignette:
##
##   vignette("flowWorkspace-Introduction", "flowWorkspace")
```

```
##
## Attaching package: 'actuar'
```

```
## The following object is masked from 'package:grDevices':
##
##     cm
```

```
BiocManager::valid()
```

```
## Warning: 4 packages out-of-date; 0 packages too new
```

```
##
## * sessionInfo()
##
## R version 4.1.0 (2021-05-18)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19042)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
```

```
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] scales_1.1.1             ggpubr_0.4.0             tibble_3.1.2
##  [4] actuar_3.1-4             ggcyto_1.20.0            flowWorkspace_4.4.0
##  [7] ncdfFlow_2.38.0          BH_1.75.0-0              RcppArmadillo_0.10.5.0.0
## [10] flowCore_2.4.0           here_1.0.1               car_3.0-11
## [13] carData_3.0-4            ggplot2_3.3.5            fitdistrplus_1.1-5
## [16] survival_3.2-11          MASS_7.3-54              vegan_2.5-7
## [19] lattice_0.20-44          permute_0.9-5            BiocManager_1.30.16
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-152        matrixStats_0.59.0  RColorBrewer_1.1-2
##  [4] httr_1.4.2          rprojroot_2.0.2     Rgraphviz_2.36.0
##  [7] backports_1.2.1     tools_4.1.0         utf8_1.2.1
## [10] R6_2.5.0            BiocGenerics_0.38.0 mgcv_1.8-36
## [13] colorspace_2.0-2    withr_2.4.2         tidyselect_1.1.1
## [16] gridExtra_2.3       curl_4.3.2          compiler_4.1.0
## [19] graph_1.70.0        Biobase_2.52.0      xml2_1.3.2
## [22] hexbin_1.28.2       stringr_1.4.0       digest_0.6.27
## [25] foreign_0.8-81      rmarkdown_2.9       rio_0.5.27
## [28] base64enc_0.1-3     jpeg_0.1-8.1        pkgconfig_2.0.3
## [31] htmltools_0.5.1.1   rlang_0.4.11        readxl_1.3.1
## [34] generics_0.1.0      dplyr_1.0.7         zip_2.2.0
## [37] magrittr_2.0.1      RProtoBufLib_2.4.0  Matrix_1.3-4
## [40] Rcpp_1.0.6          munsell_0.5.0       S4Vectors_0.30.0
## [43] fansi_0.5.0         abind_1.4-5         lifecycle_1.0.0
## [46] stringi_1.6.2       yaml_2.2.1          zlibbioc_1.38.0
## [49] expint_0.1-6        plyr_1.8.6          grid_4.1.0
## [52] parallel_4.1.0      forcats_0.5.1       crayon_1.4.1
## [55] haven_2.4.1         splines_4.1.0       hms_1.1.0
## [58] knitr_1.33          pillar_1.6.1        ggsignif_0.6.2
## [61] stats4_4.1.0        XML_3.99-0.6        glue_1.4.2
## [64] evaluate_0.14       latticeExtra_0.6-29 data.table_1.14.0
## [67] RcppParallel_5.1.4  vctrs_0.3.8         png_0.1-7
## [70] cellranger_1.1.0    tidyr_1.1.3         gtable_0.3.0
## [73] aws.s3_0.3.21       purrr_0.3.4         xfun_0.23
## [76] openxlsx_4.2.4      broom_0.7.8         rstatix_0.7.0
## [79] cytolib_2.4.0       aws.signature_0.6.0 cluster_2.1.2
## [82] ellipsis_0.3.2
##
## Bioconductor version '3.13'
##
##   * 4 packages out-of-date
##   * 0 packages too new
##
## create a valid installation with
##
```

```
##   BiocManager::install(c(
##     "isoband", "Rcpp", "stringi", "xfun"
##   ), update = TRUE, ask = FALSE)
##
## more details: BiocManager::valid()$too_new, BiocManager::valid()$out_of_date
```

```
BiocManager::install("flowCore")
```

```
## Bioconductor version 3.13 (BiocManager 1.30.16), R 4.1.0 (2021-05-18)
```

```
## Warning: package(s) not installed when version(s) same as current; use `force = TRUE` to
##   re-install: 'flowCore'
```

```
## Installation paths not writeable, unable to update packages
##   path: C:/Program Files/R/R-4.1.0/library
##   packages:
##     Matrix, mgcv
```

```
## Old packages: 'isoband', 'Rcpp', 'stringi', 'xfun'
```

```
BiocManager::install("ggcyto")
```

```
## Bioconductor version 3.13 (BiocManager 1.30.16), R 4.1.0 (2021-05-18)
```

```
## Warning: package(s) not installed when version(s) same as current; use `force = TRUE` to
##   re-install: 'ggcyto'
```

```
## Installation paths not writeable, unable to update packages
##   path: C:/Program Files/R/R-4.1.0/library
##   packages:
##     Matrix, mgcv
## Old packages: 'isoband', 'Rcpp', 'stringi', 'xfun'
```

```
BiocManager::install()
```

```
## Bioconductor version 3.13 (BiocManager 1.30.16), R 4.1.0 (2021-05-18)
```

```
## Installation paths not writeable, unable to update packages
##   path: C:/Program Files/R/R-4.1.0/library
##   packages:
##     Matrix, mgcv
```

```
## Old packages: 'isoband', 'Rcpp', 'stringi', 'xfun'
```

```
my.cols <- RColorBrewer::brewer.pal(n = 4, name = "Greys")[3:4]

# Set theme for figures in the paper
theme_set(theme_classic() +
  theme(axis.title = element_text(size = 16),
        axis.title.x = element_text(margin = margin(t = 15, b = 15)),
        axis.title.y = element_text(margin = margin(l = 15, r = 15)),
        axis.text = element_text(size = 14),
        axis.text.x = element_text(margin = margin(t = 5)),
        axis.text.y = element_text(margin = margin(r = 5)),
        #axis.line.x = element_line(size = 1),
        #axis.line.y = element_line(size = 1),
        axis.line.x = element_blank(),
        axis.line.y = element_blank(),
        axis.ticks.x = element_line(size = 1),
        axis.ticks.y = element_line(size = 1),
```

```
        axis.ticks.length = unit(.1, "in"),
        panel.border = element_rect(color = "black", fill = NA, size = 1.5),
        legend.title = element_blank(),
        legend.text = element_text(size = 14),
        strip.text = element_text(size = 14),
        strip.background = element_blank()
        ))
```

#RAC function takes in a list of activity units and returns the list ranked

```
RAC <- function(x = ""){
  x = as.vector(x)
  x.ab = x[x >0]
  x.ab.ranked = x.ab[order(x.ab, decreasing = TRUE)]
  return(x.ab.ranked)
}
```

—

```
mempotratio <-function(x = "", green_channel = "", red_channel = ""){
  fcs <- flowCore::read.FCS(file = x)
  flow <- flowCore::exprs(fcs)
  ratio <- as.data.frame(flow[,red_channel] - flow[,green_channel] + (1.5*2330169))
  rac <- RAC(x = ratio)
  ranks <- as.vector(seq(1,length(rac)))
  rac <- cbind(ranks, rac)
  return(rac)
  #scale ranked ratios?
}
```

#process function takes in the name of a fcs file, channel to be saved, and scale and returns a ranked rank activity matrix

```
processmultiple <- function(x = "", channel = "", scale = "", name = "", desc = ""){
  n <- 1
  mynames <- list()
  #make a list with names of cat(name,desc, sep = "_")
  raclist <- list()
  for(file in x){
    mynames <- cbind(mynames, cat(name,desc,sep = "_"))
  }
  for (file in x){
    if(scale == TRUE){
      fcs <- flowCore::read.FCS(file = file, transformation = "scale")
    }
    else{
      fcs <- flowCore::read.FCS(file = file)
    }
    flow <- flowCore::exprs(fcs)
    RSG_H <- as.data.frame(flow[,channel])
    rac <- RAC(x = RSG_H)
    ranks <- as.vector(seq(1,length(rac)))
    rac <- cbind(ranks, rac)
    raclist[[n]] <- as.data.frame(rac)
    n <- n + 1
  }
```

```
  names(raclist) <- mynames
  return(raclist)
}
```

```
process <- function(x = "", channel = "", scale = ""){
  if(scale == TRUE){
    fcs <- flowCore::read.FCS(file = x, transformation = "scale")
  }
  else{
    fcs <- flowCore::read.FCS(file = x)
  }
  flow <- flowCore::exprs(fcs)
  RSG_H <- as.data.frame(flow[,channel])
  rac <- RAC(x = RSG_H)
  ranks <- as.vector(seq(1,length(rac)))
  rac <- cbind(ranks, rac)
  return(as.data.frame(rac))
}
```

#Cdist function takes in a list of activity values and calculates the CDF and returns the CDF and percentage of cells contributing to CDF

```
CDist <- function(x = ""){
  x <- as.vector(x)
  sum <- sum(x)
  rank <- 1
  total <- length(x)
  cdist <- as.vector((x[1]/sum)*100)
  Per <- as.vector((rank/total) * 100)
  for(num in x){
    rank <- rank + 1
    x <- x[-1]
    current <- cdist[length(cdist)] + ((x[1]/sum)*100)
    Per <- c(Per, ((rank/total) *100))
    cdist <- c(cdist, current)
  }
  ranked <- cbind(cdist, Per)
  return(as.data.frame(ranked))

}
```

```
PBS_04 <- flowCore::read.FCS(file = "./data/20210704_MetDist/A1_PBS.fcs")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
Ecoli_04 <- flowCore::read.FCS(file = "./data/20210704_MetDist/B1_Ecoli.fcs")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
QNA_04 <- flowCore::read.FCS(file = "./data/20210704_MetDist/C1_QNA.fcs")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
QUE7_04 <- flowCore::read.FCS(file = "./data/20210704_MetDist/D1_QUE7.fcs")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```r
QUE2_04 <- flowCore::read.FCS(file = "./data/20210704_MetDist/A2_QUE2.fcs")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```r
REXYFP_04 <- flowCore::read.FCS(file = "./data/20210704_MetDist/B2_REXYFP.fcs")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```r
Peredox_04 <- flowCore::read.FCS(file = "./data/20210704_MetDist/C2_Peredox.fcs")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```r
summary(PBS_04)
```

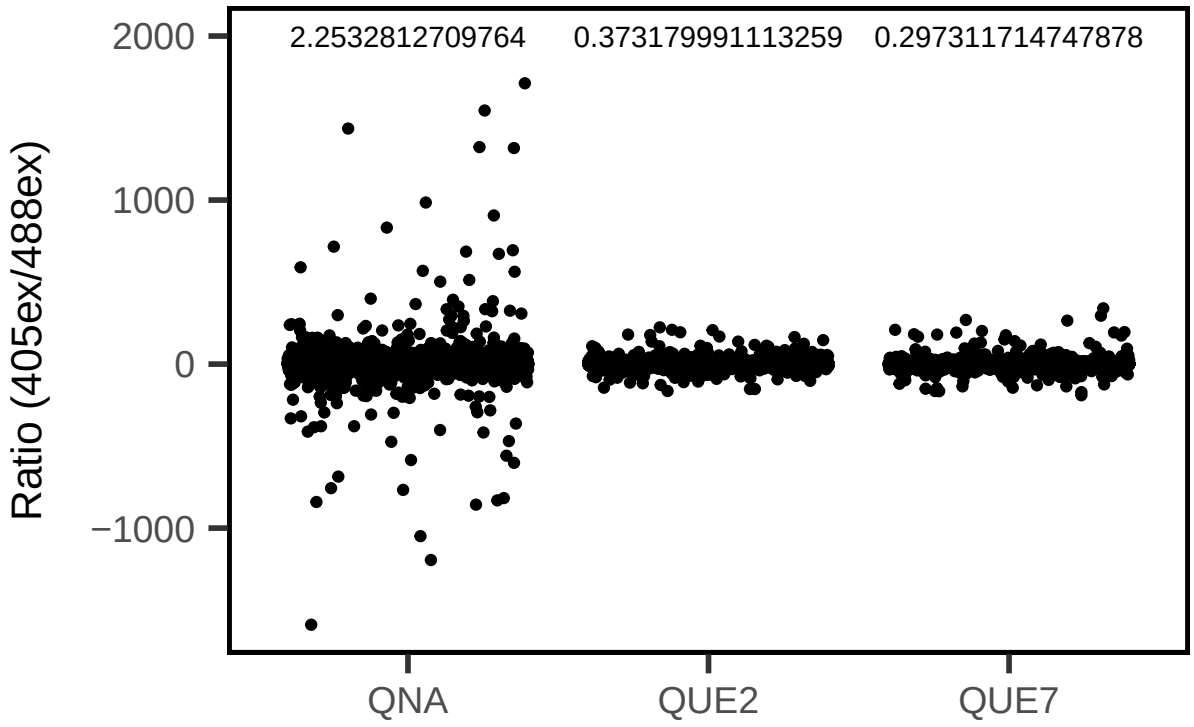
```
##              FSC-H     SSC-H      B530-H     V530-H     Y615-H      FSC-A
## Min.      251.0000     0.000  -402.00000  -322.0000  -524.0000    28.0000
```

```
## 1st Qu.    263.0000     91.000   -88.00000   -77.0000      7.0000     37.0000
## Median     280.0000    169.000   -24.00000   -15.0000    156.0000     43.0000
## Mean       804.8066   3858.366   -14.21417     1.5735    180.5159    338.0868
## 3rd Qu.    321.0000    325.000    44.00000    51.0000    316.0000     61.0000
## Max.    100000.0000 129105.000 34512.00000 80766.0000 100000.0000 100000.0000
##              SSC-A      B530-A      V530-A       Y615-A   Width     Time
## Min.     -1018.000 -3283.00000  -5734.0000 -12809.00000  12.000     3530
## 1st Qu.    -18.000   -92.00000   -253.0000   -141.00000  13.000  6364157
## Median      -1.000     2.00000   -168.0000     19.00000  15.000 12561323
## Mean      1357.527    14.80758   -128.1391     68.92258  20.507 12799537
## 3rd Qu.     34.000    93.00000    -80.0000    194.00000  21.000 19369636
## Max.    100000.000 45120.00000 100000.0000 100000.00000 500.000 26288726
```

```
QUE2_Ratio <- as.data.frame(flowCore::exprs(QUE2_04)[,c("B530-H", "V530-H")])
QUE2_Ratio$Ratio <- QUE2_Ratio$`V530-H`/QUE2_Ratio$`B530-H`
QUE2_Ratio <- QUE2_Ratio[is.finite(QUE2_Ratio$Ratio) == TRUE,]
QUE7_Ratio <- as.data.frame(flowCore::exprs(QUE7_04)[,c("B530-H", "V530-H")])
QUE7_Ratio$Ratio <- QUE7_Ratio$`V530-H`/QUE7_Ratio$`B530-H`
QUE7_Ratio <- QUE7_Ratio[is.finite(QUE7_Ratio$Ratio) == TRUE,]
QNA_Ratio <- as.data.frame(flowCore::exprs(QNA_04)[,c("B530-H", "V530-H")])
QNA_Ratio$Ratio <- QNA_Ratio$`V530-H`/QNA_Ratio$`B530-H`
QNA_Ratio <- QNA_Ratio[is.finite(QNA_Ratio$Ratio) == TRUE,]
QUE7 <- data.frame(group = "QUE7", value = QUE7_Ratio$Ratio)
QUE2 <- data.frame(group = "QUE2", value = QUE2_Ratio$Ratio)
QNA <- data.frame(group = "QNA", value = QNA_Ratio$Ratio)
ratios <- rbind(QUE7, QUE2, QNA)

QUEEN <- ggplot(data = ratios, aes(x = group, y = value))+
    geom_jitter()+
    ylab("Ratio (405ex/488ex)")+
    xlab("")+
    annotate("text", x = 1, y = 2000, label = mean(QNA_Ratio$Ratio))+
    annotate("text", x = 2, y = 2000, label = mean(QUE2_Ratio$Ratio))+
    annotate("text", x = 3, y = 2000, label = mean(QUE7_Ratio$Ratio))

QUEEN
```




```
ggsave(here("output", "QUEEN.png"))
```

```
## Saving 6.5 x 4.5 in image
```

```
ggsave(here("output", "QUEEN.png"))
```

```
## Saving 6.5 x 4.5 in image
```

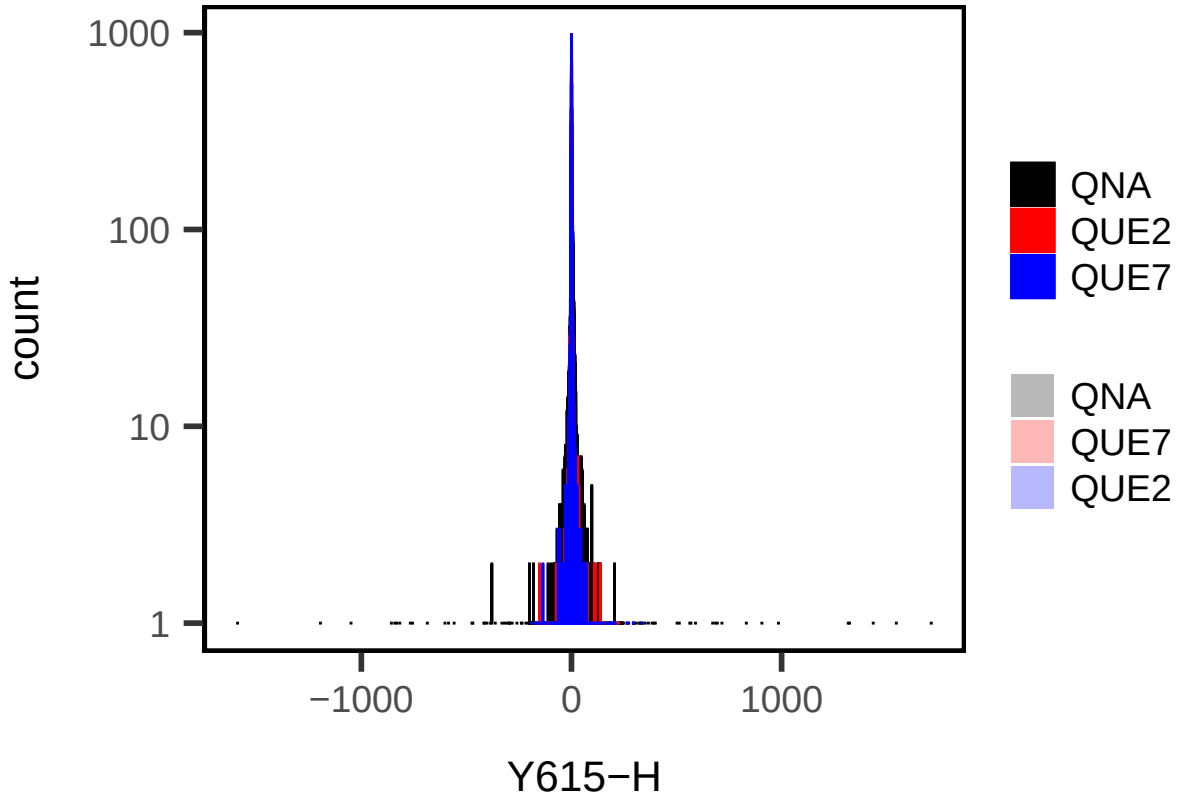
```
QUEEN_hist <- ggplot(QNA_Ratio, aes(x = Ratio, color = "QNA", fill = "QNA"))+
  geom_histogram(binwidth = 0.25)+
  geom_histogram(data = QUE2_Ratio, aes(x = Ratio, color = "QUE2", fill = "QUE2"),
                 binwidth = 0.25)+
  geom_histogram(data = QUE7_Ratio, aes(x = Ratio, color = "QUE7", fill = "QUE7"),
                 binwidth = 0.25)+
  scale_color_manual("Legend Title", limits=c("QNA", "QUE2", "QUE7"),
                     values = c("black","red", "blue"))+
  scale_fill_manual("Legend Title", limits=c("QNA", "QUE7", "QUE2"),
                    values = alpha(c("black", "red", "blue"), 0.1))+
  guides(colour = guide_legend(override.aes = list(pch = c(16, 16, 16),
                    fill = c("black", "red", "blue"))))+
  xlab("Y615-H")+
  scale_y_log10()

QUEEN_hist
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
## Warning: Removed 12484 rows containing missing values (geom_bar).
## Warning: Removed 12829 rows containing missing values (geom_bar).
## Warning: Removed 12835 rows containing missing values (geom_bar).
```




```
ggsave(here("output", "QUEEN_hist.png"))
```

```
## Saving 6.5 x 4.5 in image
## Warning: Transformation introduced infinite values in continuous y-axis
## Warning: Transformation introduced infinite values in continuous y-axis
## Warning: Transformation introduced infinite values in continuous y-axis
## Warning: Removed 12484 rows containing missing values (geom_bar).
## Warning: Removed 12829 rows containing missing values (geom_bar).
## Warning: Removed 12835 rows containing missing values (geom_bar).
```

```
ggsave(here("output", "QUEEN_hist.pdf"))
```

```
## Saving 6.5 x 4.5 in image
## Warning: Transformation introduced infinite values in continuous y-axis
## Warning: Transformation introduced infinite values in continuous y-axis
```

```
## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Removed 12484 rows containing missing values (geom_bar).

## Warning: Removed 12829 rows containing missing values (geom_bar).

## Warning: Removed 12835 rows containing missing values (geom_bar).
```

```
PBS_04_rac_y615 <- process("./data/20210704_MetDist/A1_PBS.fcs", "Y615-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
Ecoli_04_rac_y615 <- process("./data/20210704_MetDist/B1_Ecoli.fcs", "Y615-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
QNA_04_rac_y615 <- process("./data/20210704_MetDist/C1_QNA.fcs", "Y615-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
QUE7_04_rac_y615 <- process("./data/20210704_MetDist/D1_QUE7.fcs", "Y615-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
QUE2_04_rac_y615 <- process("./data/20210704_MetDist/A2_QUE2.fcs", "Y615-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
REXYFP_04_rac_y615 <- process("./data/20210704_MetDist/B2_REXYFP.fcs", "Y615-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

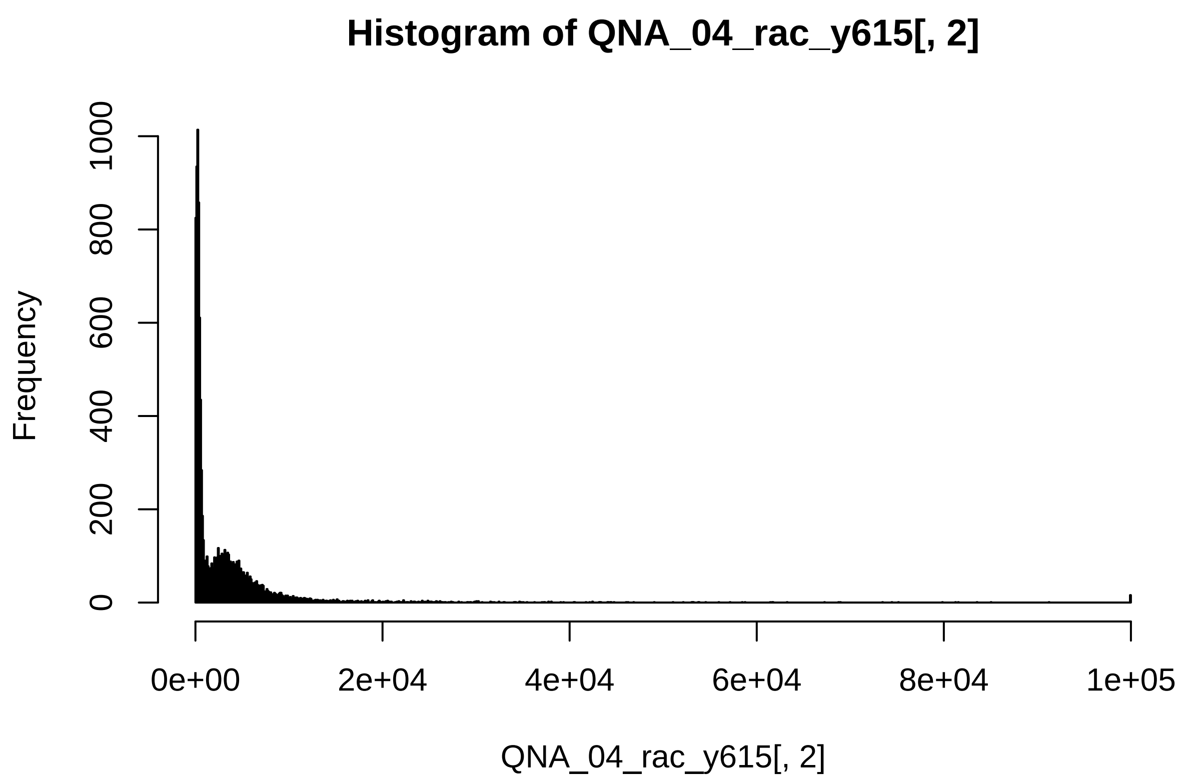
```
Peredox_04_rac_y615 <- process("./data/20210704_MetDist/C2_Peredox.fcs", "Y615-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

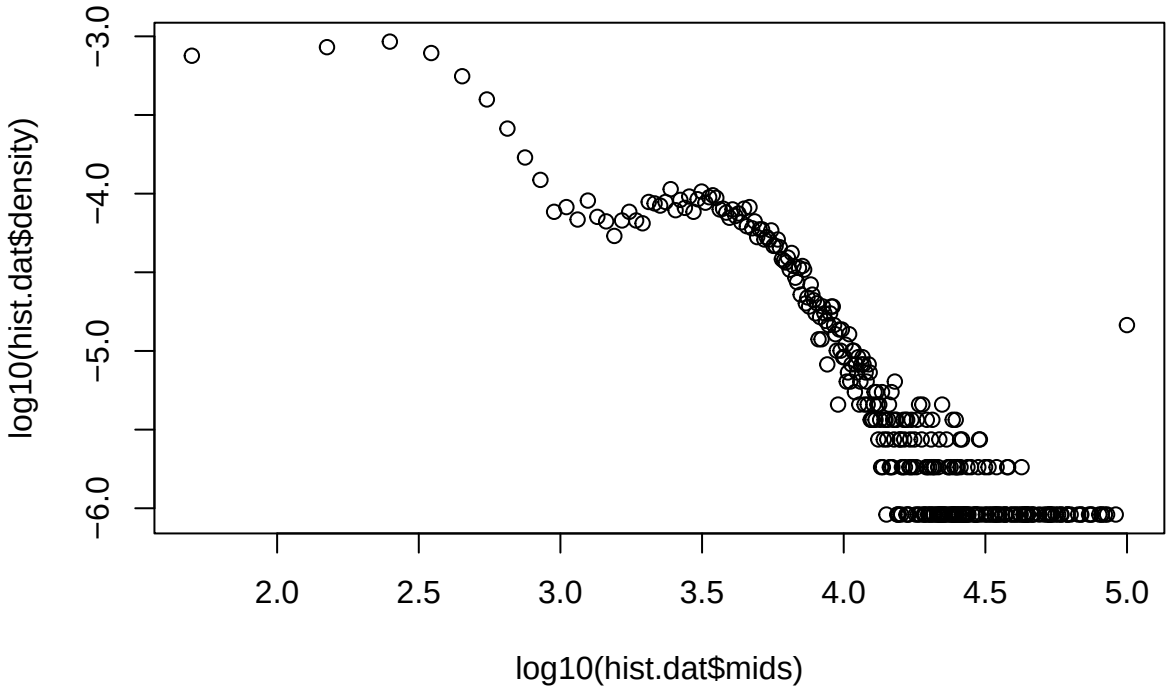
```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```r
PBS_04_cdist_y615 <- CDist(PBS_04_rac_y615[,2])
Ecoli_04_cdist_y615 <- CDist(Ecoli_04_rac_y615[,2])
QNA_04_cdist_y615 <- CDist(QNA_04_rac_y615[,2])
QUE7_04_cdist_y615 <- CDist(QUE7_04_rac_y615[,2])
QUE2_04_cdist_y615 <- CDist(QUE2_04_rac_y615[,2])
REXYFP_04_cdist_y615 <- CDist(REXYFP_04_rac_y615[,2])
Peredox_04_cdist_y615 <- CDist(Peredox_04_rac_y615[,2])

hist <- hist(QNA_04_rac_y615[,2], breaks = 1000)
```

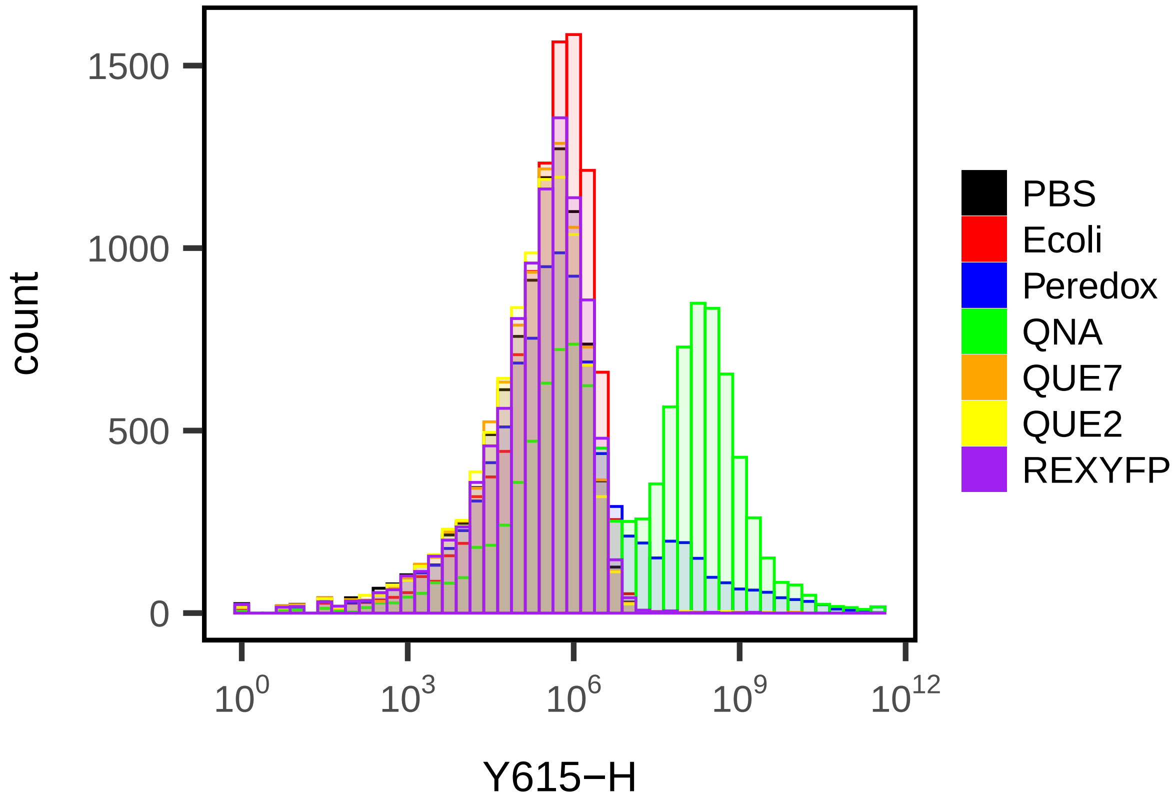


```
hist.dat <- data.frame(hist$mids,hist$counts,hist$density)
colnames(hist.dat)<- c("mids", "counts", "density")
plot(log10(hist.dat$mids), log10(hist.dat$density))
```




```
Peredox <- ggplot(PBS_04_rac_y615, aes(x = log(rac), color = "PBS", fill = "PBS"))+
  geom_histogram(binwidth = 0.25)+
  geom_histogram(data = Ecoli_04_rac_y615, aes(x = log(rac), color = "Ecoli", fill = "Ecoli"),
                 binwidth = 0.25)+
  geom_histogram(data = Peredox_04_rac_y615, aes(x = log(rac), color = "Peredox", fill = "Peredox"),
                 binwidth = 0.25)+
  geom_histogram(data =   QNA_04_rac_y615, aes(x = log(rac), color = "QNA", fill = "QNA"),
                 binwidth = 0.25)+
  geom_histogram(data =   QUE7_04_rac_y615, aes(x = log(rac), color = "QUE7", fill = "QUE7"),
                 binwidth = 0.25)+
  geom_histogram(data =   QUE2_04_rac_y615, aes(x = log(rac), color = "QUE2", fill = "QUE2"),
                 binwidth = 0.25)+
  geom_histogram(data =   REXYFP_04_rac_y615, aes(x = log(rac), color = "REXYFP", fill = "REXYFP"),
                 binwidth = 0.25)+
  scale_color_manual("Legend Title", limits=c("PBS", "Ecoli", "Peredox", "QNA", "QUE7", "QUE2", "REXYFP
                     values = c("black","red", "blue", "green", "orange", "yellow", "purple"))+
  scale_fill_manual("Legend Title", limits=c("PBS", "Ecoli", "Peredox", "QNA", "QUE7", "QUE2", "REXYFP"
                    values = alpha(c("black", "red", "blue", "green", "orange", "yellow", "purple"), 0.
  guides(colour = guide_legend(override.aes = list(pch = c(16, 16, 16, 16, 16, 16, 16),
                    fill = c("black", "red", "blue", "green", "orange", "yellow", "purple"))))+
  xlab("Y615-H")+
  scale_x_continuous(labels = label_math(expr = 10^.x, format = force))
```

```
Peredox
```




```
ggsave(here("output", "Peredox_04.png"))
```

```
## Saving 6.5 x 4.5 in image
```

```
ggsave(here("output", "Peredox_04.png"))
```

```
## Saving 6.5 x 4.5 in image
```

#Comparison of CCCP and non-CCCP E.coli samples

```
aa_GC <- processmultiple(c("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T1_S.fcs","./data/FCS/EAM_2019
```

```
## aa_rac_Ecoli_GC_1_2aa_rac_Ecoli_GC_1_2

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
names(aa_GC)
```

```
## [1] NA NA
```

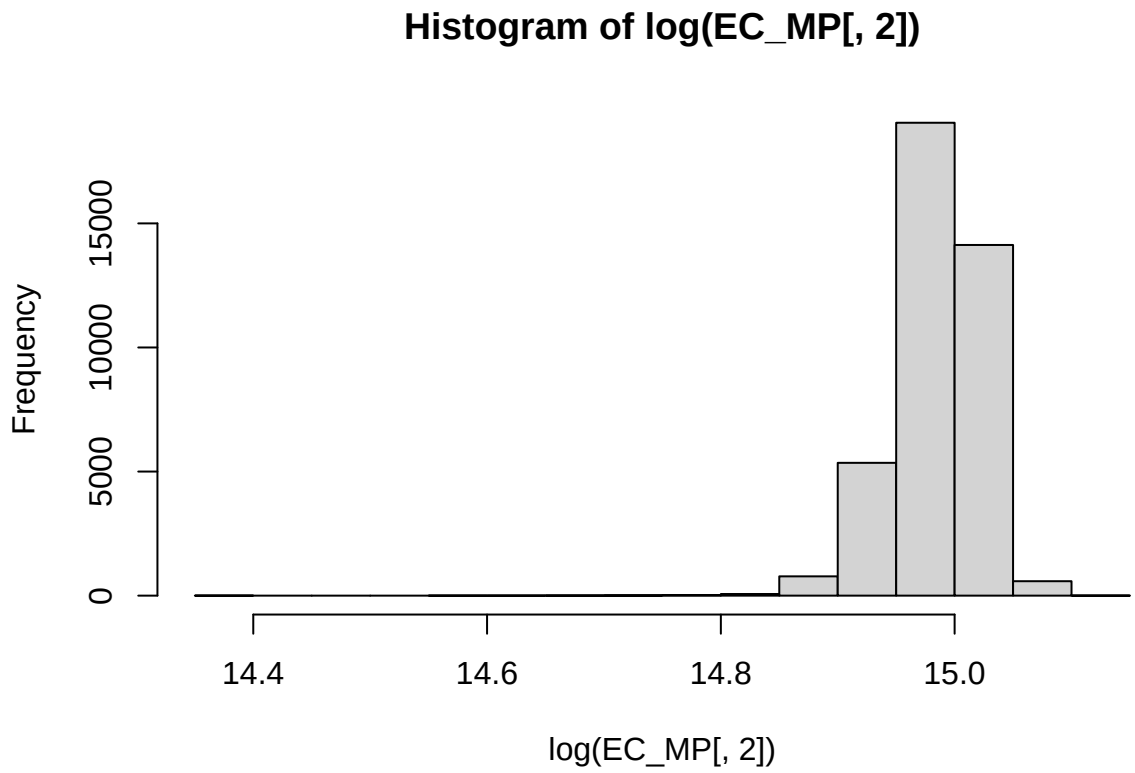
```
EC_MP <- mempotratio("./data/FCS/EAM_20190607_MemPot/Specimen1_EC_MP_T5.fcs", "BL1-H", "BL4-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

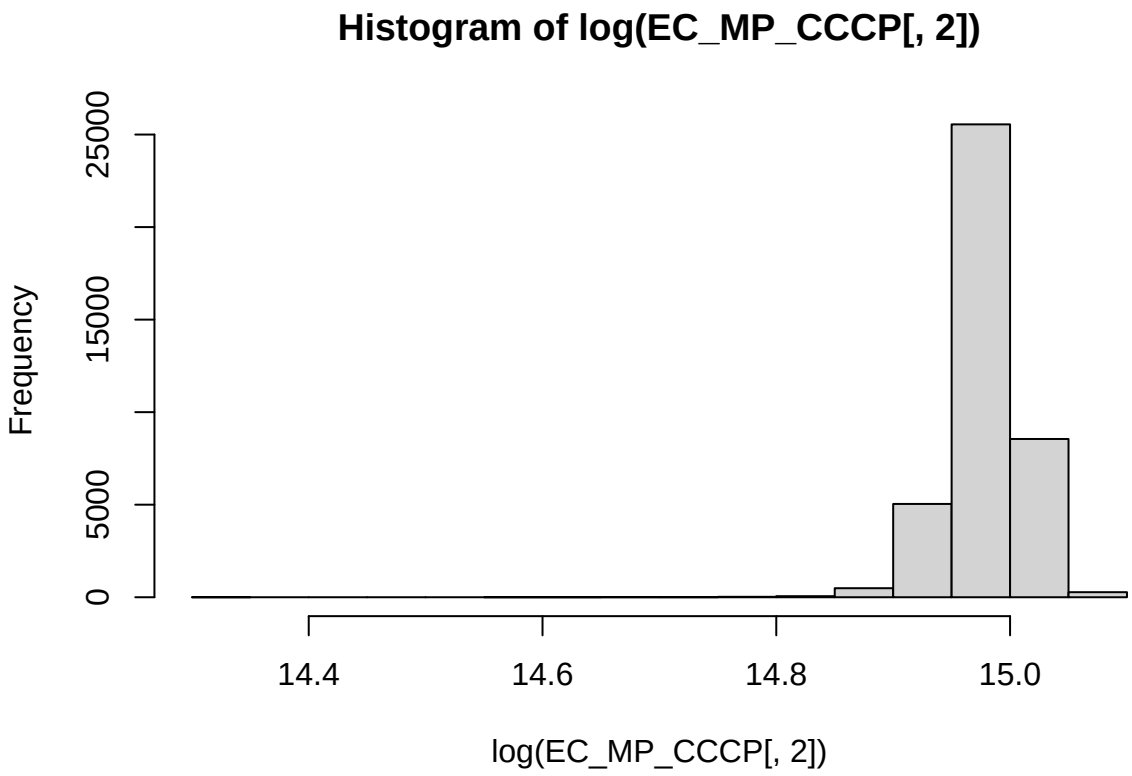
```
EC_MP_CCCP <- mempotratio("./data/FCS/EAM_20190607_MemPot/Specimen1_EC_MP_CCCP_T5.fcs", "BL1-H", "BL4-H
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

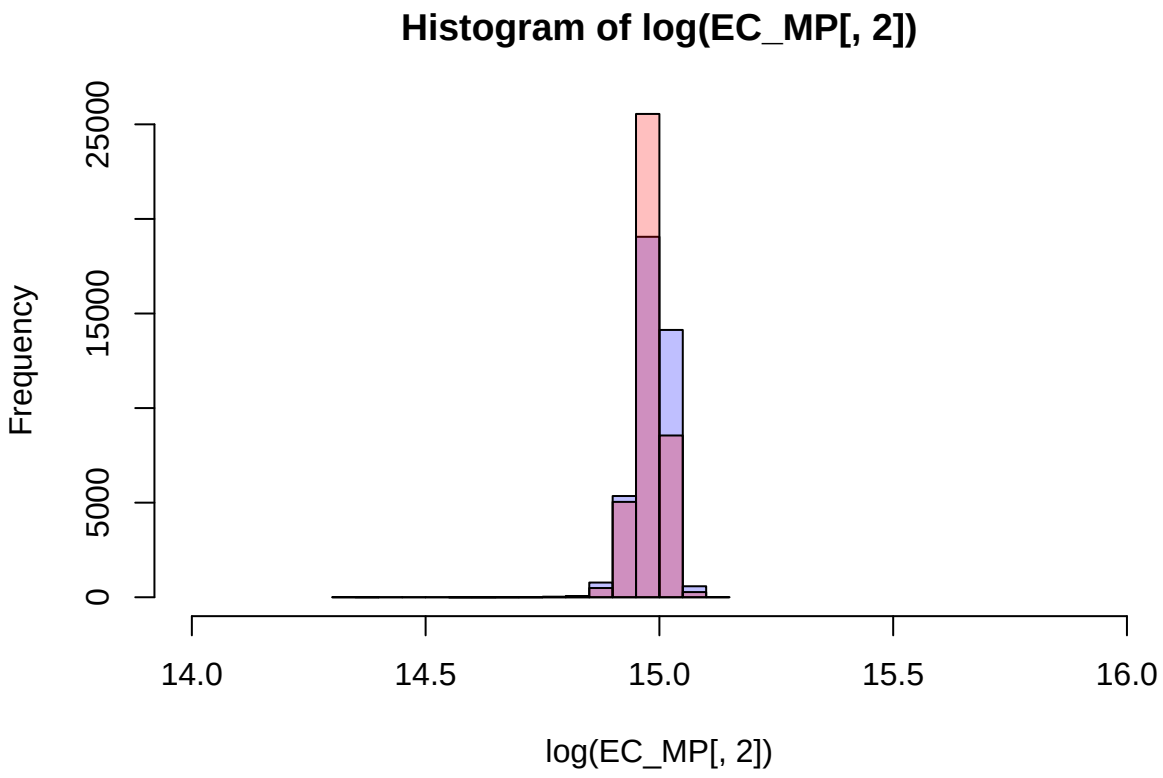
```
p1 <- hist(log(EC_MP[,2]))
```




```
p2 <- hist(log(EC_MP_CCCP[,2]))
```




```
plot(p1, col=rgb(0,0,1,1/4), xlim=c(14,16), ylim = c(0, 25000))
plot(p2, col=rgb(1,0,0,1/4), xlim= c(14,16), ylim = c(0,25000), add = T)
```




```
MempotNOCCCP <- mempotratio("./data/FCS/EAM_20190605_MemPot_Test/Specimen1_ECOLI-CCCP_E1.fcs", "BL4-H",
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
MempotCCCP <- mempotratio("./data/FCS/EAM_20190605_MemPot_Test/Specimen1_ECOLI+CCCP_E1.fcs", "BL4-H", "
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

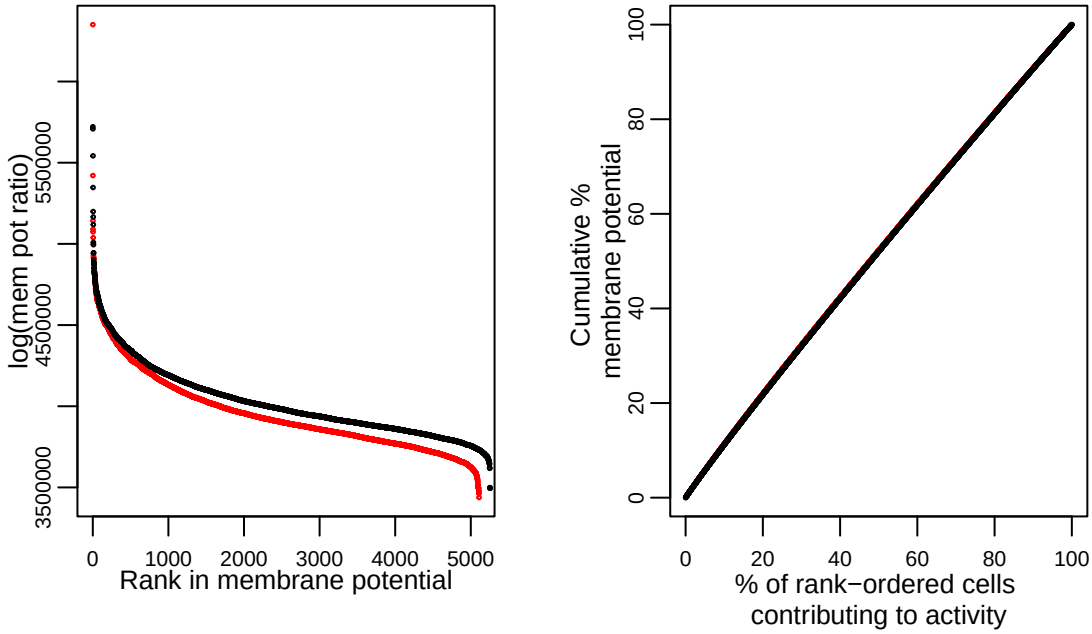
```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
par(fig = c(0,0.525,0,1))
plot(MempotNOCCCP[,1], MempotNOCCCP[,2], type = 'p', xlab = "", ylab = "", cex = 0.25, col = "red", yax
points(MempotCCCP[,1], MempotCCCP[,2], type = 'p', cex = 0.25)
mtext("Rank in membrane potential", side = 1, line = 1.1, cex = 0.8)
mtext("log(mem pot ratio)", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7, mgp = c(3, 0.5, 0))

CdistMempotNOCCCP <- CDist(MempotNOCCCP[,2])
CdistMempotCCCP <- CDist(MempotCCCP[,2])

par(fig = c(0.475, 1,0,1), new = TRUE)
plot(CdistMempotNOCCCP[,2], CdistMempotNOCCCP[,1], type = 'p', xlab = "", ylab = "", cex = 0.25, col =
points(CdistMempotCCCP[,2], CdistMempotCCCP[,1], type = 'p', cex = 0.25)
mtext("% of rank-ordered cells \ncontributing to activity", side = 1, line = 2, cex = 0.8)
```

```
mtext("Cumulative % \nmembrane potential", side = 2, line = 1.1, cex = 0.8)
axis(2, at = c(0,20,40,60,80,100), labels = c("0","20","40","60","80","100"), cex.axis=0.7, mgp = c(3,0
axis(1, cex.axis = 0.7, mgp = c(3,0.5,0), at = c(0,20,40,60,80,100), labels = TRUE)
```



#GROWTH CURVE ANALYSIS

##Process growth rate fcs files

```
aa_rac_Ecoli_GC_1 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T1_S.fcs", "BL1-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
aa_rac_Ecoli_GC_2 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T2_S.fcs", "BL1-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
aa_rac_Ecoli_GC_3 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T3_S.fcs", "BL1-H")
aa_rac_Ecoli_GC_4 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T4_S.fcs", "BL1-H")
aa_rac_Ecoli_GC_5 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T5_S.fcs", "BL1-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
```

```
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
aa_rac_Ecoli_GC_6 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T6_S.fcs", "BL1-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
aa_rac_Ecoli_GC_7 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T7_S.fcs", "BL1-H")
aa_Cdist_Ecoli_GC_1 <- CDist(aa_rac_Ecoli_GC_1[,2])
aa_Cdist_Ecoli_GC_2 <- CDist(aa_rac_Ecoli_GC_2[,2])
aa_Cdist_Ecoli_GC_3 <- CDist(aa_rac_Ecoli_GC_3[,2])
aa_Cdist_Ecoli_GC_4 <- CDist(aa_rac_Ecoli_GC_4[,2])
aa_Cdist_Ecoli_GC_5 <- CDist(aa_rac_Ecoli_GC_5[,2])
aa_Cdist_Ecoli_GC_6 <- CDist(aa_rac_Ecoli_GC_6[,2])
aa_Cdist_Ecoli_GC_7 <- CDist(aa_rac_Ecoli_GC_7[,2])

ra_rac_Ecoli_GC_1 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T1_S.fcs", "BL1-H", TRUE)
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
ra_rac_Ecoli_GC_2 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T2_S.fcs", "BL1-H", TRUE)
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
ra_rac_Ecoli_GC_3 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T3_S.fcs", "BL1-H", TRUE)
ra_rac_Ecoli_GC_4 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T4_S.fcs", "BL1-H", TRUE)
ra_rac_Ecoli_GC_5 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T5_S.fcs", "BL1-H", TRUE)
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
ra_rac_Ecoli_GC_6 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T6_S.fcs", "BL1-H", TRUE)
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
ra_rac_Ecoli_GC_7 <- process("./data/FCS/EAM_20190531_GrowthCurve/Specimen1_T7_S.fcs", "BL1-H", TRUE)
ra_Cdist_Ecoli_GC_1 <- CDist(ra_rac_Ecoli_GC_1[,2])
ra_Cdist_Ecoli_GC_2 <- CDist(ra_rac_Ecoli_GC_2[,2])
ra_Cdist_Ecoli_GC_3 <- CDist(ra_rac_Ecoli_GC_3[,2])
ra_Cdist_Ecoli_GC_4 <- CDist(ra_rac_Ecoli_GC_4[,2])
ra_Cdist_Ecoli_GC_5 <- CDist(ra_rac_Ecoli_GC_5[,2])
ra_Cdist_Ecoli_GC_6 <- CDist(ra_rac_Ecoli_GC_6[,2])
ra_Cdist_Ecoli_GC_7 <- CDist(ra_rac_Ecoli_GC_7[,2])

OD600_Ecoli_GC <- read.table(file = "./data/FCS/EAM_20190531_GrowthCurve/20190531_EAM_GrowthCurve.txt",
```

#OD600 vs. Abundance curves and fits

```
#fit logistic growth model to growth curve for OD600
coef(lm(logit(OD600_Ecoli_GC$OD600/0.9)~OD600_Ecoli_GC$minutes))
```

```
## Warning in logit(OD600_Ecoli_GC$OD600/0.9): proportions remapped to (0.025,
## 0.975)
```

```
##            (Intercept) OD600_Ecoli_GC$minutes
##            -2.80522455             0.03145876
```

```
growthcurve_OD<-nls(OD600_Ecoli_GC$OD600~phi1/(1+exp(-(phi2+phi3*OD600_Ecoli_GC$minutes))),
 start=list(phi1=0.9,phi2=-2.805,phi3=0.031), data = OD600_Ecoli_GC, trace=TRUE)
```

```
## 0.005858180 (1.28e+00): par = (0.9 -2.805 0.031)
## 0.002652262 (3.35e-01): par = (0.9523784 -2.744861 0.02813712)
## 0.002391533 (2.29e-02): par = (0.9500425 -2.804009 0.02903252)
## 0.002390358 (1.93e-03): par = (0.9511796 -2.802424 0.02897693)
## 0.002390349 (1.88e-04): par = (0.9510892 -2.803024 0.02898619)
## 0.002390349 (1.84e-05): par = (0.951098 -2.802994 0.02898552)
## 0.002390349 (1.82e-06): par = (0.9510971 -2.802999 0.0289856)
```

```
phi1_OD<-coef(growthcurve_OD)[1]
phi2_OD<-coef(growthcurve_OD)[2]
phi3_OD<-coef(growthcurve_OD)[3]
x<-c(min(OD600_Ecoli_GC$minutes):max(OD600_Ecoli_GC$minutes))
y<-phi1_OD/(1+exp(-(phi2_OD+phi3_OD*x)))
predict_OD600<-data.frame(x,y)

#fit logistic growth model to growth curve for abs_count
coef(lm(logit(OD600_Ecoli_GC$abs_count/490000000)~OD600_Ecoli_GC$minutes))
```

```
##            (Intercept) OD600_Ecoli_GC$minutes
##            -3.16248907             0.03057669
```

```
growthcurve_ac<-nls(OD600_Ecoli_GC$abs_count~phi1/(1+exp(-(phi2+phi3*OD600_Ecoli_GC$minutes))),
 start=list(phi1=490000000,phi2=-3.162,phi3=0.031), data = OD600_Ecoli_GC, trace=TRUE)
```

```
## 7.080901e+15 (1.53e+00): par = (4.9e+08 -3.162 0.031)
## 6.039736e+15 (1.14e+00): par = (528102888 -2.519456 0.02062201)
## 2.605337e+15 (5.26e-02): par = (542424390 -2.708979 0.02280859)
## 2.598008e+15 (6.61e-03): par = (544554401 -2.730712 0.02287843)
## 2.597891e+15 (9.01e-04): par = (543932935 -2.735335 0.02294012)
## 2.597889e+15 (1.21e-04): par = (543955889 -2.735779 0.02294204)
## 2.597889e+15 (1.62e-05): par = (543945052 -2.735867 0.02294317)
## 2.597889e+15 (2.14e-06): par = (543945282 -2.735876 0.02294321)
```

```
phi1_ac<-coef(growthcurve_ac)[1]
phi2_ac<-coef(growthcurve_ac)[2]
phi3_ac<-coef(growthcurve_ac)[3]
y<-phi1_ac/(1+exp(-(phi2_ac+phi3_ac*x)))
predict_abscount<-data.frame(x,y)
```

#Growth Curve OD600 vs. Abs count fits

```
#generate png file
png(filename="./output/GrowthCurve.OD600_abscount_fits.png",
     width = 900, height = 900, res = 96*2)

par(fig=c(0,1,0.35,1))
color <- c("red", "orange", "yellow", "green", "cyan", "blue", "purple")
plot(OD600_Ecoli_GC$minutes, OD600_Ecoli_GC$OD600, col = color, pch = 19, cex = 1, ylab = "", xlab= "",
mtext("Time(mins)", side = 1, line = 1.1, cex = 0.8)
```

```
mtext("OD600", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7,  mgp = c(3, 0.5, 0))
lines(predict_OD600)

par(fig=c(0,1,0,0.65), new = TRUE)
color <- c("red", "orange", "yellow", "green", "cyan", "blue", "purple")
plot(OD600_Ecoli_GC$minutes, OD600_Ecoli_GC$abs_count, col = color, pch = 19, cex = 1, ylab = "", xlab=
mtext("Time(mins)", side = 1, line = 1.1, cex = 0.8)
mtext("abs. count", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7,  mgp = c(3, 0.5, 0))
lines(predict_abscount)

dev.off()
```

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

#GC lines relative abundance CDF and RAC using OD600

```
#generate png file
png(filename="./output/GrowthCurve.RSGdistribution.png",
     width = 900, height = 900, res = 96*2)

par(fig=c(0,1,0.4,1))
color <- c("red", "orange", "yellow", "green", "cyan", "blue", "purple")
plot(OD600_Ecoli_GC$minutes, OD600_Ecoli_GC$OD600, col = color, pch = 19, cex = 1, ylab = "", xlab= "",
mtext("Time(mins)", side = 1, line = 1.1, cex = 0.8)
mtext("OD600", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7,  mgp = c(3, 0.5, 0))
lines(predict_OD600)

par(fig=c(0,0.55,0,0.65), new=TRUE)
plot(ra_rac_Ecoli_GC_7[,1], log(ra_rac_Ecoli_GC_7[,2]), type = 'p', xlab = "", ylab = "", cex = 0.25, c
mtext("Rank in activity", side = 1, line = 1.1, cex = 0.8)
mtext("log(RSG-H value)", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7, mgp = c(3, 0.5, 0))
points(ra_rac_Ecoli_GC_2[,1], log(ra_rac_Ecoli_GC_2[,2]), type = 'p', cex = 0.25, col = "orange")
points(ra_rac_Ecoli_GC_3[,1], log(ra_rac_Ecoli_GC_3[,2]), type = 'p', cex = 0.25, col = "yellow")
points(ra_rac_Ecoli_GC_4[,1], log(ra_rac_Ecoli_GC_4[,2]), type = 'p', cex = 0.25, col = "green")
points(ra_rac_Ecoli_GC_5[,1], log(ra_rac_Ecoli_GC_5[,2]), type = 'p', cex = 0.25, col = "cyan")
points(ra_rac_Ecoli_GC_6[,1], log(ra_rac_Ecoli_GC_6[,2]), type = 'p', cex = 0.25, col = "blue")
points(ra_rac_Ecoli_GC_1[,1], log(ra_rac_Ecoli_GC_1[,2]), type = 'p', cex = 0.25, col = "purple")
box()

par(fig=c(0.45,1,0,0.65), new=TRUE)
plot(ra_Cdist_Ecoli_GC_1[,2], ra_Cdist_Ecoli_GC_1[,1], type = 'p', xlab = "", ylab = "", cex = 0.25, co
mtext("% of rank-ordered cells \ncontributing to activity", side = 1, line = 2, cex = 0.8)
mtext("Cumulative % \nRSG activity", side = 2, line = 1.1, cex = 0.8)
axis(2, at = c(0,20,40,60,80,100), labels = c("0","20","40","60","80","100"), cex.axis=0.7, mgp = c(3,0
axis(1, cex.axis = 0.7, mgp = c(3,0.5,0), at = c(0,20,40,60,80,100), labels = TRUE)
points(ra_Cdist_Ecoli_GC_2[,2], ra_Cdist_Ecoli_GC_2[,1], type = 'p', cex = 0.25, col = "orange")
```

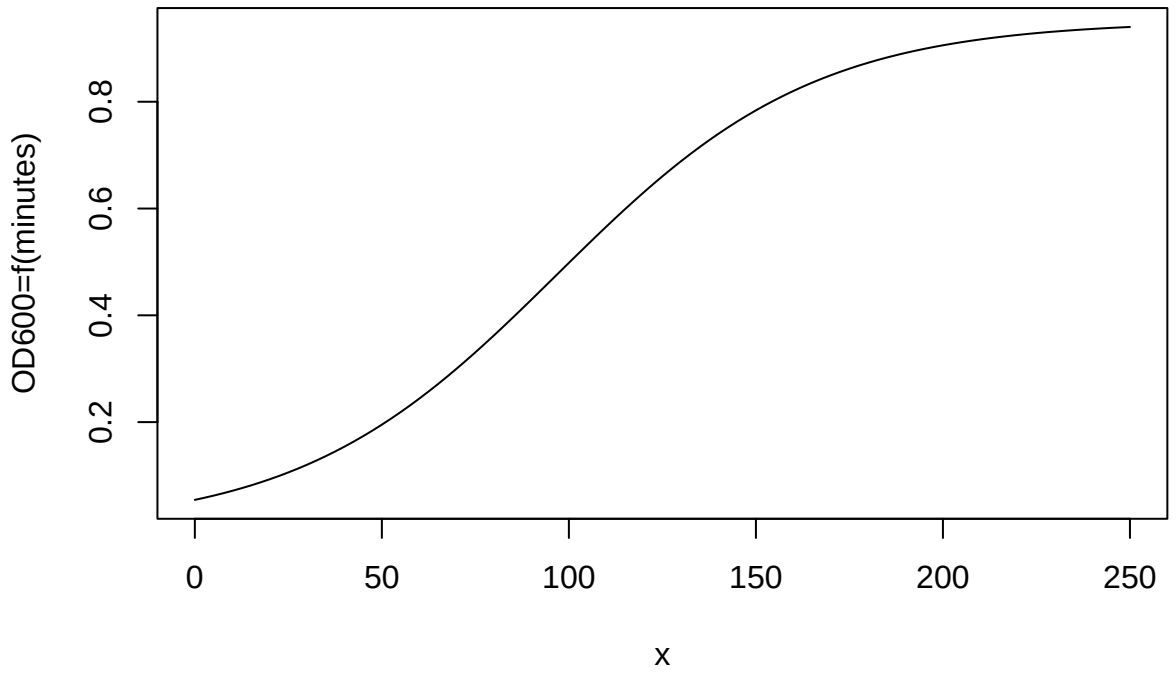
```
points(ra_Cdist_Ecoli_GC_3[,2], ra_Cdist_Ecoli_GC_3[,1], type = 'p', cex = 0.25, col = "yellow")
points(ra_Cdist_Ecoli_GC_4[,2], ra_Cdist_Ecoli_GC_4[,1], type = 'p', cex = 0.25, col = "green")
points(ra_Cdist_Ecoli_GC_5[,2], ra_Cdist_Ecoli_GC_5[,1], type = 'p', cex = 0.25, col = "cyan")
points(ra_Cdist_Ecoli_GC_6[,2], ra_Cdist_Ecoli_GC_6[,1], type = 'p', cex = 0.25, col = "blue")
points(ra_Cdist_Ecoli_GC_7[,2], ra_Cdist_Ecoli_GC_7[,1], type = 'p', cex = 0.25, col = "purple")
box()

dev.off()
```

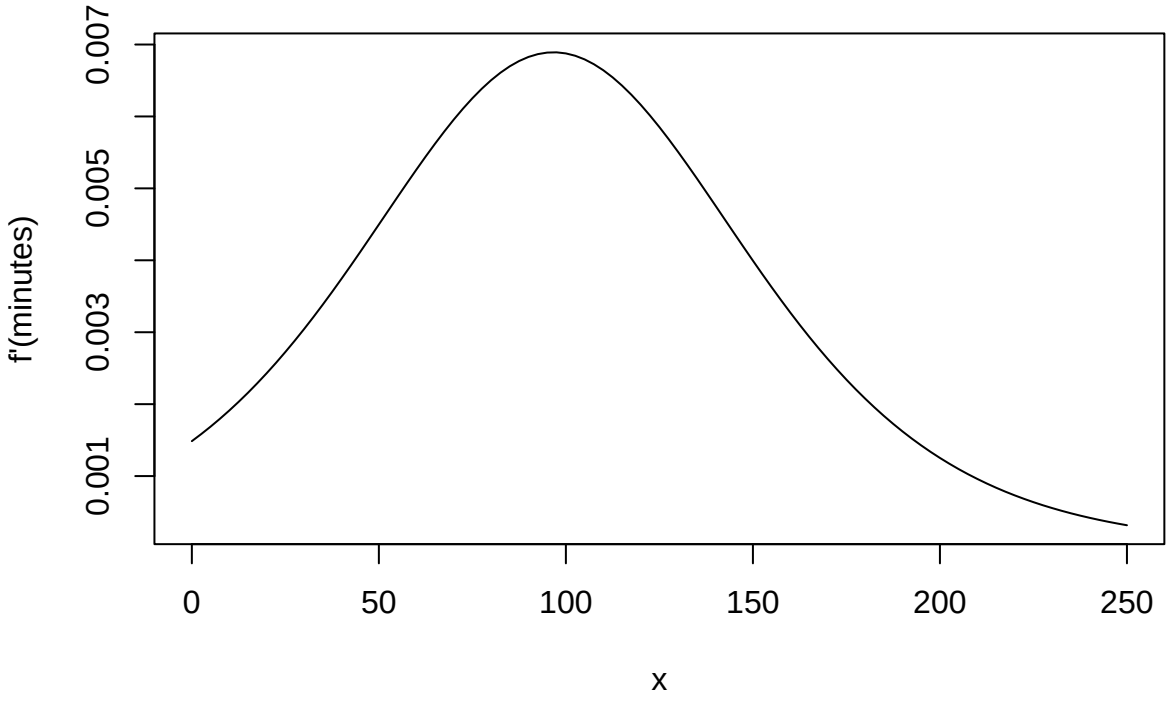
```
## pdf
##   2
```

#Growth curve pareto distribution fits against OD

```
OD600 <- function(x) (phi1_OD/(1+exp(-(phi2_OD+phi3_OD*x))))
curve(OD600, 0, 250, ylab = "OD600=f(minutes)")
```

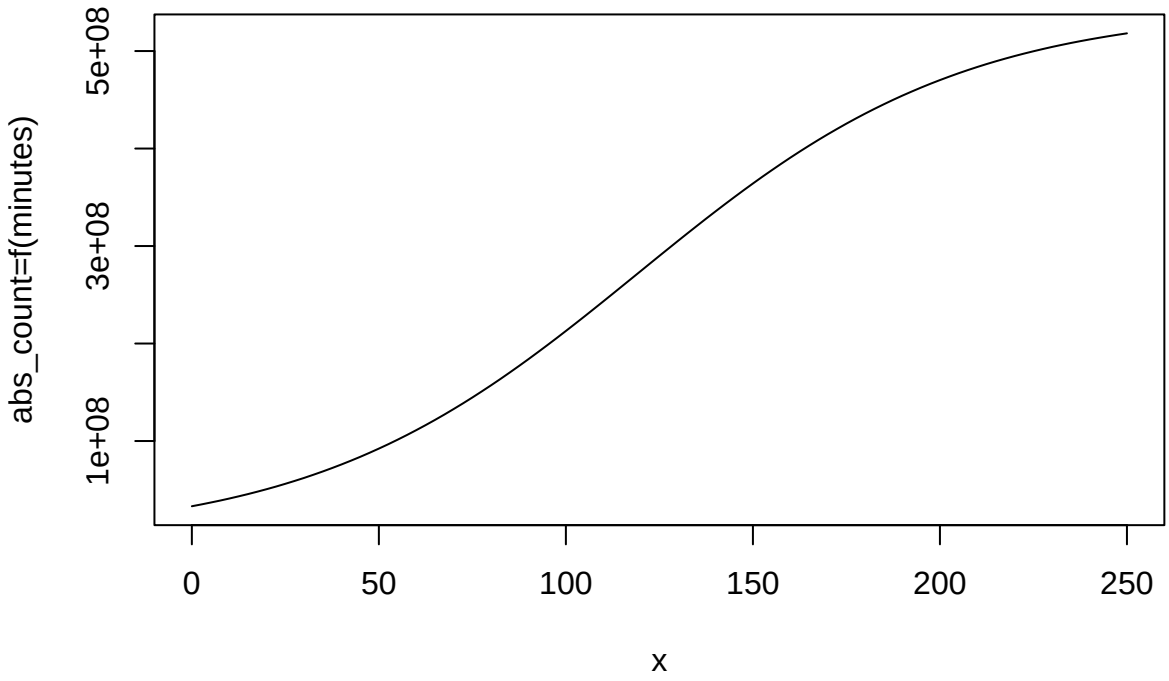



```
deriv_OD600 <- function(x) {}
body(deriv_OD600) <- D(body(OD600), 'x')
curve(deriv_OD600, 0, 250, ylab = "f'(minutes)")
```

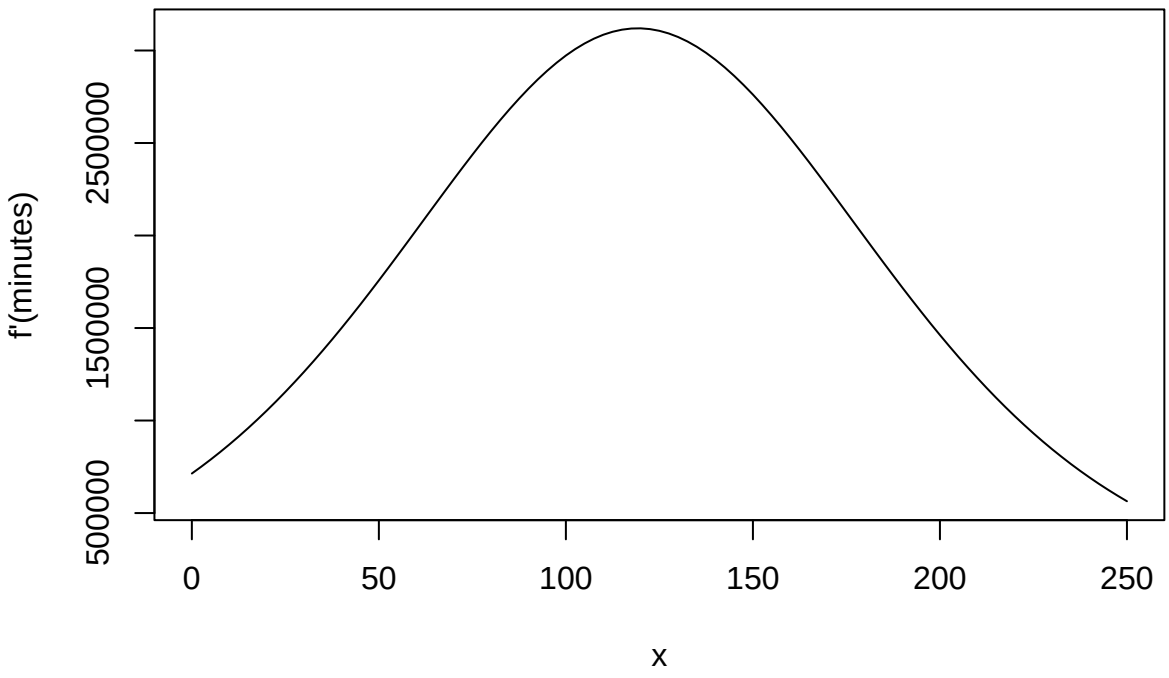



```
deriv <- deriv_OD600(OD600_Ecoli_GC$minutes)

abs_count <- function(x) (phi1_ac/(1+exp(-(phi2_ac+phi3_ac*x))))
curve(abs_count, 0, 250, ylab = "abs_count=f(minutes)")
```




```
deriv_abs_Count <- function(x){}
body(deriv_abs_Count) <- D(body(abs_count), 'x')
curve(deriv_abs_Count, 0, 250, ylab = "f'(minutes)")
```




```
der_abs_count <- deriv_abs_Count(OD600_Ecoli_GC$minutes)

pareto_ra_rac_Ecoli_GC_1 <- fitdist(ra_rac_Ecoli_GC_1[,2], "pareto", start = list(shape = 0.1, scale = 1
```

```
## $start.arg
## $start.arg$shape
## [1] 0.1
##
## $start.arg$scale
## [1] 10
##
##
## $fix.arg
## NULL
```

```
pareto_ra_rac_Ecoli_GC_2 <- fitdist(ra_rac_Ecoli_GC_2[,2], "pareto", start = list(shape = 0.1, scale = 1
```

```
## $start.arg
## $start.arg$shape
## [1] 0.1
##
## $start.arg$scale
## [1] 10
##
##
## $fix.arg
## NULL
```

```
pareto_ra_rac_Ecoli_GC_3 <- fitdist(ra_rac_Ecoli_GC_3[,2], "pareto", start = list(shape = 0.1, scale = 
```

```
## $start.arg
## $start.arg$shape
## [1] 0.1
##
## $start.arg$scale
## [1] 10
##
##
## $fix.arg
## NULL
```

```
pareto_ra_rac_Ecoli_GC_4 <- fitdist(ra_rac_Ecoli_GC_4[,2], "pareto", start = list(shape = 0.1, scale = 
```

```
## $start.arg
## $start.arg$shape
## [1] 0.1
##
## $start.arg$scale
## [1] 10
##
##
## $fix.arg
## NULL
```

```
pareto_ra_rac_Ecoli_GC_5 <- fitdist(ra_rac_Ecoli_GC_5[,2], "pareto", start = list(shape = 0.1, scale = 
```

```
## $start.arg
## $start.arg$shape
## [1] 0.1
##
## $start.arg$scale
## [1] 10
##
##
## $fix.arg
## NULL
```

```
pareto_ra_rac_Ecoli_GC_6 <- fitdist(ra_rac_Ecoli_GC_6[,2], "pareto", start = list(shape = 0.1, scale = 
```

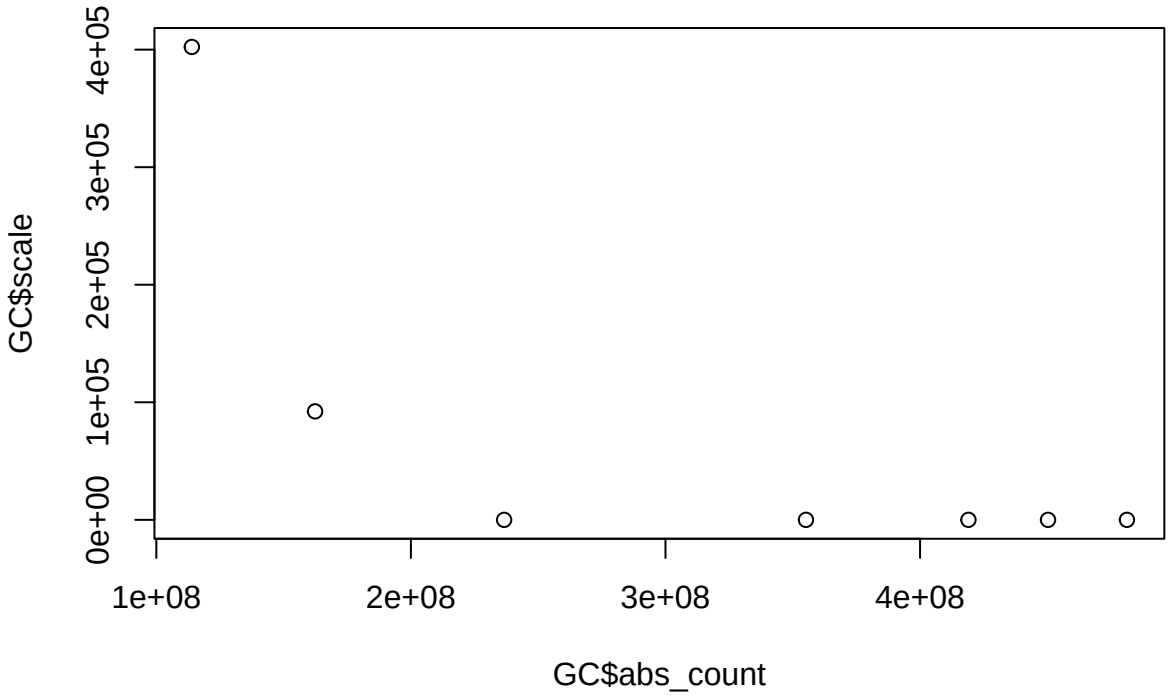
```
## $start.arg
## $start.arg$shape
## [1] 0.1
##
## $start.arg$scale
## [1] 10
##
##
## $fix.arg
## NULL
```

```
pareto_ra_rac_Ecoli_GC_7 <- fitdist(ra_rac_Ecoli_GC_7[,2], "pareto", start = list(shape = 0.1, scale = 
```

```
## $start.arg
## $start.arg$shape
## [1] 0.1
```

```
##
## $start.arg$scale
## [1] 10
##
##
## $fix.arg
## NULL
```

```
shape <- data.frame()
scale <- data.frame()
shapescale <- list(pareto_ra_rac_Ecoli_GC_1, pareto_ra_rac_Ecoli_GC_2, pareto_ra_rac_Ecoli_GC_3, pareto
for(n in shapescale){
  shape <- rbind(shape, n$estimate[1])
  scale <- rbind(scale, n$estimate[2])
}
GC <- data.frame(deriv, OD600_Ecoli_GC$OD600, OD600_Ecoli_GC$abs_count, shape, scale)
names(GC) <- c("SGR", "OD600", "abs_count", "shape", "scale")
plot(GC$abs_count, GC$scale)
```




```
png(filename="./output/GrowthCurve.Pareto_parameters.png",
     width = 900, height = 900, res = 96*2)

par(fig=c(0,0.5,0.5,1))
plot(GC$OD600, log(GC$shape), type = 'p', pch = 19, ylab = "", xlab = "", yaxt = "n", xaxt = "n", col =
mtext("OD600", side = 1, line = 1.2, cex = 0.8)
mtext("log(shape)", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
```

```
axis(1, cex.axis = 0.7,  mgp = c(3, 0.5, 0))
par(fig=c(0.5,1,0.5,1), new = TRUE)
plot(GC$OD600, log(GC$scale), type = 'p', pch = 19, ylab = "", xlab = "", yaxt = "n", xaxt = "n", col =
mtext("OD600", side = 1, line = 1.2, cex = 0.8)
mtext("log(scale)", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7,  mgp = c(3, 0.5, 0))

par(fig=c(0,0.5,0.25,0.75), new = TRUE)
plot(GC$SGR, log(GC$shape), type = 'p', pch = 19, ylab = "", xlab = "", yaxt = "n", xaxt = "n", col = "
mtext("?", side = 1, line = 1.2, cex = 0.8)
mtext("log(shape)", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7,  mgp = c(3, 0.5, 0))
par(fig=c(0.5,1,0.25,0.75), new = TRUE)
plot(GC$SGR, log(GC$scale), type = 'p', pch = 19, ylab = "", xlab = "", yaxt = "n", xaxt = "n", col = "
mtext("?", side = 1, line = 1.2, cex = 0.8)
mtext("log(scale)", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7,  mgp = c(3, 0.5, 0))

par(fig=c(0,0.5,0,0.5), new = TRUE)
plot(GC$abs_count, log(GC$shape), type = 'p', pch = 19, ylab = "", xlab = "", yaxt = "n", xaxt = "n")
mtext("abs. count", side = 1, line = 1.2, cex = 0.8)
mtext("log(shape)", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7,  mgp = c(3, 0.5, 0))
par(fig=c(0.5,1,0,0.5), new = TRUE)
plot(GC$abs_count, log(GC$scale), type = 'p', pch = 19, ylab = "", xlab = "", yaxt = "n", xaxt = "n")
mtext("abs.count", side = 1, line = 1.2, cex = 0.8)
mtext("log(scale)", side = 2, line = 1.1, cex = 0.8)
axis(2,cex.axis=0.7,  mgp = c(3, 0.5, 0))
axis(1, cex.axis = 0.7,  mgp = c(3, 0.5, 0))

dev.off()
```

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

#MURI LINES

#Process MURI lines for and return ranked lists

```
#relative abundance
ra_rac_Spo0A_1day_S1 <- process("./data/FCS/EAM_20190403_MURI_1day/Stained_0S1.fcs", "BL1-H", TRUE)
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
ra_rac_Spo0A_10day_S1 <- process("./data/FCS/EAM_20190403_MURI_10day/Specimen1_1S2.fcs", "BL1-H", TRUE)
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
ra_rac_Spo0A_100day_S1 <- process("./data/FCS/EAM_20190403_MURI_100day/Stained_2S1-1.fcs", "BL1-H", TRUE
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
ra_Cdist_Spo0A_1day_S1 <- CDist(ra_rac_Spo0A_1day_S1[,2])
ra_Cdist_Spo0A_10day_S1 <- CDist(ra_rac_Spo0A_10day_S1[,2])
ra_Cdist_Spo0A_100day_S1 <- CDist(ra_rac_Spo0A_100day_S1[,2])

#absolute abundance
aa_rac_Spo0A_1day_S1 <- process("./data/FCS/EAM_20190403_MURI_1day/Stained_0S1.fcs", "BL1-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
aa_rac_Spo0A_10day_S1 <- process("./data/FCS/EAM_20190403_MURI_10day/Specimen1_1S2.fcs", "BL1-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'

## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
aa_rac_Spo0A_100day_S1 <- process("./data/FCS/EAM_20190403_MURI_100day/Stained_2S1-1.fcs", "BL1-H")
```

```
## Warning in readFCSdata(con, offsets, txt, transformation, which.lines, scale, : Some data values of
##  To avoid truncation, either fix $PnR before generating FCS or set 'truncate_max_range = FALSE'
```

```
aa_Cdist_Spo0A_1day_S1 <- CDist(aa_rac_Spo0A_1day_S1[,2])
aa_Cdist_Spo0A_10day_S1 <- CDist(aa_rac_Spo0A_10day_S1[,2])
aa_Cdist_Spo0A_100day_S1 <- CDist(aa_rac_Spo0A_100day_S1[,2])
```

Relative and Absolute Abundance MURI lines

```
png(filename="./output/MURI.Spo0A_RSG_ra_aa.png",
     width = 1800, height = 900, res = 96*2)

par(fig=c(0,0.5,0,1))

plot(ra_rac_Spo0A_1day_S1[,1], log(ra_rac_Spo0A_1day_S1[,2]), type = 'p', axes = F, xlab = "Rank in acti
mtext("log(RSG-H value)", side = 2, line = 3, cex = 1.4)
mtext("Relative abundance", side = 3, line = 1, cex = 2)
points(ra_rac_Spo0A_10day_S1[,1], log(ra_rac_Spo0A_10day_S1[,2]), type = 'p', col = "red")
points(ra_rac_Spo0A_100day_S1[,1], log(ra_rac_Spo0A_100day_S1[,2]), type = 'p', col = "blue")
box()
axis(side = 1, labels = T, cex.axis = 1.25)
axis(side = 2, las = 1, cex.axis = 1.25)

par(fig=c(0.5,1,0,1), new = TRUE)

plot(aa_rac_Spo0A_1day_S1[,1], log(aa_rac_Spo0A_1day_S1[,2]), type = 'p', axes = F, xlab = "Rank in acti
mtext("log(RSG-H value)", side = 2, line = 3, cex = 1.4)
mtext("Absolute abundance", side = 3, line = 1, cex = 2)
points(aa_rac_Spo0A_10day_S1[,1], log(aa_rac_Spo0A_10day_S1[,2]), type = 'p', col = "red")
points(aa_rac_Spo0A_100day_S1[,1], log(aa_rac_Spo0A_100day_S1[,2]), type = 'p', col = "blue")
legend(10000,4.5, c("1 day", "10 day", "100 day"), c("black", "red", "blue"), cex = 1)
box()
axis(side = 1, labels = T, cex.axis = 1.25)
axis(side = 2, las = 1, cex.axis = 1.25)

dev.off()
```

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

```
#MURI lines absolute abundance CDF and RAC
png(filename="./output/MURI.Spo0A_RSG_aaCDF.png",
     width = 1800, height = 900, res = 96*2)

par(fig=c(0,0.5,0,1))

plot(aa_rac_Spo0A_1day_S1[,1], log(aa_rac_Spo0A_1day_S1[,2]), type = 'p', axes = F, xlab = "", ylab = "
mtext("Rank in activity", side = 1, line = 2.75, cex = 1)
mtext("log(RedoxSensor Green activity)", side = 2, line = 2.75, cex = 1)
points(aa_rac_Spo0A_10day_S1[,1], log(aa_rac_Spo0A_10day_S1[,2]), type = 'p', col = "red", cex = 0.75)
points(aa_rac_Spo0A_100day_S1[,1], log(aa_rac_Spo0A_100day_S1[,2]), type = 'p', col = "blue", cex = 0.75
box()
```

```
axis(side = 1, labels = T)
axis(side = 2, las = 1)

par(fig=c(0.5,1,0,1), new = TRUE)


plot(aa_Cdist_Spo0A_1day_S1[,2], aa_Cdist_Spo0A_1day_S1[,1], type = 'p', axes = F, xlab = "", ylab = ""
mtext("% of rank-ordered cells \n contributing to activity", side = 1, line = 3.5)
mtext("Cumulative % \n RedoxSensor Green activity", side = 2, line = 2.5)
points(aa_Cdist_Spo0A_10day_S1[,2], aa_Cdist_Spo0A_10day_S1[,1], type = 'p', col = "red", cex = 0.75)
points(aa_Cdist_Spo0A_100day_S1[,2], aa_Cdist_Spo0A_100day_S1[,1], type = 'p', col = "blue", cex = 0.75)
abline(1,1)
box()
axis(side = 1, labels = T)
axis(side = 2, las = 1)
legend(65,30, c("1 day", "10 day", "100 day"), c("black", "red", "blue"))
text(90,80, "1:1")

mtext("Absolute abundance", outer = TRUE, cex = 2, line =-2.5)

dev.off()
```

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

##Figure for MURI Equipment Grant Report (relative abundance activity)

```
#MURI lines relative abundance CDF and RAC
png(filename="./output/MURI.Spo0A_RSG_raCDF.png",
     width = 1800, height = 900, res = 96*2)

par(fig=c(0,0.5,0,1))

plot(ra_rac_Spo0A_1day_S1[,1], log(ra_rac_Spo0A_1day_S1[,2]), type = 'p', axes = F, xlab = "", ylab = "
mtext("Rank in activity", side = 1, line = 2.75, cex = 1)
mtext("log(RedoxSensor Green activity)", side = 2, line = 2.75, cex = 1)
points(ra_rac_Spo0A_10day_S1[,1], log(ra_rac_Spo0A_10day_S1[,2]), type = 'p', col = "red", cex = 0.75)
points(ra_rac_Spo0A_100day_S1[,1], log(ra_rac_Spo0A_100day_S1[,2]), type = 'p', col = "blue", cex = 0.75
box()
axis(side = 1, labels = T)
axis(side = 2, las = 1)

par(fig=c(0.5,1,0,1), new = TRUE)

plot(ra_Cdist_Spo0A_1day_S1[,2], ra_Cdist_Spo0A_1day_S1[,1], type = 'p', axes = F, xlab = "", ylab = ""
mtext("% of rank-ordered cells \n contributing to activity", side = 1, line = 3.5)
mtext("Cumulative % \n RedoxSensor Green activity", side = 2, line = 2.5)
points(ra_Cdist_Spo0A_10day_S1[,2], ra_Cdist_Spo0A_10day_S1[,1], type = 'p', col = "red", cex = 0.75)
points(ra_Cdist_Spo0A_100day_S1[,2], ra_Cdist_Spo0A_100day_S1[,1], type = 'p', col = "blue", cex = 0.75)
legend(65,30, c("1 day", "10 day", "100 day"), c("black", "red", "blue"))
abline(1,1)
text(90,80, "1:1")
box()
axis(side = 1, labels = T)
axis(side = 2, las = 1)
```

```
mtext("Relative abundance", outer = TRUE, cex = 2, line =-2.5)
dev.off()
```

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