Tuberculosis Bulk RNA-seq Analysis

Table of contents

1 Downloading and Formatting the Data

(download data from GSE, establish needed local directories, format data objects)

(describe following code)

```
library(tidyverse) # dplyr, tidyr, tibble, readr
library(readxl)
library(Biobase)
library(GEOquery)

dir.create(file.path("data_raw"), showWarnings = FALSE) # establish directory to raw data downloads

supplemental_dir <- file.path("data_raw", "GSE107994") # establish path to specific study directory
```

(describe following code)

```
if (!dir.exists(supplemental_dir) || length(list.files(supplemental_dir)) == 0) { # checks for supple
  getGEOSuppFiles("GSE107994", baseDir = "data_raw") # accesses GEO for supplemental files (gene cour
}

counts_file <- list.files( # grabs the specific file name for the raw gene counts data
  path = supplemental_dir,
  pattern = "Raw_counts.*Leicester.*\\.xlsx$",
  full.names = TRUE
)

counts <- readxl::read_xlsx(counts_file) # function to import .xlsx file as data table
```

(describe following code)

```
gse <- getGEO("GSE107994", GSEMatrix = TRUE) # grabs the series matrix from GEO
phenotypes <- Biobase::pData(gse[[1]]) |> as_tibble() # formats GSE download as tabular sample metada
```

(describe following code)

```
dir.create("data_processed", showWarnings = FALSE) # create data_processed directory folder if it doe

readr::write_tsv(phenotypes, file.path("data_processed", "GSE107994_phenotypes.tsv")) # save phenotyp
readr::write_tsv(counts, file.path("data_processed", "GSE107994_counts_raw.tsv")) # save counts local
```

2 Inspecting the Data

(snapshot of counts and phenotypes tables,)

```
print(counts)
```

```
# A tibble: 58,051 x 178
   Genes    Gene_name Gene_biotype Leicester_with_progr~1 Leicester_with_progr~2
   <chr>    <chr>     <chr>                         <dbl>                   <dbl>
 1 ENSG000~ TSPAN6    protein_cod~                      1                      16
 2 ENSG000~ TNMD      protein_cod~                      0                       0
 3 ENSG000~ DPM1      protein_cod~                    215                     263
 4 ENSG000~ SCYL3     protein_cod~                    233                     333
 5 ENSG000~ C1orf112  protein_cod~                     54                      57
 6 ENSG000~ FGR       protein_cod~                  21694                   18130
 7 ENSG000~ CFH       protein_cod~                     46                      44
 8 ENSG000~ FUCA2     protein_cod~                    604                     428
 9 ENSG000~ GCLC      protein_cod~                     69                     153
10 ENSG000~ NFYA      protein_cod~                    310                     420
# i 58,041 more rows
# i abbreviated names: 1: Leicester_with_progressor_longitudinal_Sample1,
#   2: Leicester_with_progressor_longitudinal_Sample2
# i 173 more variables: Leicester_with_progressor_longitudinal_Sample3 <dbl>,
#   Leicester_with_progressor_longitudinal_Sample4 <dbl>,
#   Leicester_with_progressor_longitudinal_Sample5 <dbl>,
#   Leicester_with_progressor_longitudinal_Sample6 <dbl>, ...
```

```
print(phenotypes)
```

```
# A tibble: 175 x 60
   title             geo_accession status submission_date last_update_date type
   <chr>             <chr>         <chr>  <chr>           <chr>            <chr>
 1 Leicester_with_p~ GSM2886274    Publi~ Dec 12 2017     May 15 2019      SRA
 2 Leicester_with_p~ GSM2886275    Publi~ Dec 12 2017     May 15 2019      SRA
 3 Leicester_with_p~ GSM2886276    Publi~ Dec 12 2017     May 15 2019      SRA
 4 Leicester_with_p~ GSM2886277    Publi~ Dec 12 2017     May 15 2019      SRA
 5 Leicester_with_p~ GSM2886278    Publi~ Dec 12 2017     May 15 2019      SRA
 6 Leicester_with_p~ GSM2886279    Publi~ Dec 12 2017     May 15 2019      SRA
 7 Leicester_with_p~ GSM2886280    Publi~ Dec 12 2017     May 15 2019      SRA
 8 Leicester_with_p~ GSM2886281    Publi~ Dec 12 2017     May 15 2019      SRA
 9 Leicester_with_p~ GSM2886282    Publi~ Dec 12 2017     May 15 2019      SRA
10 Leicester_with_p~ GSM2886283    Publi~ Dec 12 2017     May 15 2019      SRA
# i 165 more rows
# i 54 more variables: channel_count <chr>, source_name_ch1 <chr>,
#   organism_ch1 <chr>, characteristics_ch1 <chr>, characteristics_ch1.1 <chr>,
#   characteristics_ch1.2 <chr>, characteristics_ch1.3 <chr>,
#   characteristics_ch1.4 <chr>, characteristics_ch1.5 <chr>,
#   characteristics_ch1.6 <chr>, characteristics_ch1.7 <chr>,
#   characteristics_ch1.8 <chr>, characteristics_ch1.9 <chr>, ...
```

3 Data Cleaup

(reformat data, use one of the BioConductor packages to normalize, then use normalized data to plot a PCA)

```
counts_mat <- counts |> # convert counts tibble into a simple matrix
  select(-Gene_name, -Gene_biotype) |>
  column_to_rownames("Genes") |>
  as.matrix()

colnames(counts_mat) <- phenotypes$geo_accession[match(colnames(counts_mat), phenotypes$title)] # rep

phenotypes_slim <- phenotypes |> # subset phenotypes to the columns of interest
  select(
    case = geo_accession,
    patient_id = 'patient_id:ch1',
    group = 'group:ch1',
    tb_disease_type = 'tb_disease_type:ch1',
    smear_result = 'smear_result:ch1',
    outlier = 'outlier:ch1',
    gender = 'gender:ch1',
    ethnicity = 'ethnicity:ch1',
    birthplace = 'birth_place:ch1',
    age_baseline = 'age_at_baseline_visit:ch1',
    timepoint_months = 'timepoint_months:ch1',
    visit_date = 'visit_date:ch1',
    title
) |>
  mutate(
    group = factor(group, levels = c("Control", "Active_TB", "LTBI", "LTBI_Progressor")),
    tb_disease_type = factor(tb_disease_type),
    smear_result = factor(smear_result, levels = c("Negative", "Positive")),
    gender = factor(gender)
  )
```

4 Scenario 1

(look at baseline samples, outlier = no, all disease types, and both sexes) (DESeq2, edgeR, and limma analyses of this subset)