rawdata_normalization

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x tidyr::extract()   masks magrittr::extract()
## x dplyr::filter()    masks stats::filter()
## x dplyr::lag()       masks stats::lag()
## x purrr::set_names() masks magrittr::set_names()
```

Reading in data

```r
# Reading in raw data
rawdata <-
   read.table(
      "Core_facility_results/data/gene_counts.tsv",
      header = TRUE,
      sep = "\t",
      check.names = FALSE
   )


## Removal of pre activated AMs
rawdata <-
 rawdata %>% dplyr::select(-c("16AMUntreated", "17AMUntreated", "19AMMtbAUX", "20AMMtbAUX"))



rownames(rawdata) <- rawdata[, 1]

# Reading in Gene info
gene.info <- read_tsv("Core_facility_results/data/gene_info.tsv")
# gene.info <- read.csv("Core_facility_results/data/gene_info.tsv", sep = "\t")
# names(gene.info)[1:(ncol(gene.info)-1)] <- names(gene.info)[2:ncol(gene.info)]
# gene.info <- gene.info %>% rownames_to_column("Gene_ID")


#Changing first column to Gene ID for joining
colnames(gene.info)[1] <- "Gene_ID"

#Reading in Sample info
sample.info <-
```

```
   read.csv("Core_facility_results/data/sample_info.tsv", sep = "\t")

## Removal of pre activated AMs from the metainfo
sample.info <- sample.info[-c(16, 17, 19, 20), ]

# Add condition colum to sample.info containing group and treatment
sample.info[, "Condition"] <-
   factor(paste(sample.info$Sample_Group,
                sample.info$Treatment, sep = "."))


genes <- semi_join(gene.info, rawdata, by = "Gene_ID")
rawdata <- rawdata %>% dplyr::select(-Gene_ID)

rawdata <- as.matrix(rawdata)
```

DGE Object

```
# Defining group for DGE object
group <- sample.info$Condition

# Creating dge object which will contain read counts, sample info and gene info
dge_object2 <- DGEList(rawdata, group = group)


#adding treatment to dge$samples
Treatment <- factor(sample.info$Treatment)
dge_object2$samples$Treatment <- Treatment

#adding cell type
Cell_type <- factor(sample.info$Sample_Group)
dge_object2$samples$Cell_type <- Cell_type

# Removing duplicate gene entries
genes <- genes[!duplicated(genes$Gene_ID), ]
# Adding gene info to dge object
dge_object2$genes <- genes

# For later use
samplenames <- colnames(rawdata)
```

```
# Removing duplicate gene entries
genes <- genes[!duplicated(genes$Gene_ID),]
# Adding gene info to dge object
dge_object2$genes <- genes
```

Filtering low counts

```
# Creating a model matrix without intercept
mm <- model.matrix(~0 + group)

# Naming the columns in the model matrix
colnames(mm) <- gsub("group", "", colnames(mm))
```

```
# Finding genes to remove using edgeR flterByExpr()
keep.exprs <- filterByExpr(dge_object2, mm)
dge_object2 <- dge_object2[keep.exprs,, keep.lib.sizes=FALSE]
```

Calculating normalization factors

```
unormalized_dge <- dge_object2

# TMM normalization
dge_object2 <- calcNormFactors(dge_object2, method = "TMM")

dge_object2$samples
```

```
##                           group lib.size norm.factors Treatment Cell_type
## 1iMACUntreated  iMACs.Untreated 15998962    1.0891601 Untreated     iMACs
## 2iMACUntreated  iMACs.Untreated 17409260    1.1559715 Untreated     iMACs
## 3iMACUntreated  iMACs.Untreated 14740955    1.1034883 Untreated     iMACs
## 4iMACMtbAUX        iMACs.MtbAUX 18017027    0.9419256    MtbAUX     iMACs
## 5iMACMtbAUX        iMACs.MtbAUX 16224426    1.0040381    MtbAUX     iMACs
## 6iMACMtbAUX        iMACs.MtbAUX 16440071    0.7976455    MtbAUX     iMACs
## 7iMACLPS              iMACs.LPS 19233878    0.7138140       LPS     iMACs
## 8MDMUntreated     MDM.Untreated 16222615    0.9473939 Untreated       MDM
## 9MDMUntreated     MDM.Untreated 17017126    0.9431496 Untreated       MDM
## 10MDMUntreated    MDM.Untreated 19136785    0.9304014 Untreated       MDM
## 11MDMMtbAUX          MDM.MtbAUX 16783933    0.9766687    MtbAUX       MDM
## 12MDMMtbAUX          MDM.MtbAUX 19244732    0.9203157    MtbAUX       MDM
## 13MDMMtbAUX          MDM.MtbAUX 15293231    0.9369498    MtbAUX       MDM
## 14MDMLPS                MDM.LPS 15441454    0.9400197       LPS       MDM
## 15AMUntreated      AM.Untreated 13566293    1.0642814 Untreated        AM
## 18AMMtbAUX            AM.MtbAUX 13066588    1.0040790    MtbAUX        AM
## 21AMLPS                  AM.LPS 13983059    1.0074879       LPS        AM
## 22THP1Untreated  THP1.Untreated 18334982    1.2554687 Untreated      THP1
## 23THP1LPS              THP1.LPS 14535895    1.2765425       LPS      THP1
## 24THP1MtbAUX        THP1.MtbAUX 18204303    1.1770835    MtbAUX      THP1
```

```
# Counts per million
tmm <- cpm(dge_object2)

# Log Counts per million
## This is the normalized counts used for WGCNA analysis
norm_exp_matrix_am_rm <- cpm(dge_object2, log = TRUE, prior.count = 1)

norm_exp_matrix_am_rm_notlog <- cpm(dge_object2, log = FALSE, prior.count = 1)
```

Making different versions of the normalized data for use in other scripts

```
norm_exp_as_df_am_rm <-
    norm_exp_matrix_am_rm %>% as.data.frame() %>% rownames_to_column("Gene_ID")

#norm_exp_as_df_am_rm <- map_df(norm_exp_as_df_am_rm, ~gsub("-4.0476267*", NA, .x))

norm_exp_as_df_am_rm_notlog <-
```

```
    norm_exp_matrix_am_rm_notlog %>% as.data.frame() %>% rownames_to_column("Gene_ID")


avg_norm_exp_as_df_am_rm <- norm_exp_as_df_am_rm %>%
   mutate(
      "iMACs.Untreated" = rowMeans(norm_exp_as_df_am_rm[2:4]),
      "iMACs.MtbAUX" = rowMeans(norm_exp_as_df_am_rm[5:7]),
      "MDM.Untreated" = rowMeans(norm_exp_as_df_am_rm[9:11]),
      "MDM.MtbAUX" = rowMeans(norm_exp_as_df_am_rm[12:14])
   ) %>%
   dplyr::select(c(-2:-7,-9:-14)) %>%
   relocate(10:13, .after = 1) %>% relocate(6, .after = 3)


colnames(avg_norm_exp_as_df_am_rm) <- c("Gene_ID", as.vector(unique(group)))


#notlog
avg_norm_exp_as_df_am_rm_notlog <- norm_exp_as_df_am_rm_notlog %>%
   mutate(
      "iMACs.Untreated" = rowMeans(norm_exp_as_df_am_rm_notlog[2:4]),
      "iMACs.MtbAUX" = rowMeans(norm_exp_as_df_am_rm_notlog[5:7]),
      "MDM.Untreated" = rowMeans(norm_exp_as_df_am_rm_notlog[9:11]),
      "MDM.MtbAUX" = rowMeans(norm_exp_as_df_am_rm_notlog[12:14])
   ) %>%
   dplyr::select(c(-2:-7,-9:-14)) %>%
   relocate(10:13, .after = 1) %>% relocate(6, .after = 3)


colnames(avg_norm_exp_as_df_am_rm_notlog) <- c("Gene_ID", as.vector(unique(group)))



z_transformed_norm_exp_am_rm <-
    t(scale(
      t(
        norm_exp_as_df_am_rm %>% as.tibble() %>% column_to_rownames(var = "Gene_ID") %>%
          as.matrix()
      )
    ))




  z_transformed_avg_norm_exp_am_rm <-
    t(scale(
      t(
        avg_norm_exp_as_df_am_rm %>% as.tibble() %>%
          column_to_rownames(var = "Gene_ID") %>%
          as.matrix()
      )
    ))
```

```r
avg_norm_exp_as_matrix_am_rm <- avg_norm_exp_as_df_am_rm %>% as.tibble() %>%
          column_to_rownames(var = "Gene_ID") %>%
          as.matrix()


# norm_exp_as_df_am_rm
# norm_exp_matrix_am_rm
# avg_norm_exp_as_df_am_rm
# avg_norm_exp_as_matrix_am_rm
# z_transformed_norm_exp_am_rm
# z_transformed_avg_norm_exp_am_rm
```