T2D: Exercise and Genetic Expression Profile

Program

Francois Collin, Ph.D.

`r format(Sys.time(), '%Y-%m-%d')`

Table of contents

Preamble

Francois Collin
2022-07-01

Development version:

- only the program is displayed within these pages.
- no data is attached to the repository or displayed within the pages.
- no output is displayed within the pages.

Outputs will be included and made available within the program if the associated manuscript is accepted for publication in peer-review journal.

Analysis Environment

The analysis was realized with R (v4.2.0, R Core Team 2022). Note the use of the additional packages downloaded from the RStudio package manager repository (freeze date 2022-05-04, repos) and complemented by Bioconductor packages (release 3.15):

- `BiocParallel`
- `car` for regression model test (version 3.0-13, Fox and Weisberg 2019)
- `cowplot`
- `DESeq2` for differential expression analysis and variance stabilizing transformation (version 1.36.0, Love, Huber, and Anders 2014)
- `emmeans` for least-Squares Means (LSM) estimations (version 1.7.3, Lenth 2022)
- `flextable` for output table formatting (version 0.7.0, Gohel 2022)
- `ggplot2` for graphics (version 3.3.6, Wickham 2016)
- `hexbin`
- `lintr`
- `missMDA`
- `multcomp`
- `multcompView`

- `MultiAssayExperiment` for management of multi-omics experiment objects (version 1.22.0, Ramos et al. 2017)
- `tidyr` for data wrangling (version 1.2.0, Wickham and Girlich 2022).

To install:

- WGCNA
- dynamicTreeCut

The computational environment was containerized into a Docker image build upon `rocker/verse:4.2.0`.

1 Missingness in adlb and advs

Program 03

Francois Collin
2022-07-01

Data missingness was addressed by a missing-data imputation algorithm, employed to impute missing values while minimizing bias on results (R package `missMDA`, Josse and Husson 2016)); "observed cases" and "imputed data" later refers to the exclusion/inclusion of imputed data.

The analysis was realized with R (v4.2.0, R Core Team 2022). Note the use of the additional package downloaded from the RStudio package manager repository (freeze date 2022-05-04, repos):

- `ggplot2` for graphics (version 3.3.6, Wickham 2016).

1.1 Program settings

```r
params <- yaml::read_yaml("_prog.yml")
devtools::load_all("src/pkg/dbs.data")
```

```
i Loading dbs.data
```

```r
devtools::load_all("src/pkg/latarnia.utils")
```

```
i Loading latarnia.utils
```

```
Loading required package: grid
```

```
Loading required package: shiny
```

```
knitr::opts_chunk$set(results = params$knitr$results)
```

1.2 Data Preparation

```
adsl <- dbs.data::adsl
advs <- dbs.data::advs
adlb <- dbs.data::adlb
```

1.3 Figure 03 output 01

```
library(ggplot2)
gg <- rbind(adlb, advs) |>
  ggplot(aes(subjid, paramcd, fill = dtype)) +
  scale_fill_manual(values = c("white", "gray75")) +
  geom_tile(color = "gray50") +
  theme_minimal() +
  theme(
    text = element_text(size = 7),
    axis.title = element_blank(),
    axis.text.x = element_text(angle = 90, hjust = 1),
    legend.position = "none",
    legend.title = element_blank()
  )
```

```
p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 3.1",
      "Colored raster - Data missingness per Subject and Parameter",
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_figure(gg, width = unit(5.2, "inches"), height = unit(3.5, "inches")) |>
  add_footer(
    paste0("Program t2d_03_bds / Env ayup_dbs:", params$dock$version),
    params$version
  )
```

```
export_as(
  p,
  file = file.path(params$paths$grh, "fig_03_01.pdf"),
  file_graph_alone = file.path(params$paths$grh, "fig_03_01_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_03_01.pdf
```

```
[log] output saved as: ../tlg/graph/fig_03_01_af.pdf (annot. free)
```

```
show_slate(p)
```

1.4 Session Informations

```
sessioninfo::session_info()
```

2 Demographics and Baseline Anthropometrics

Program 08

Francois Collin
2022-07-01

Demographics data characterized the diabetes groups in terms of age, number of training composing the exercise intervention, BMI and diet at baseline. This was completed by a fine description of the baseline anthropometrics leading differences between diabetes groups. Both characterizations relied on one-way analysis of variance, the diabetes effect significance was ruled by a Fisher test, least mean square estimations were obtained for every diabetes group along with their 95% confidence interval, and pairwise difference estimation and significance relied on Tukey's method.

The analysis was realized with R (v4.2.0, R Core Team 2022). Note the use of the additional packages downloaded from the RStudio package manager repository (freeze date 2022-05-04, repos):

- `car` for regression model test (version 3.0-13, Fox and Weisberg 2019)
- `emmeans` for least-Squares Means (LSM) estimations (version 1.7.3, Lenth 2022)
- `flextable` for output table formatting (version 0.7.0, Gohel 2022)
- `tidyr` for data wrangling (version 1.2.0, Wickham and Girlich 2022).

Target:

☒ Table: Demographics and baseline anthropometrics are tested via an Anova.
☒ Supp. Table: Post-hoc estimations / tests by diabetes groups.
☒ Supp. Table: extension of the anova to additional ADVS/ADLB parameters.

Specifications:

- Variable order

```r
params <- yaml::read_yaml("_prog.yml")
devtools::load_all("src/pkg/dbs.data")
```

```
i Loading dbs.data
```

```r
devtools::load_all("src/pkg/latarnia.utils")
```

```
i Loading latarnia.utils

Loading required package: grid

Loading required package: shiny
```

```r
knitr::opts_chunk$set(results = params$knitr$results)
```

```r
adsl <- dbs.data::adsl
advs <- dbs.data::advs
adlb <- dbs.data::adlb
```

```r
ads <- adlb |>
  rbind(advs) |>
  subset(dtype == "" & avisit == "Baseline") |>
  subset(select = c(subjid, paramcd, avisit, aval, dtype)) |>
  rbind(
    adsl |>
      subset(select = -c(diab, diabcd)) |>
      tidyr::pivot_longer(
        cols = c("age", "trainn"), names_to = "paramcd", values_to = "aval"
      ) |>
      within(dtype <- "") |>
      within(avisit <- "Baseline")
  ) |>
  (\(x) merge(x = adsl[c("subjid", "diabcd")], y = x, by = "subjid"))() |>
  (\(df, fct = "diabcd") {
    df[paste0(fct, "_n")] <- factor_n(df, fct, id = "subjid", sep = " ")
    df
  })() |>
  within(
    paramcd <- factor(
      paramcd,
      levels = c(
        "age", "trainn", "GLU0", "GLU30", "GLU60", "GLU120",
        "INSULIN0", "INSULIN30", "INSULIN60", "INSULIN120",
        "HBA1C", "HOMAB", "HOMAIR", "MATSUDA",
        "TRIG", "CHOL", "LDL", "HDL", "VO2MAXE", "WEIGHT", "BMI",
        "FATMASS", "BODYFATP", "LEANMASS", "VAT",
        "DCAL", "FATACFR", "DCARBT", "DFATT", "DPROT", "SMMASS",
        "VO2MAXLBM", "VO2MAXML"
      )
    )
  ) |>
  (\(df) df[order(df$paramcd), ])()
```

```
format_pval <- function(x) {
  p <- round(x, 5)
  ifelse(
    test = p < 0.0001,
    yes = "<0.0001",
    no = ifelse(
      test = p < 0.001,
      yes = format(round(p, 4), nsmall = 4),
      no = ifelse(
        test = p < 0.01,
        yes = format(round(p, 3), nsmall = 3),
        no = format(round(p, 2), nsmall = 2)
      )
    )
  )
}

lm_by_paramcd <- function(x,
                          dep_var = "aval",
                          indep_var = "diabcd_n",
                          covariate = NULL) {

  formula <- paste(
    dep_var, "~",
    if (!is.null(covariate)) paste(covariate, "+"),
    indep_var
  )
  formula <- as.formula(formula)

  lapply(
    x,
    \(x) list(data = x, lm = lm(formula, data = x))
  )
}

make_specs <- function(var) as.formula(paste("~", var))
lsm_by_param <- function(x, indep_var = "diabcd_n") {
  lapply(
    x,
    \(x) {
      mod_em <- emmeans::emmeans(x$lm, specs = make_specs(indep_var))
      y <- as.data.frame(mod_em)
      cbind(
        paramcd = unique(x$data$paramcd),
        y,
        diabcd_f = car::Anova(x$lm)[indep_var, "Pr(>F)"]
      )
    }
  )
}

lsm_pairs_by_param <- function(x, indep_var = "diabcd_n")
  lapply(
    x,
    \(x) {
      mod_em <- emmeans::emmeans(
        x$lm, specs = indep_var, contr = "revpairwise"
      )
      y <- merge(
        as.data.frame(mod_em$contrast)[c("contrast", "p.value")],
        confint(mod_em)$contrasts
      )
      cbind(paramcd = unique(x$data$paramcd), y)
    }
  )
```

2.1 Tab 08 01 - Demographics and Baseline Anthropometrics by Diabetes Group

```r
tab_08_01_raw <- ads |>
  subset(
    paramcd %in% c(
      "age", "trainn", "GLU0", "GLU30", "GLU60", "GLU120",
      "INSULIN0", "INSULIN30", "INSULIN60", "INSULIN120",
      "HBA1C", "HOMAB", "HOMAIR", "MATSUDA",
      "TRIG", "CHOL", "LDL", "HDL", "VO2MAXE", "WEIGHT", "BMI",
      "FATMASS", "BODYFATP", "LEANMASS", "VAT"
    )
  ) |>
  (\(x) split(x, f = x$paramcd, drop = TRUE))() |>
  lm_by_paramcd() |>
  lsm_by_param() |>
  (\(x) Reduce(rbind, x))()

library(tidyr)
```

```
Attaching package: 'tidyr'

The following object is masked from 'package:testthat':

    matches
```

```r
tab_08_01 <- tab_08_01_raw |>
  (\(df) df[order(df$diabcd_n), ])() |>
  within({
    val <- paste0(
      signif(emmean, 3),
      " (", signif(lower.CL, 3), ", ", signif(upper.CL, 3), ")"
    )
    pval <- format_pval(diabcd_f)
  }) |>
  pivot_wider(
    id_cols = c("paramcd", "pval", "diabcd_f"),
    values_from = "val",
    names_from = "diabcd_n"
  )
tab_08_01
```

```
library(flextable)
```

```
Attaching package: 'flextable'

The following objects are masked from 'package:latarnia.utils':

    add_footer, add_header
```

```
tab_08_01_ft <- tab_08_01 |>
  subset(select = -diabcd_f) |>
  flextable() |>
  autofit() |>
  add_header_lines(wrap_long_lines(
    "Analysis Set: Full Analysis Set - Observed Cases at baseline"
  )) |>
  set_caption(
    caption = wrap_long_lines(
      "Tab 08 01 - Analysis of Variance / Least Means Square estimations
      (95% Confidence Interval) of Demographics Parameters and Baseline
      Anthropometrics by Diabetes Group"
    )
  ) |>
  footnote(
    part = "header",
    i = 2, j = 2,
    value = as_paragraph(
      "Note: pval, p value of diabetes group effect test by F test."
    ),
    ref_symbols = "a"
  ) |>
  footnote(
    value = as_paragraph(
      "Source: ADSL and ADVS/ADLB observed cases at baseline."
    ),
    ref_symbols = ""
  ) |>
```

```r
  theme_booktabs()
tab_08_01_ft
```

```
Warning: Warning: fonts used in `flextable` are ignored because the `pdflatex`
engine is used and not `xelatex` or `lualatex`. You can avoid this warning
by using the `set_flextable_defaults(fonts_ignore=TRUE)` command or use a
compatible engine by defining `latex_engine: xelatex` in the YAML header of the
R Markdown document.
```

```r
bnm <- "tab_08_01"
dir_tab <- params$paths$tab
dir_dta <- params$paths$dta

file.path(dir_tab, paste0(bnm, "_ft.RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_08_01_ft, file = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_01_ft.RData
```

```r
file.path(dir_dta, paste0(bnm, ".RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_08_01, file = .)
```

```
[output] Table saved as ../data/tab_08_01.RData
```

```r
file.path(dir_tab, paste(bnm, sep = ".", "docx")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_docx(tab_08_01_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_01.docx
```

```r
file.path(dir_tab, paste(bnm, sep = ".", "html")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_html(tab_08_01_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_01.html
```

```
file.path(dir_dta, paste(bnm, sep = ".", "csv")) %T>%
  message("[output] Table saved as ", .) %>%
  write.csv(tab_08_01, file = ., row.names = FALSE)
```

```
[output] Table saved as ../data/tab_08_01.csv
```

2.2 Tab 08 02 - Post-hoc: Demographics and Baseline Anthropometrics by Diabetes Group

```
tab_08_02_raw <- ads |>
  subset(
    paramcd %in% c(
      "age", "trainn", "GLU0", "GLU30", "GLU60", "GLU120",
      "INSULIN0", "INSULIN30", "INSULIN60", "INSULIN120",
      "HBA1C", "HOMAB", "HOMAIR", "MATSUDA",
      "TRIG", "CHOL", "LDL", "HDL", "VO2MAXE", "WEIGHT", "BMI",
      "FATMASS", "BODYFATP", "LEANMASS", "VAT"
    )
  ) |>
  (\(x) split(x, f = x$paramcd, drop = TRUE))() |>
  lm_by_paramcd(indep_var = "diabcd") |>
  lsm_pairs_by_param(indep_var = "diabcd")  |>
  (\(x) Reduce(rbind, x))()

library(tidyr)
tab_08_02 <- tab_08_02_raw |>
  subset(select = c(
    paramcd, contrast, estimate, SE, df, p.value, lower.CL, upper.CL
  ))

library(flextable)
tab_08_02_ft <- tab_08_02 |>
  (\(x) split(x, f = x$paramcd))() |>
  lapply(
    \(x) {
      x$p.value <- format(round(x$p.value, 5))
      x$estimate <- format(signif(x$estimate, 5))
      x$SE <- format(signif(x$SE, 6))
      x$lower.CL <- format(signif(x$lower.CL, 5))
      x$upper.CL <- format(signif(x$upper.CL, 5))
      x
    }) |>
  (\(x) Reduce(rbind, x))()|>
  flextable() |>
  fontsize(size = 9, part = "all") |>
  autofit() |>
  add_header_lines(
    "Analysis Set: Full Analysis Set - Observed Cases at baseline"
  ) |>
  set_caption(
    caption = wrap_long_lines(
      "Tab 08 02 - Post-hoc tests for the Analysis of Variance of
      Demographics and Baseline Anthropometrics by Diabetes Group"
    )
  ) |>
  add_footer_lines(c(
    "CL, 95% Confidence Limit; SE, Standard Error.",
    wrap_long_lines(
      "Note: P value adjustment by Tukey's method for comparing a family of
      3 estimates."
    ),
    "Source: ADSL and ADVS/ADLB observed cases at baseline."
  )) |>
  theme_booktabs()

tab_08_02_ft
```

```
Warning: Warning: fonts used in `flextable` are ignored because the `pdflatex`
engine is used and not `xelatex` or `lualatex`. You can avoid this warning
by using the `set_flextable_defaults(fonts_ignore=TRUE)` command or use a
compatible engine by defining `latex_engine: xelatex` in the YAML header of the
R Markdown document.
```

```
bnm <- "tab_08_02"
dir_tab <- params$paths$tab
dir_dta <- params$paths$dta

file.path(dir_tab, paste0(bnm, "_ft.RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_08_02_ft, file = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_02_ft.RData
```

```
file.path(dir_dta, paste0(bnm, ".RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_08_02, file = .)
```

```
[output] Table saved as ../data/tab_08_02.RData
```

```
file.path(dir_tab, paste(bnm, sep = ".", "docx")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_docx(tab_08_02_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_02.docx
```

```
file.path(dir_tab, paste(bnm, sep = ".", "html")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_html(tab_08_02_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_02.html
```

```
file.path(dir_dta, paste(bnm, sep = ".", "csv")) %T>%
  message("[output] Table saved as ", .) %>%
  write.csv(tab_08_02, file = ., row.names = FALSE)
```

```
[output] Table saved as ../data/tab_08_02.csv
```

2.3 Tab 08 03 - Demographics and Baseline Anthropometrics by Diabetes Group (additional parameters)

```
tab_08_03_raw <- ads |>
  subset(
    !paramcd %in% c(
      "age", "trainn", "GLU0", "GLU30", "GLU60", "GLU120",
      "INSULIN0", "INSULIN30", "INSULIN60", "INSULIN120",
```

```r
      "HBA1C", "HOMAB", "HOMAIR", "MATSUDA",
      "TRIG", "CHOL", "LDL", "HDL", "VO2MAXE", "WEIGHT", "BMI",
      "FATMASS", "BODYFATP", "LEANMASS", "VAT"
    )
  ) |>
  (\(x) split(x, f = x$paramcd, drop = TRUE))() |>
  lm_by_paramcd() |>
  lsm_by_param() |>
  (\(x) Reduce(rbind, x))()

library(tidyr)
tab_08_03 <- tab_08_03_raw |>
  (\(df) df[order(df$diabcd_n), ])() |>
  within({
    val <- paste0(
      signif(emmean, 3),
      " (", signif(lower.CL, 3), ", ", signif(upper.CL, 3), ")"
    )
    pval <- format_pval(diabcd_f)
  }) |>
  pivot_wider(
    id_cols = c("paramcd", "pval", "diabcd_f"),
    values_from = "val",
    names_from = "diabcd_n"
  )
tab_08_03
```

```r
library(flextable)
tab_08_03_ft <- tab_08_03 |>
  subset(select = -diabcd_f) |>
  flextable() |>
  autofit() |>
  add_header_lines(wrap_long_lines(
    "Analysis Set: Full Analysis Set - Observed Cases at baseline"
  )) |>
  set_caption(
    caption = wrap_long_lines(
      "Tab 08 03 - Analysis of Variance / Least Means Square estimations
      (95% Confidence Interval) of Demographics Parameters and Baseline
      Anthropometrics by Diabetes Group for Supplementary Parameters"
    )
  ) |>
  footnote(
    part = "header",
    i = 2, j = 2,
    value = as_paragraph(
      "Note: pval, p value of diabetes group effect test by F test."
    ),
    ref_symbols = "a"
  ) |>
  add_footer_lines("Source: ADSL and ADVS/ADLB observed cases at baseline.") |>
  theme_booktabs()
tab_08_03_ft
```

```
Warning: Warning: fonts used in `flextable` are ignored because the `pdflatex`
engine is used and not `xelatex` or `lualatex`. You can avoid this warning
by using the `set_flextable_defaults(fonts_ignore=TRUE)` command or use a
compatible engine by defining `latex_engine: xelatex` in the YAML header of the
R Markdown document.
```

```
bnm <- "tab_08_03"
dir_tab <- params$paths$tab
dir_dta <- params$paths$dta

file.path(dir_tab, paste0(bnm, "_ft.RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_08_03_ft, file = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_03_ft.RData
```

```
file.path(dir_dta, paste0(bnm, ".RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_08_03, file = .)
```

```
[output] Table saved as ../data/tab_08_03.RData
```

```
file.path(dir_tab, paste(bnm, sep = ".", "docx")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_docx(tab_08_03_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_03.docx
```

```r
file.path(dir_tab, paste(bnm, sep = ".", "html")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_html(tab_08_03_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_03.html
```

```r
file.path(dir_dta, paste(bnm, sep = ".", "csv")) %T>%
  message("[output] Table saved as ", .) %>%
  write.csv(tab_08_03, file = ., row.names = FALSE)
```

```
[output] Table saved as ../data/tab_08_03.csv
```

2.4 Tab 08 04 - Post-hoc: Demographics and Baseline Anthropometrics by Diabetes Group (additional parameters)

```r
tab_08_04_raw <- ads |>
  subset(
    ! paramcd %in% c(
      "age", "trainn", "GLU0", "GLU30", "GLU60", "GLU120",
      "INSULIN0", "INSULIN30", "INSULIN60", "INSULIN120",
      "HBA1C", "HOMAB", "HOMAIR", "MATSUDA",
      "TRIG", "CHOL", "LDL", "HDL", "VO2MAXE", "WEIGHT", "BMI",
      "FATMASS", "BODYFATP", "LEANMASS", "VAT"
    )
  ) |>
  (\(x) split(x, f = x$paramcd, drop = TRUE))() |>
  lm_by_paramcd(indep_var = "diabcd") |>
  lsm_pairs_by_param(indep_var = "diabcd") |>
  (\(x) Reduce(rbind, x))()

library(tidyr)
tab_08_04 <- tab_08_04_raw |>
  subset(select = c(
    paramcd, contrast, estimate, SE, df, p.value, lower.CL, upper.CL
  ))
```

```
library(flextable)
tab_08_04_ft <- tab_08_04 |>
  (\(x) split(x, f = x$paramcd))() |>
  lapply(
    \(x) {
      x$p.value <- format(round(x$p.value, 5))
      x$estimate <- format(signif(x$estimate, 5))
      x$SE <- format(signif(x$SE, 6))
      x$lower.CL <- format(signif(x$lower.CL, 5))
      x$upper.CL <- format(signif(x$upper.CL, 5))
      x
    }) |>
  (\(x) Reduce(rbind, x))()|>
  flextable() |>
  fontsize(size = 9, part = "all") |>
  autofit() |>
  add_header_lines(
    "Analysis Set: Full Analysis Set - Observed Cases at baseline"
  ) |>
  set_caption(
    caption = wrap_long_lines(
      "Tab 08 04 - Post-hoc tests for the Analysis of Variance of
      Demographics and Baseline Anthropometrics by Diabetes Group"
    )
  ) |>
  footnote(
    value = as_paragraph(wrap_long_lines(
      "CL, 95% Confidence Limit; SE, Standard Error."
    )),
    ref_symbols = ""
  ) |>
  footnote(
    value = as_paragraph(wrap_long_lines(
      "Note: P value adjustment by Tukey's method for comparing a family of
      3 estimates."
    )),
    ref_symbols = ""
  )|>
  footnote(
    value = as_paragraph(wrap_long_lines(
      "Source: ADSL and ADVS/ADLB observed cases at
      baseline."
    )),
    ref_symbols = ""
  ) |>
  theme_booktabs()

tab_08_04_ft
```

```
Warning: Warning: fonts used in `flextable` are ignored because the `pdflatex`
engine is used and not `xelatex` or `lualatex`. You can avoid this warning
by using the `set_flextable_defaults(fonts_ignore=TRUE)` command or use a
compatible engine by defining `latex_engine: xelatex` in the YAML header of the
R Markdown document.
```

```
bnm <- "tab_08_04"
dir_tab <- params$paths$tab
dir_dta <- params$paths$dta

file.path(dir_tab, paste0(bnm, "_ft.RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_08_04_ft, file = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_04_ft.RData
```

```
file.path(dir_dta, paste0(bnm, ".RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_08_04, file = .)
```

```
[output] Table saved as ../data/tab_08_04.RData
```

```
file.path(dir_tab, paste(bnm, sep = ".", "docx")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_docx(tab_08_04_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_04.docx
```

```
file.path(dir_tab, paste(bnm, sep = ".", "html")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_html(tab_08_04_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_08_04.html
```

```
file.path(dir_dta, paste(bnm, sep = ".", "csv")) %T>%
  message("[output] Table saved as ", .) %>%
  write.csv(tab_08_04, file = ., row.names = FALSE)
```

```
[output] Table saved as ../data/tab_08_04.csv
```

2.5 Session Informations

```
sessioninfo::session_info()
```

3 Anthropometrics Changes From Baseline

Program 09

Francois Collin
2022-07-01

Diabetes group estimations in anthropometrics changes from baseline were obtained and tested by analysis of covariance models (Ancova), relying once more on the Fisher test; to increase accuracy and statistical power, estimations were adjusted for baseline values (Vickers 2001; Van Breukelen 2006; Committee for Medicinal Products for Human Use (CHMP) 2015; O'Connell et al. 2017).

The analysis was realized with R (v4.2.0, R Core Team 2022). Note the use of the additional packages downloaded from the RStudio package manager repository (freeze date 2022-05-04, repos):

- `car` for regression model test (version 3.0-13, Fox and Weisberg 2019)
- `emmeans` for least-Squares Means (LSM) estimations (version 1.7.3, Lenth 2022)
- `flextable` for output table formatting (version 0.7.0, Gohel 2022)
- `tidyr` for data wrangling (version 1.2.0, Wickham and Girlich 2022).

3.1 Settings

```
cfg_prog <- yaml::read_yaml("_prog.yml")
devtools::load_all("src/pkg/dbs.data")
```

```
i Loading dbs.data
```

```
devtools::load_all("src/pkg/latarnia.utils")
```

```
i Loading latarnia.utils
```

```
Loading required package: grid
```

```
Loading required package: shiny
```

```r
source("R/inches.R")

knitr::opts_chunk$set(results = cfg_prog$knitr$results)
```

3.2 Data

```r
adsl <- dbs.data::adsl
advs <- dbs.data::advs
adlb <- dbs.data::adlb
```

```r
ads <- adlb |>
  rbind(advs) |>
  subset(basetype == "" & avisit != "Baseline") |>
  subset(select = c(subjid, paramcd, avisit, base, chg)) |>
  (\(x) merge(x = adsl[c("subjid", "diabcd")], y = x, by = "subjid"))() |>
  (\(df, fct = "diabcd") {
    df[paste0(fct, "_n")] <- factor_n(df, fct, id = "subjid", sep = " ")
    df
  })()

head(ads)
```

3.3 Helper functions

```r
format_pval <- function(x) {
  p <- round(x, 5)
  ifelse(
    test = p < 0.0001,
    yes = "<0.0001",
    no = ifelse(
      test = p < 0.001,
      yes = format(round(p, 4), nsmall = 4),
      no = ifelse(
        test = p < 0.01,
        yes = format(round(p, 3), nsmall = 3),
        no = format(round(p, 2), nsmall = 2)
      )
    )
  )
}

lm_by_paramcd <- function(x,
                          dep_var = "aval",
                          indep_var = "diabcd_n",
                          covariate = NULL) {

  formula <- paste(
    dep_var, "~",
    if (!is.null(covariate)) paste(covariate, "+"),
    indep_var
  )
  formula <- as.formula(formula)

  lapply(x, \(x) list(data = x, lm = lm(formula, data = x)))
}

make_specs <- function(var) as.formula(paste("~", var))
lsm_by_param <- function(x, indep_var = "diabcd_n") {
  lapply(
    x,
    \(x) {
      mod_em <- emmeans::emmeans(x$lm, specs = make_specs(indep_var))
      y <- as.data.frame(mod_em)
      cbind(
        paramcd = unique(x$data$paramcd),
        y,
        diabcd_f = car::Anova(x$lm)[indep_var, "Pr(>F)"]
      )
    }
  )
}
```

3.4 Tab 09 01 - Ancova - Anthropometrics Changes from Baseline by Diabetes Group

```r
tab_09_01_raw <- ads |>
  (\(x) split(x, f = x$paramcd))() |>
  lm_by_paramcd(dep_var = "chg", covariate = "base", indep_var = "diabcd_n") |>
  lsm_by_param(indep_var = "diabcd_n") |>
  (\(x) Reduce(rbind, x))()

tab_09_01 <- tab_09_01_raw |>
  (\(df) df[order(df$diabcd_n), ])() |>
  within({
    val <- paste0(
      signif(emmean, 3),
      " (", signif(lower.CL, 3), ", ", signif(upper.CL, 3), ")"
    )
    pval <- format_pval(diabcd_f)
  }) |>
  tidyr::pivot_wider(
    id_cols = c("paramcd", "pval", "diabcd_f"),
    values_from = "val",
    names_from = "diabcd_n"
  ) |>
  (\(x) x[order(x$diabcd_f), ])()
tab_09_01
```

```r
library(flextable)
```

```
Attaching package: 'flextable'

The following objects are masked from 'package:latarnia.utils':

    add_footer, add_header
```

```r
wrap_line <- function(x) paste(strwrap(x, width = 80), collapse = " ")
tab_09_01_ft <- tab_09_01 |>
  subset(select = -diabcd_f) |>
  flextable() |>
  autofit() |>
  footnote(
    part = "header",
    i = 1, j = 2,
    value = as_paragraph(
      "Note: pval, p value of diabetes group effect test by F test."
    ),
    ref_symbols = "a"
  ) |>
  add_footer_lines(c(
    wrap_line("Source: Full Analysis Set, observed cases at baseline and post
    intervention."),
    "Note: rows are ordered by increasing p values, most significant on top."
  )) |>
  add_header_lines("Analysis Set: Full Analysis Set - Observed Cases") |>
  set_caption(
    caption =  wrap_long_lines(
      "Tab 09 01 - Analysis of Covariance / Least Means Square estimations of
      Anthropometrics Changes from Baseline by Diagnosis Group at
      Month 3 (95% Confidence Interval) Adjusted for Baseline"
    )
  ) |>
  theme_booktabs()
tab_09_01_ft
```

```
Warning: Warning: fonts used in `flextable` are ignored because the `pdflatex`
engine is used and not `xelatex` or `lualatex`. You can avoid this warning
by using the `set_flextable_defaults(fonts_ignore=TRUE)` command or use a
compatible engine by defining `latex_engine: xelatex` in the YAML header of the
R Markdown document.
```

```
bnm <- "tab_09_01"
dir_tab <- cfg_prog$paths$tab
dir_dta <- cfg_prog$paths$dta

file.path(dir_tab, paste0(bnm, "_ft.RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_09_01_ft, file = .)
```

```
[output] Table saved as ../tlg/tables/tab_09_01_ft.RData
```

```
file.path(dir_dta, paste0(bnm, ".RData")) %T>%
  message("[output] Table saved as ", .) %>%
  save(tab_09_01, file = .)
```

```
[output] Table saved as ../data/tab_09_01.RData
```

```
file.path(dir_tab, paste(bnm, sep = ".", "docx")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_docx(tab_09_01_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_09_01.docx
```

```
file.path(dir_tab, paste(bnm, sep = ".", "html")) %T>%
  message("[output] Table saved as ", .) %>%
  save_as_html(tab_09_01_ft, path = .)
```

```
[output] Table saved as ../tlg/tables/tab_09_01.html
```

```
file.path(dir_dta, paste(bnm, sep = ".", "csv")) %T>%
  message("[output] Table saved as ", .) %>%
  write.csv(tab_09_01, file = ., row.names = FALSE)
```

```
[output] Table saved as ../data/tab_09_01.csv
```

```
detach(package:flextable)
```

3.5 Example: Glucose 120 - Fig 09 01

```
library(ggplot2)
dta <- dbs.data::adlb |>
  subset(
    paramcd == "GLU120" &
      dtype == "" &
      avisit == "Month 3" &
      basetype == ""
  ) |>
  merge(x = adsl[c("subjid", "diabcd")], y = _, by = "subjid")

dta$lm_pred <- predict(lm(aval ~ base, data = dta))
lim <- range(c(dta$aval, dta$base))

gg1 <- ggplot(dta, aes(base, aval, color = diabcd)) +
  geom_point() +
  ylab("Post exercise intervention") +
  xlab("Baseline") +
  geom_segment(aes(xend = base, yend = lm_pred)) +
  geom_smooth(method = "lm", se = FALSE, color = "black", linetype = 1) +
  scale_color_viridis_d(begin = .1, end = .9, direction = -1) +
  coord_cartesian(xlim = lim, ylim = lim) +
  theme_minimal() +
  theme(legend.position = "top")

gg2 <- ggplot(dta, aes(base, fill = diabcd, color = diabcd)) +
  geom_boxplot(alpha = .5, show.legend = FALSE) +
  coord_cartesian(xlim = lim) +
  ylab("diabcd") +
  scale_color_viridis_d(begin = .1, end = .9, direction = -1) +
  scale_fill_viridis_d(begin = .1, end = .9, direction = -1) +
  theme_minimal() +
  theme(
    legend.position = "top",
    axis.title.x = element_blank(),
    axis.text.y = element_blank(),
    panel.grid = element_blank()
  )

gg <- cowplot::plot_grid(
  gg1, gg2,
  nrow = 2,
  align = "v",
  rel_heights = c(1, .15)
)
```

```
`geom_smooth()` using formula 'y ~ x'
```

```
p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 9.1",
      strwrap(
        "Scatter plot - Glucose 120 at Month 3 in relation to Baseline by
        treatment group", width = 80
      ),
      "Analysis Set: Full Analysis Set - Observed Cases"
    )
  ) |>
  add_figure(gg, height = inches(5.5), width = inches(4.5))|>
  add_note(c(
    "Note: the table prog 08 output 01 focuses on the x-axis and check for
     differences at baseline, the boxplots gives an overview of these
     baseline variation.",
    "Note: the table tab prog 09 output 01 shows the effect of exercise in
     each group. The p.values is provided and accounts for the diagnostic group
     effect, while taking into account the fact that the higher the value at
     baseline, the higher after the exercise intervention. Actually,
     the analysis of covariance model focus on how much above or below the
     general regression (black line) is positionned a diagnosis group.
     E.g., +20 for T2D indicates that the result after exercise is 20 units
     higher than expected if there were no diagnostic group effect; ... however,
     the confidence interval extend from -13.5 and +53.8 which means that
     i) if we repeated the study, 100 times, it would result 95 times from
     -13.5 to 53.8
     ii) the interval includes 0 so we can't conclude to a significant
     increase or decrease in Glucose 120 due to the exercise intervention."
  )) |>
  add_footer(
    c(
      "Program t2d_09_chg / Env ayup_dbs:v0.1.0-alpha",
      format(Sys.time(), format = "%Y-%m-%d %H:%M (%Z)")
    ),
    cfg_prog$version
  )

export_as(
  p,
  file = file.path(cfg_prog$paths$grh, "fig_09_01.pdf"),
  file_graph_alone = file.path(cfg_prog$paths$grh, "fig_09_01_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_09_01.pdf
```

```
[log] output saved as: ../tlg/graph/fig_09_01_af.pdf (annot. free)
```

```
show_slate(p)
```

3.6 Session Informations

```
sessioninfo::session_info()
```

3.7 Untidied information about analysis of change from baseline

Questions: for the analysis of a post-treatment values, should we analyse the change from baseline or percentage change from baseline? Should we adjust for the baseline?

Back to 2003, in the context of randomized clinical trial, the European Medicines Agency (COMMITTEE FOR PROPRIETARY MEDICINAL PRODUCTS 2003), when the endpoint is studied as a change from baseline, the adjustment for baseline improves the accuracy in comparison to non-baseline adjustment; estimates becomes also equivalent to the standard linear model, the choice of change from baseline analysis or raw value is then only a question of interpretability. They renewed the recommendation in 2015 (Committee for Medicinal Products for Human Use (CHMP) 2015).

From the academic side, the topic was repeatedly studied:

- Van Breukelen (2006): > *"In randomized studies both methods [Anova, no BL adjustment vs Ancova] > are unbiased, but ANCOVA has more power"*
- Liu et al. (2009) also highlighted the benefits of adjustment for baseline as a covariate.
- this was later confirmed by Zhang et al. (2014).
- More recently O'Connell et al. (2017) also defended the superiority of the Ancova-change: $y_i = \beta_0 + \beta_1 X_i + \beta_2 Y_{0i,=BL} + \varepsilon_i$

> _“Consistent with existing literature, our results demonstrate that each method leads to unbiased treatment effect estimates, and based on precision of estimates, 95% coverage probability, and power, ANCOVA modeling of either change scores or post-treatment score as the outcome, prove to be the most effective._”

Most of the authors above are specifically working on randomized trial, Vickers (2001) also brought some light on the topic, and highlighted in addition that: working with percentage change is generally a bad idea. The extended to a theoretical works also indicated that the percentage change from baseline *“will also fail to protect from bias in the case of baseline imbalance and will lead to an excess of trials with non-normally distributed outcome data”.*

4 RNASeq - Refresher

Program 06

Francois Collin
2022-07-01

```
params <- yaml::read_yaml("_prog.yml")
```

Target:

- ☒ refresh the differential expression analysis technics with DESeq2.
- ☒ upgrade environment for differential expression analysis.
- ☒ evaluate the impact of confounder adjustment on a specific use case: compare miRNA expression between T2D and NGT at baseline.

```
devtools::load_all("src/pkg/dbs.data")
```

```
i Loading dbs.data
```

```
devtools::load_all("src/pkg/latarnia.utils")
```

```
i Loading latarnia.utils
```

```
Loading required package: grid
```

```
Loading required package: shiny
```

```
knitr::opts_chunk$set(results = params$knitr$results)
library(assertthat)
source("R/ngs.R")
```

4.0.1 Data preparation

```r
adsl <- dbs.data::adsl
advs <- dbs.data::advs
adlb <- dbs.data::adlb

#' Subjid and Visit to Sample
#'
subjvis_to_spl <- function(df) paste0(df$subjid, "v", df$avisitn)

ads <- adlb |>
  subset(
    paramcd %in% c(
      "CHOL", "HBA1C", "HDL", "HOMAB", "HOMAIR", "LDL", "MATSUDA", "TRIG"
    )
  ) %>%
  rbind(advs) |>
  subset(select = -c(ct, dtype, param, base, basetype, chg, pchg)) |>
  tidyr::pivot_wider(names_from = "paramcd", values_from = "aval") |>
  (\(df) merge(adsl, df, by = "subjid"))() |>
  (\(df) S4Vectors::DataFrame(df, row.names = subjvis_to_spl(df)))() |>
  (\(df) {
    assertthat::assert_that(all(table(subjvis_to_spl(df)) == 1))
    df
  })()

ads
```

```r
rna <- list(# There will be mi-RNA data.
  mrna = dbs.data::mrna_raw,
  premirna = dbs.data::premirna_raw,
  mirna = dbs.data::mirna_raw
)

rna[c("premirna", "mirna")] <- lapply(
  X = rna[c("premirna", "mirna")],
  FUN = format_mirna
)

# Rows represent genes.
rna <- lapply(X = rna, FUN = function(x) y <- x[rowSums(x) > 0, ])
rna <- lapply(X = rna, as.matrix)
assertthat::assert_that(all(colnames(rna$premirna) == colnames(rna$mirna)))
rna$allmirna <- rbind(rna$premirna, rna$mirna)
library(testthat)
test_that("rna features discriminated in noexpr, expr", {
  lapply(
    X = rna,
    FUN = function(x) expect_true(all(rowSums(x) > 0))
  )
})
```

```r
library(MultiAssayExperiment)
```

```
Loading required package: SummarizedExperiment


Loading required package: MatrixGenerics


Loading required package: matrixStats



Attaching package: 'MatrixGenerics'


The following objects are masked from 'package:matrixStats':

    colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
    colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
    colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
    colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
    colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
    colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
    colWeightedMeans, colWeightedMedians, colWeightedSds,
    colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
    rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
    rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
    rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
    rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
    rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
    rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
    rowWeightedSds, rowWeightedVars
```

```
Loading required package: GenomicRanges

Loading required package: stats4

Loading required package: BiocGenerics


Attaching package: 'BiocGenerics'


The following objects are masked from 'package:stats':

    IQR, mad, sd, var, xtabs


The following objects are masked from 'package:base':

    anyDuplicated, append, as.data.frame, basename, cbind, colnames,
    dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
    grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
    order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
    rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
    union, unique, unsplit, which.max, which.min


Loading required package: S4Vectors


Attaching package: 'S4Vectors'


The following objects are masked from 'package:base':

    expand.grid, I, unname


Loading required package: IRanges

Loading required package: GenomeInfoDb

Loading required package: Biobase
```

```
Welcome to Bioconductor

    Vignettes contain introductory material; view with
    'browseVignettes()'. To cite Bioconductor, see
    'citation("Biobase")', and for packages 'citation("pkgname")'.


Attaching package: 'Biobase'

The following object is masked from 'package:MatrixGenerics':

    rowMedians

The following objects are masked from 'package:matrixStats':

    anyMissing, rowMedians
```

```r
#' (Sample-)Map Arrays
#'
#' Use the colnames of `x` to deduce the `primary` and `colnames`.
#' This is used to generate the sample mapping between colData and Experiments.
#'
#' @param x (`dataframe`).
#'
#' @note In our case, primary and colnames are equivalent, colnames could
#' be different from primary names when a biological sample has different
#' names in the biological assays (e.g. machine constraint, technical
#' repetitions).
#'
#' @seealso [MultiAssayExperiment::listToMap()]
#' @examples
#' \dontrun{
#' lapply(rna, map_arrays)
#' MultiAssayExperiment::listToMap(lapply(rna, map_arrays))
#' }
#'
map_arrays <- function(x) {
  y <- data.frame(colname = colnames(x))
  y$primary <- y$colname
  y
}
```

```r
besd_mae <- MultiAssayExperiment(
  experiments = ExperimentList(rna),
  colData = ads,
  sampleMap = listToMap(lapply(rna, map_arrays))
)

besd_mae
```

4.1 DE: Baseline, all micro RNA, no confounding factor (`dds_1`)

```r
ctrl <- yaml::read_yaml("_prog.yml")$rna

ngs_assay <- "allmirna"

filter_for_depth <- function(mae, assay, depth_threshold) {
  mae[, colSums(mae[[ngs_assay]]) > depth_threshold, ]
}

filter_for_visit <- function(mae, visit) {
  mae[, colData(mae)$avisit == visit, ]
}

filter_for_low_expr <- function(mae, assay, cpm_threshold, frac_cols = 1 / 2) {
  # Genes expressed at least cpm_threshold in frac_cols columns
  mae[
    rowSums(cpm(mae[[assay]]) > cpm_threshold) >
      ncol(mae[[assay]]) * frac_cols,
    ,
  ]
}

ads <- besd_mae |>
  (\(mae) mae[ , , ngs_assay])() |>
  filter_for_depth("allmirna", ctrl$depth_threshold[[ngs_assay]]) |>
  filter_for_visit("Baseline") |>
  filter_for_low_expr("allmirna", ctrl$cpm_threshold[[ngs_assay]])
```

```
Warning: 'experiments' dropped; see 'metadata'
```

```
harmonizing input:
  removing 282 sampleMap rows not in names(experiments)
```

```r
ads

dds_1 <- DESeq2::DESeqDataSetFromMatrix(
  countData = ads[[ngs_assay]],
  colData = colData(ads),
  design = stats::formula(~ diabcd)
)

dds_1_res <- DESeq2::DESeq(
  object = dds_1,
  quiet = FALSE, # default: FALSE
  minReplicatesForReplace = 7, # default: 7
  useT = FALSE, # default: FALSE
  minmu = 0.5, # default: 0.5
  parallel = TRUE,
  BPPARAM = BiocParallel::bpparam()
)
```

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates: 2 workers
```

```
mean-dispersion relationship
```

```
-- note: fitType='parametric', but the dispersion trend was not well captured by the
   function: y = a/x + b, and a local regression fit was automatically substituted.
   specify fitType='local' or 'mean' to avoid this message next time.
```

```
final dispersion estimates, fitting model and testing: 2 workers
```

```
-- replacing outliers and refitting for 13 genes
-- DESeq argument 'minReplicatesForReplace' = 7
-- original counts are preserved in counts(dds)
```

```
estimating dispersions
```

```
fitting model and testing
```

```r
dds_1_de <- DESeq2::results(
  dds_1_res,
  contrast = c("diabcd", test = "T2D", ref = "NGT"),
  pAdjustMethod = ctrl$adj_meth
) |>
  (\(df) {
    df$feature <- rownames(df)
    df
  }) () |>
  within(log_padj <- -1 * log10(padj))

library(ggplot2)
dds_1_gg <-
  dds_1_de |> as.data.frame() |>
  ggplot(mapping = aes(log2FoldChange, log_padj, fill = log10(..count..))) +
  geom_hline(yintercept = -1 * log10(c(0.05, 0.001)), lty = 2, lwd = .5) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  annotate(
    geom  = "label", x = -Inf, y = -1 * log10(0.05),
    label = "p = 0.05",
    fill = "white", hjust = "left", size = 2, alpha = 1
  ) +
  annotate(
    geom  = "label", x = -Inf, y = -1 * log10(0.001),
    label = "p = 0.001",
    fill = "white", hjust = "left", size = 2, alpha = 1
  ) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex() +
  scale_fill_gradient(low = "black", high = "gray90") +
  theme_minimal() +
  theme(legend.position = "bottom", asp = 2 / 3)
```

```r
dds_1_gg
```

4.2 DE: Baseline, all micro RNA, accounting for Age, BMI, DCAL, Trainn (dds_2)

```r
ctrl <- yaml::read_yaml("_prog.yml")$rna

ngs_assay <- "allmirna"

filter_for_depth <- function(mae, assay, depth_threshold) {
  mae[, colSums(mae[[ngs_assay]]) > depth_threshold, ]
}

filter_for_visit <- function(mae, visit) {
  mae[, colData(mae)$avisit == visit, ]
}

filter_for_low_expr <- function(mae, assay, cpm_threshold, frac_cols = 1 / 2) {
  # Genes expressed at least cpm_threshold in frac_cols columns
  mae[
    rowSums(cpm(mae[[assay]]) > cpm_threshold) >
      ncol(mae[[assay]]) * frac_cols,
    ,
  ]
}

scale_confounder <- function(mae, confounder) {
  for (i in seq_along(confounder)) {
    cfd <- confounder[i]
    colData(mae)[cfd] <- scale(colData(mae)[cfd])
  }
  mae
}
ads <- besd_mae |>
  (\(mae) mae[ , , ngs_assay])() |>
  filter_for_depth("allmirna", ctrl$depth_threshold[[ngs_assay]]) |>
  filter_for_visit("Baseline") |>
  filter_for_low_expr("allmirna", ctrl$cpm_threshold[[ngs_assay]]) |>
  scale_confounder(confounder = c("age", "trainn", "BMI", "DCAL"))
```

```
Warning: 'experiments' dropped; see 'metadata'
```

```
harmonizing input:
  removing 282 sampleMap rows not in names(experiments)
```

```r
ads

dds_2 <- DESeq2::DESeqDataSetFromMatrix(
  countData = ads[[ngs_assay]],
  colData = colData(ads),
  design = stats::formula(~ age + BMI + trainn + DCAL + diabcd)
)

dds_2_res <- DESeq2::DESeq(
  object = dds_2,
  quiet = FALSE, # default: FALSE
  minReplicatesForReplace = 7, # default: 7
  useT = FALSE, # default: FALSE
  minmu = 0.5, # default: 0.5
  parallel = TRUE,
  BPPARAM = BiocParallel::bpparam()
)
```

```
estimating size factors

estimating dispersions

gene-wise dispersion estimates: 2 workers

mean-dispersion relationship

-- note: fitType='parametric', but the dispersion trend was not well captured by the
   function: y = a/x + b, and a local regression fit was automatically substituted.
   specify fitType='local' or 'mean' to avoid this message next time.

final dispersion estimates, fitting model and testing: 2 workers
```

```r
dds_2_de <- DESeq2::results(
  dds_2_res,
  contrast = c("diabcd", test = "T2D", ref = "NGT"),
  pAdjustMethod = ctrl$adj_meth
) |>
```

```
  (\(df) {
    df$feature <- rownames(df)
    df
  }) () |>
  within(log_padj <- -1 * log10(padj))

library(ggplot2)
dds_2_gg <-
  dds_2_de |> as.data.frame() |>
  ggplot(mapping = aes(log2FoldChange, log_padj, fill = log10(..count..))) +
  geom_hline(yintercept = -1 * log10(c(0.05, 0.001)), lty = 2, lwd = .5) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  annotate(
    geom  = "label", x = -Inf, y = -1 * log10(0.05),
    label = "p = 0.05",
    fill = "white", hjust = "left", size = 2, alpha = 1
  ) +
  annotate(
    geom  = "label", x = -Inf, y = -1 * log10(0.001),
    label = "p = 0.001",
    fill = "white", hjust = "left", size = 2, alpha = 1
  ) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex() +
  scale_fill_gradient(low = "black", high = "gray90") +
  theme_minimal() +
  theme(legend.position = "bottom", asp = 2 / 3)
```

```
dds_2_gg
```

4.3 Comparison with/without confounding factors

```
theme_fun <- function(...) {
  theme_minimal() +
    theme(
      title = element_text(size = 9),
      text = element_text(size = 9)
    ) +
```

```r
    theme(...)
}

gg_1_2 <- rbind(
  within(as.data.frame(dds_1_de), facet <- "No confounding factors"),
  within(as.data.frame(dds_2_de), facet <- "~Age + BMI + DCAL + Train")
) |>
  ggplot(mapping = aes(log2FoldChange, log_padj, fill = log10(..count..))) +
  geom_hline(yintercept = -1 * log10(c(0.05, 0.001)), lty = 2, lwd = .5) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  annotate(
    geom  = "label", x = -Inf, y = -1 * log10(0.05),
    label = "p = 0.05",
    fill = "white", hjust = "left", size = 2, alpha = 1
  ) +
  annotate(
    geom  = "label", x = -Inf, y = -1 * log10(0.001),
    label = "p = 0.001",
    fill = "white", hjust = "left", size = 2, alpha = 1
  ) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex() +
  scale_fill_gradient(low = "black", high = "gray90") +
  facet_wrap(facet ~ ., ncol = 2) +
  theme_fun(
    legend.position = "bottom"
  ) +
  theme(
    legend.key.width = unit(5, "lines"),
    legend.key.height = unit(.8, "lines")
  )

res <- merge(
  as.data.frame(dds_1_de),
  as.data.frame(dds_2_de),
  by = "feature",
  all = TRUE,
  suffixes = c(".asis", ".cfd")
)
```

```
fun_label <- function(df,
                      x = "log2FoldChange.asis",
                      y = "log2FoldChange.cfd") {
  cor_fun <- function(meth = "pearson") {
    round(cor(df[[x]], df[[y]], method = meth), 2)
  }
  paste0(
    "atop(",
    "r == ", cor_fun(), ",",
    "rho == ", cor_fun("spearman"),
    ")"
  )
}

lim <- range(unlist(res[c("log2FoldChange.asis", "log2FoldChange.cfd")]))
gg_cor_lfc <- ggplot(res, aes(log2FoldChange.asis, log2FoldChange.cfd)) +
  geom_hex() +
  scale_fill_viridis_c(option = "F", begin = .1, end = .9) +
  geom_abline(slope = 1, intercept = 0, col = "red") +
  annotate(
    "label", x = -Inf, y = Inf, hjust = 0, vjust = 1,
    label = fun_label(res),
    parse = TRUE,
    family = "mono",
    size = 3
  )+
  coord_cartesian(xlim = lim, ylim = lim) +
  labs(
    title = "Log Fold Change (LFC)",
    subtitle = "With / Without Adjustment for Confounding Factors",
    x = "No Adjustment",
    y = "Adjusted for Confounding Factors"
  ) +
  theme_fun(asp = 1)

lim <- range(unlist(res[c("log_padj.asis", "log_padj.cfd")]))
gg_cor_pval <- ggplot(res, aes(log_padj.asis, log_padj.cfd)) +
  geom_hex() +
  scale_fill_viridis_c(option = "D", begin = .1, end = .9) +
  geom_abline(slope = 1, intercept = 0, col = "green2") +
  annotate(
```

```r
    "label", x = -Inf, y = Inf, hjust = 0, vjust = 1,
    label = fun_label(res, "log_padj.asis", "log_padj.cfd"),
    parse = TRUE,
    family = "mono",
    size = 3
  )+
  coord_cartesian(xlim = lim, ylim = lim, clip = "off") +
  labs(
    title = expression("Significance: "*-1 %.% log10(padj)),
    subtitle = "With / Without Adjustment for Confounding Factors",
    x = "No Adjustment",
    y = "Adjusted for Confounding Factors"
  ) +
  theme_fun(asp = 1)

library(cowplot)
p <- plot_grid(
  plot_grid(gg_1_2) + theme(plot.background = element_rect(color = "black")),
  plot_grid(
    plot_grid(gg_cor_lfc) +
      theme(plot.background = element_rect(color = "black")),
    plot_grid(gg_cor_pval) +
      theme(plot.background = element_rect(color = "black")),
    labels = c("B", "C")
  ),
  ncol = 1, rel_heights = c(3, 2),
  labels = c("A", NA)
)


p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 6.1",
      strwrap(
        "Volcano plot - Level and significance of Differential Expression among
        all miRNA at Baseline between T2D and NGT", width = 80
      ),
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_note(c(
    "A: Left panel accounts for Age, BMI, DCAL (diet) and number of trainings in
    the estimation and test of the differential expression of every gene; it
    may present marginal differences with the version presented 2 years ago
    likely due to slight variations in stochastic elements (e.g. missing
    data imputation).",
    "A: Right panel discards any confounding factors.",
    "B, C: Scatter plots comparing
    the Log Fold Change estimations (B)/
    the significance (C, the higher the more significant)
    with (y axis) without (x axis) adjustment for confounding factors with
    annotation corresponding to the Pearson's correlation (r) and
    Spearman's rank correlation (rho).",
    "Hexbin representation: the intensity of each hexagonal bin accounts for
    the number of genes found in the area it covers."
  )) |>
  add_figure(p, height = .9) |>
  add_footer(
    "Program t2d_06_rna / Env ayup_dbs:v0.1.0-alpha",
    params$version
  )
```

```
Warning: Removed 1 rows containing missing values (geom_text).
```

```
export_as(
  p,
  file = file.path(params$paths$grh, "fig_06_01.pdf"),
  file_graph_alone = file.path(params$paths$grh, "fig_06_01_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_06_01.pdf
```

```
[log] output saved as: ../tlg/graph/fig_06_01_af.pdf (annot. free)
```

```
show_slate(p)
```

4.4 Session Informations

```
sessioninfo::session_info()
```

5 mRNA / micro RNA

Program 07

Francois Collin
2022-07-01

```
cfg_prog <- yaml::read_yaml("_prog.yml")
```

The variation in messenger RNA (mRNA) and micro-RNA (miRNA) abundance measured by RNA-Seq associated either with the dysglycemia and/or the 3-month exercise intervention were investigated by a Differential Expression analysis (DE). At baseline, the DE model included only the effect of the dysglycemia-constrast group, consistently with the Anova model applied for demographics and baseline anthropometrics. To do so, genetic features which were never found expressed were discarded. Then RNA-Seq libraries were included if reaching a read depth threshold (total number of reads per sample) fixed by visual examination of the association between sample gene diversity and sample depth. For the remaining libraries, genetic features were included if the count per million reads (CPM) was greater than 2 in at least 50% of the samples. The DE analysis implementation proposed by Love, Huber, and Anders (2014) with the R package `DESeq2` was used to fit the models; it is based on a negative binomial distribution accounting for read variability, dispersion corrected after trends seen across all samples and genes (Dündar, Skrabanek, and Zumbo 2018). In addition, the number of tested genes being high, raw p.values were therefore adjusted according to Benjamini and Hochberg's method also called False-Discovery Rate (FDR).

The analysis was realized with R (v4.2.0, R Core Team 2022). Note the use of the additional packages downloaded from the RStudio package manager repository (freeze date 2022-05-04, repos):

- `DESeq2` for differential expression analysis and variance stabilizing transformation (version 1.36.0, Love, Huber, and Anders 2014)
- `ggplot2` for graphics (version 3.3.6, Wickham 2016)
- `MultiAssayExperiment` for management of multi-omics experiment objects (version 1.22.0, Ramos et al. 2017).

Target:

☒ miRNA DE at baseline without confounding factors.

```r
devtools::load_all("src/pkg/dbs.data")
```

```
i Loading dbs.data
```

```r
devtools::load_all("src/pkg/latarnia.utils")
```

```
i Loading latarnia.utils
```

```
Loading required package: grid
```

```
Loading required package: shiny
```

```r
knitr::opts_chunk$set(results = cfg_prog$knitr$results)
library(assertthat)
library(ggplot2)
source("R/ngs.R")
source("R/inches.R")
```

5.0.1 Materials and Methods

5.0.1.1 Helper functions

```r
# Help for data pre-processing
filter_for_depth <- function(mae, assay, depth_threshold) {
  mae[, colSums(mae[[ngs_assay]]) > depth_threshold, ]
}

filter_for_visit <- function(mae, visit) {
  mae[, colData(mae)$avisit == visit, ]
}

filter_for_low_expr <- function(mae, assay, cpm_threshold,
                                frac_cols = cfg_prog$rna$cpm_threshold$fraccol
) {
  # Genes expressed at least cpm_threshold in frac_cols columns
  mae[
    rowSums(cpm(mae[[assay]]) > cpm_threshold) >
      ncol(mae[[assay]]) * frac_cols,
    ,
  ]
}

# Differential expression helper functions:
de_by_ctrs <- function(df,
                       ctrs,
                       adj_meth = ctrl$adj_meth) {
  lapply(
    ctrs,
    fit = df,
    adj_meth = adj_meth,
    FUN = function(x, fit, adj_meth) {
      y <- DESeq2::results(fit, contrast = x, pAdjustMethod = adj_meth)
      y$feature <- rownames(y)
      y$log_padj <- -1 * log10(y$padj)
      y$ctrs <- paste(x["test"], "vs", x["ref"])
      as.data.frame(y)
    }
  )
}
```

5.0.1.2 Data preparation

```
adsl <- dbs.data::adsl
advs <- dbs.data::advs
adlb <- dbs.data::adlb

#' Subjid and Visit to Sample
#'
subjvis_to_spl <- function(df) paste0(df$subjid, "v", df$avisitn)

ads <- adlb |>
  rbind(advs) |>
  subset(select = -c(ct, dtype, param, base, basetype, chg, pchg)) |>
  tidyr::pivot_wider(names_from = "paramcd", values_from = "aval") |>
  merge(adsl, y = _, by = "subjid") |>
```

```r
  (\(df) S4Vectors::DataFrame(df, row.names = subjvis_to_spl(df)))() |>
  subset(select = c(subjid, diabcd, diab, avisitn, avisit)) |>
  (\(df) {
    assertthat::assert_that(all(table(subjvis_to_spl(df)) == 1))
    df
  })()

ads
```

```r
rna <- list(
  mrna = dbs.data::mrna_raw,
  premirna = dbs.data::premirna_raw,
  mirna = dbs.data::mirna_raw
)

rna[c("premirna", "mirna")] <- lapply(
  X = rna[c("premirna", "mirna")],
  FUN = format_mirna
)

# Rows represent genes
rna <- lapply(X = rna, FUN = function(x) y <- x[rowSums(x) > 0, ])
rna <- lapply(X = rna, as.matrix)
assertthat::assert_that(all(colnames(rna$premirna) == colnames(rna$mirna)))
rna$allmirna <- rbind(rna$premirna, rna$mirna)

library(testthat)
test_that("rna features discriminated in noexpr, expr", {
  lapply(X = rna, FUN = \(x) expect_true(all(rowSums(x) > 0)))
})
```

```r
library(MultiAssayExperiment)
```

```
Loading required package: SummarizedExperiment

Loading required package: MatrixGenerics

Loading required package: matrixStats


Attaching package: 'MatrixGenerics'
```

```
The following objects are masked from 'package:matrixStats':

    colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
    colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
    colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
    colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
    colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
    colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
    colWeightedMeans, colWeightedMedians, colWeightedSds,
    colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
    rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
    rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
    rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
    rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
    rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
    rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
    rowWeightedSds, rowWeightedVars

Loading required package: GenomicRanges

Loading required package: stats4

Loading required package: BiocGenerics


Attaching package: 'BiocGenerics'

The following objects are masked from 'package:stats':

    IQR, mad, sd, var, xtabs

The following objects are masked from 'package:base':

    anyDuplicated, append, as.data.frame, basename, cbind, colnames,
    dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
    grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
    order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
    rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
    union, unique, unsplit, which.max, which.min

Loading required package: S4Vectors
```

```
Attaching package: 'S4Vectors'


The following objects are masked from 'package:base':

    expand.grid, I, unname


Loading required package: IRanges


Loading required package: GenomeInfoDb


Loading required package: Biobase


Welcome to Bioconductor

    Vignettes contain introductory material; view with
    'browseVignettes()'. To cite Bioconductor, see
    'citation("Biobase")', and for packages 'citation("pkgname")'.



Attaching package: 'Biobase'


The following object is masked from 'package:MatrixGenerics':

    rowMedians


The following objects are masked from 'package:matrixStats':

    anyMissing, rowMedians
```

```
#' (Sample-)Map Arrays
#'
#' Use the column names of `x` to deduce the `primary` and `colnames`.
#' This is used to generate the sample mapping between colData and Experiments.
#'
#' @param x (`dataframe`).
#'
#' @note In our case, primary and colnames are equivalent, colnames could
#' be different from primary names when a biological sample has different
#' names in the biological assays (e.g. machine constraint, technical
#' repetitions).
#'
#' @seealso [MultiAssayExperiment::listToMap()]
#' @examples
#' \dontrun{
#'   lapply(rna, map_arrays)
#'   MultiAssayExperiment::listToMap(lapply(rna, map_arrays))
#' }
#'
map_arrays <- function(x) {
  y <- data.frame(colname = colnames(x))
  y$primary <- y$colname
  y
}

besd_mae <- MultiAssayExperiment(
  experiments = ExperimentList(rna),
  colData = ads,
  sampleMap = listToMap(lapply(rna, map_arrays))
)

besd_mae
```

5.1 Sample depth threshold determination

A scatter plot representing sample gene diversity in relationship with sample depth evidenced a threshold for miRNA reads at 5e+06 reads for which approximately a million read increase was associated with more than a 2-fold increase in gene diversity. The depth threshold for mRNA was fixed at , although with a less important but still substantial drop in gene variety below that threshold.

5.1.0.1 miRNA

5.1.0.2 Static figure

```
assay <- "allmirna"
```

```r
vline <- cfg_prog$rna$depth_threshold[[assay]]
dtaplot <- assays(besd_mae)[[assay]]
dtaplot <- data.frame(
  n_gene = colSums(dtaplot > 0),
  depth  = colSums(dtaplot)
)

dtaplot <- cbind(
  dtaplot,
  data.frame(colData(besd_mae)[rownames(dtaplot),])
)

gg <- ggplot(
  data = dtaplot,
  mapping = aes(x = depth, y = n_gene, colour = diab)
) +
  geom_point() +
  geom_vline(xintercept = vline) +
  xlab("Sample depth") +
  ylab("Gene diversity (number of genes with reads)") +
  coord_cartesian(ylim = c(0, NA)) +
  theme_minimal() +
  theme(legend.position = "top")
```

5.1.0.3 Interactive figure

```r
plotly::ggplotly(gg)
```

5.1.0.4 mRNA

5.1.0.5 Static figure

```r
assay <- "mrna"
```

```r
vline <- cfg_prog$rna$depth_threshold[[assay]]
dtaplot <- assays(besd_mae)[[assay]]
dtaplot <- data.frame(
  n_gene = colSums(dtaplot > 0),
  depth   = colSums(dtaplot)
)

dtaplot <- cbind(
  dtaplot,
  data.frame(colData(besd_mae)[rownames(dtaplot),])
)

gg <- ggplot(
  data = dtaplot,
  mapping = aes(x = depth, y = n_gene, colour = diab)
) +
  geom_point() +
  geom_vline(xintercept = vline) +
  xlab("Sample depth") +
  ylab("Gene diversity (number of genes with reads)") +
  coord_cartesian(ylim = c(0, NA)) +
  theme_minimal() +
  theme(legend.position = "top")
```

5.1.0.6 Interactive figure

```r
plotly::ggplotly(gg)
```

5.2 `dds_1` - DE: Baseline, all micro RNA

```r
ctrl <- cfg_prog$rna
ngs_assay <- "allmirna"

dds_1_fit <- besd_mae |>
  (\(mae) mae[ , , ngs_assay])() |>
  filter_for_depth(ngs_assay, ctrl$depth_threshold[[ngs_assay]]) |>
  filter_for_visit("Baseline") |>
  filter_for_low_expr(ngs_assay, ctrl$cpm_threshold[[ngs_assay]]) |>
  (\(mae) DESeq2::DESeqDataSetFromMatrix(
    countData = mae[[ngs_assay]],
    colData = colData(x = mae),
    design = stats::formula(~ diabcd)
  ) )() |>
  DESeq2::DESeq(
    parallel = TRUE,
    BPPARAM = BiocParallel::bpparam(),
    fitType = "local"
  )
```

```
Warning: 'experiments' dropped; see 'metadata'

harmonizing input:
  removing 282 sampleMap rows not in names(experiments)

estimating size factors

estimating dispersions

gene-wise dispersion estimates: 2 workers

mean-dispersion relationship

final dispersion estimates, fitting model and testing: 2 workers

-- replacing outliers and refitting for 13 genes
-- DESeq argument 'minReplicatesForReplace' = 7
-- original counts are preserved in counts(dds)

estimating dispersions

fitting model and testing
```

```
  dds_1_est <-
    dds_1_fit |>
    de_by_ctrs(
      ctrs = list(
        c("diabcd", test = "T2D", ref = "NGT"),
        c("diabcd", test = "T2D", ref = "IGT"),
        c("diabcd", test = "IGT", ref = "NGT")
      )
    ) |>
    Reduce(rbind, x = _)
```

5.2.1 Fig 07 01

```
pval <- c(0.05, 0.001)
log_adj <- pretty(dds_1_est$log_padj)
brk <- setNames(
  c(log_adj, -1 * log10(pval)),
  c(log_adj, paste0("padj=", pval))
)
dds_1_gg <-
  dds_1_est |>
  ggplot(aes(log2FoldChange, log_padj, fill = log10(..count..))) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex() +
  scale_fill_gradient(low = "black", high = "gray90") +
  scale_y_continuous(breaks = brk) +
  facet_grid(. ~ ctrs) +
  theme_minimal() +
  theme(
    text = element_text(size = 7),
    title =  element_text(size = 7),
    legend.position = "bottom",
    legend.key.height = unit(.5, "lines"),
    legend.key.width = unit(3, "lines"),
    legend.text.align = 0,
    panel.grid.minor = element_blank()
  )
```

```
p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 7.1",
      strwrap(
        "Volcano plot - Response Size and Significance of Differential
        Expression among all miRNA at Baseline", width = 80
      ),
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_figure(dds_1_gg, height = inches(3), width = inches(6)) |>
  add_footer(
    c(
      "Program t2d_07_mir / Env ayup_dbs:v0.1.0-alpha",
      format(Sys.time(), format = "%Y-%m-%d %H:%M (%Z)")
    ),
    cfg_prog$version
  )
```

```
export_as(
  p,
  file = file.path(cfg_prog$paths$grh, "fig_07_01.pdf"),
  file_graph_alone = file.path(cfg_prog$paths$grh, "fig_07_01_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_07_01.pdf
```

```
[log] output saved as: ../tlg/graph/fig_07_01_af.pdf (annot. free)
```

```
show_slate(p)
```

5.2.2 Fig 07 02

```
pval <- c(0.05, 0.001)
log_adj <- pretty(dds_1_est$log_padj)
brk <- setNames(
  c(log_adj, -1 * log10(pval)),
  c(log_adj, paste0("padj=", pval))
)

dds_1_est$label <- gsub(
  "hsa-(mir|miR|let)-",
  dds_1_est$feature,
  replacement = ""
)
dds_1_est$type <- substr(dds_1_est$feature, start = 5, stop = 7)
head(dds_1_est)

dds_1_gg <- ggplot(
  dds_1_est,
  mapping = aes(x = log2FoldChange, y = log_padj, col = type)
) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex(mapping = aes(fill = log10(..count..)), color = "transparent") +
  geom_point(
    data = subset(dds_1_est, padj < 0.001 & abs(log2FoldChange) > 1),
    size = 1,
    shape = 3
  ) +
  geom_text(
    data = subset(dds_1_est, padj < 0.001 & abs(log2FoldChange) > 1),
    aes(label = label),
    vjust = -0.5,
    size = 2,
    show.legend = FALSE
  ) +
  scale_fill_gradient(low = "black", high = "gray90") +
  scale_color_manual(values = c("#FF5003", "#003F7D")) +
  scale_y_continuous(breaks = brk) +
  facet_wrap(. ~ ctrs, ncol = 3) +
  coord_cartesian(clip = "off") +
  theme_minimal() +
  theme(
    text = element_text(size = 7),
    title =  element_text(size = 7),
    legend.position = "bottom",
    legend.key.height = unit(.5, "lines"),
    legend.key.width = unit(2, "lines"),
    legend.text.align = 0,
    panel.grid.minor = element_blank()
  )
```

```
p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 7.2",
      strwrap(
        "Volcano plot - Response Size and Significance of Differential
        Expression among all miRNA at Baseline", width = 80
      ),
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_figure(dds_1_gg, height = inches(3), width = inches(6)) |>
  add_footer(
    c(
      "Program t2d_07_mir / Env ayup_dbs:v0.1.0-alpha",
      format(Sys.time(), format = "%Y-%m-%d %H:%M (%Z)")
    ),
    cfg_prog$version
  )

export_as(
  p,
  file = file.path(cfg_prog$paths$grh, "fig_07_02.pdf"),
  file_graph_alone = file.path(cfg_prog$paths$grh, "fig_07_02_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_07_02.pdf
```

```
[log] output saved as: ../tlg/graph/fig_07_02_af.pdf (annot. free)
```

```
show_slate(p)
```

5.3 dds_2 - DE: Baseline, mRNA

```
ctrl <- cfg_prog$rna
ngs_assay <- "mrna"

dds_2_fit <- besd_mae |>
  (\(mae) mae[ , , ngs_assay])() |>
  filter_for_depth(ngs_assay, ctrl$depth_threshold[[ngs_assay]]) |>
  filter_for_visit("Baseline") |>
  filter_for_low_expr(ngs_assay, ctrl$cpm_threshold[[ngs_assay]]) |>
  (\(mae) DESeq2::DESeqDataSetFromMatrix(
    countData = mae[[ngs_assay]],
    colData = colData(x = mae),
    design = stats::formula(~ diabcd)
  ) )() |>
  DESeq2::DESeq(
    parallel = TRUE,
    BPPARAM = BiocParallel::bpparam()
  )
```

```
Warning: 'experiments' dropped; see 'metadata'
```

```
harmonizing input:
  removing 282 sampleMap rows not in names(experiments)
```

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates: 2 workers
```

```
mean-dispersion relationship
```

```
final dispersion estimates, fitting model and testing: 2 workers
```

```
-- replacing outliers and refitting for 13 genes
-- DESeq argument 'minReplicatesForReplace' = 7
-- original counts are preserved in counts(dds)
```

```
estimating dispersions

fitting model and testing
```

```r
dds_2_est <-
  dds_2_fit |>
  de_by_ctrs(
    ctrs = list(
      c("diabcd", test = "T2D", ref = "NGT"),
      c("diabcd", test = "T2D", ref = "IGT"),
      c("diabcd", test = "IGT", ref = "NGT")
    )
  ) |>
  Reduce(rbind, x = _)
```

5.3.1 Fig 07 03

```r
pval <- c(0.05, 0.001)
log_adj <- pretty(dds_2_est$log_padj)
brk <- setNames(
  c(log_adj, -1 * log10(pval)),
  c(log_adj, paste0("padj=", pval))
)

dds_2_gg <-
  dds_2_est |>
  ggplot(aes(log2FoldChange, log_padj, fill = log10(..count..))) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex() +
  scale_fill_gradient(low = "black", high = "gray90") +
  scale_y_continuous(breaks = brk) +
  facet_grid(. ~ ctrs) +
  theme_minimal() +
  theme(
    text = element_text(size = 7),
    title =  element_text(size = 7),
    legend.position = "bottom",
    legend.key.height = unit(.5, "lines"),
    legend.key.width = unit(3, "lines"),
    legend.text.align = 0,
    panel.grid.minor = element_blank()
  )
```

```r
p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 7.3",
      strwrap(
        "Volcano plot - Response Size and Significance of Differential
        Expression among all miRNA at Baseline", width = 80
      ),
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_figure(dds_2_gg, height = inches(3), width = inches(6)) |>
  add_footer(
    c(
      "Program t2d_07_mir / Env ayup_dbs:v0.1.0-alpha",
      format(Sys.time(), format = "%Y-%m-%d %H:%M (%Z)")
    ),
    cfg_prog$version
  )
```

```r
export_as(
  p,
  file = file.path(cfg_prog$paths$grh, "fig_07_03.pdf"),
  file_graph_alone = file.path(cfg_prog$paths$grh, "fig_07_03_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_07_03.pdf
```

```
[log] output saved as: ../tlg/graph/fig_07_03_af.pdf (annot. free)
```

```r
show_slate(p)
```

5.3.2 Fig 07 04

```r
dds_2_est$label <- gsub("ENSG0+", "", dds_2_est$feature)
head(dds_2_est)

pval <- c(0.05, 0.001)
log_adj <- pretty(dds_2_est$log_padj)
brk <- setNames(
  c(log_adj, -1 * log10(pval)),
  c(log_adj, paste0("padj=", pval))
)

dds_2_gg <- ggplot(dds_2_est, mapping = aes(x = log2FoldChange, y = log_padj)) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex(mapping = aes(fill = log10(..count..)), color = "transparent") +
  geom_point(
    data = subset(dds_2_est, padj < 0.001 & abs(log2FoldChange) > 2),
    size = 1,
    shape = 3,
    color = "royalblue"
  ) +
  geom_text(
    data = subset(dds_2_est, padj < 0.001 & abs(log2FoldChange) > 2),
    aes(label = label),
    vjust = -0.5,
    size = 2,
    color = "royalblue",
    show.legend = FALSE
  ) +
  scale_fill_gradient(low = "black", high = "gray90") +
  scale_color_manual(values = c("#FF5003", "#003F7D")) +
  scale_y_continuous(breaks = brk) +
  facet_wrap(. ~ ctrs, ncol = 3) +
  coord_cartesian(clip = "off") +
  theme_minimal() +
  theme(
    text = element_text(size = 7),
    title =  element_text(size = 7),
    legend.position = "bottom",
    legend.key.height = unit(.5, "lines"),
    legend.key.width = unit(2, "lines"),
    legend.text.align = 0,
    panel.grid.minor = element_blank()
  )

p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 7.4",
      strwrap(
        "Volcano plot - Response Size and Significance of Differential
        Expression among all miRNA at Baseline", width = 80
      ),
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_figure(dds_2_gg, height = inches(3), width = inches(6)) |>
  add_footer(
    c(
      "Program t2d_07_mir / Env ayup_dbs:v0.1.0-alpha",
      format(Sys.time(), format = "%Y-%m-%d %H:%M (%Z)")
    ),
    cfg_prog$version
  )

export_as(
  p,
  file = file.path(cfg_prog$paths$grh, "fig_07_04.pdf"),
  file_graph_alone = file.path(cfg_prog$paths$grh, "fig_07_04_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_07_04.pdf

[log] output saved as: ../tlg/graph/fig_07_04_af.pdf (annot. free)
```

```r
  show_slate(p)
```

5.4 dds_3 - DE: Month 3, all micro RNA

```
ctrl <- cfg_prog$rna
ngs_assay <- "allmirna"

dds_3_fit <- besd_mae |>
  (\(mae) mae[ , , ngs_assay])() |>
  filter_for_depth(ngs_assay, ctrl$depth_threshold[[ngs_assay]]) |>
  filter_for_visit("Month 3") |>
  filter_for_low_expr(ngs_assay, ctrl$cpm_threshold[[ngs_assay]]) |>
  (\(mae) DESeq2::DESeqDataSetFromMatrix(
    countData = mae[[ngs_assay]],
    colData = colData(x = mae),
    design = stats::formula(~ diabcd)
  ) )() |>
  DESeq2::DESeq(
    parallel = TRUE,
    BPPARAM = BiocParallel::bpparam(),
    fitType = "local"
  )
```

```
Warning: 'experiments' dropped; see 'metadata'
```

```
harmonizing input:
  removing 282 sampleMap rows not in names(experiments)
```

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates: 2 workers
```

```
mean-dispersion relationship
```

```
final dispersion estimates, fitting model and testing: 2 workers
```

```
-- replacing outliers and refitting for 13 genes
-- DESeq argument 'minReplicatesForReplace' = 7
-- original counts are preserved in counts(dds)
```

```
estimating dispersions

fitting model and testing
```

```
dds_3_est <-
  dds_3_fit |>
  de_by_ctrs(
    ctrs = list(
      c("diabcd", test = "T2D", ref = "NGT"),
      c("diabcd", test = "T2D", ref = "IGT"),
      c("diabcd", test = "IGT", ref = "NGT")
    )
  ) |>
  Reduce(rbind, x = _)
```

5.4.1 Fig 07 05

```
pval <- c(0.05, 0.001)
log_adj <- pretty(dds_3_est$log_padj)
brk <- setNames(
  c(log_adj, -1 * log10(pval)),
  c(log_adj, paste0("padj=", pval))
)
dds_3_gg <-
  dds_3_est |>
  ggplot(aes(log2FoldChange, log_padj, fill = log10(..count..))) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex() +
  scale_fill_gradient(low = "black", high = "gray90") +
  scale_y_continuous(breaks = brk) +
  facet_grid(. ~ ctrs) +
  theme_minimal() +
  theme(
    text = element_text(size = 7),
    title =  element_text(size = 7),
    legend.position = "bottom",
    legend.key.height = unit(.5, "lines"),
    legend.key.width = unit(3, "lines"),
    legend.text.align = 0,
    panel.grid.minor = element_blank()
  )

p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 7.5",
      strwrap(
        "Volcano plot - Response Size and Significance of Differential
        Expression among all miRNA at Month 3", width = 80
      ),
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_figure(dds_3_gg, height = inches(3), width = inches(6)) |>
  add_footer(
    c(
      "Program t2d_07_mir / Env ayup_dbs:v0.1.0-alpha",
      format(Sys.time(), format = "%Y-%m-%d %H:%M (%Z)")
    ),
    cfg_prog$version
  )
```

```
Warning: Removed 350 rows containing non-finite values (stat_binhex).
```

```
export_as(
  p,
  file = file.path(cfg_prog$paths$grh, "fig_07_05.pdf"),
  file_graph_alone = file.path(cfg_prog$paths$grh, "fig_07_05_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_07_05.pdf
```

```
[log] output saved as: ../tlg/graph/fig_07_05_af.pdf (annot. free)
```

```
show_slate(p)
```

5.4.2 Fig 07 06

```
pval <- c(0.05, 0.001)
log_adj <- pretty(dds_3_est$log_padj)
brk <- setNames(
  c(log_adj, -1 * log10(pval)),
  c(log_adj, paste0("padj=", pval))
)

dds_3_est$label <- gsub(
  "hsa-(mir|miR|let)-",
  dds_3_est$feature,
  replacement = ""
)
dds_3_est$type <- substr(dds_3_est$feature, start = 5, stop = 7)
head(dds_3_est)

dds_3_gg <- ggplot(
  dds_3_est,
  mapping = aes(x = log2FoldChange, y = log_padj, col = type)
) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex(mapping = aes(fill = log10(..count..)), color = "transparent") +
  geom_point(
    data = subset(dds_3_est, padj < 0.001 & abs(log2FoldChange) > 1),
    size = 1,
    shape = 3
  ) +
  geom_text(
    data = subset(dds_3_est, padj < 0.001 & abs(log2FoldChange) > 1),
    aes(label = label),
    vjust = -0.5,
    size = 2,
    show.legend = FALSE
  ) +
  scale_fill_gradient(low = "black", high = "gray90") +
  scale_color_manual(values = c("#FF5003", "#003F7D")) +
  scale_y_continuous(breaks = brk) +
  facet_wrap(. ~ ctrs, ncol = 3) +
  coord_cartesian(clip = "off") +
  theme_minimal() +
  theme(
    text = element_text(size = 7),
    title =  element_text(size = 7),
    legend.position = "bottom",
    legend.key.height = unit(.5, "lines"),
    legend.key.width = unit(2, "lines"),
    legend.text.align = 0,
    panel.grid.minor = element_blank()
  )

p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 7.6",
      strwrap(
        "Volcano plot - Response Size and Significance of Differential
        Expression among all miRNA at Month 3", width = 80
      ),
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_figure(dds_3_gg, height = inches(3), width = inches(6)) |>
  add_footer(
    c(
      "Program t2d_07_mir / Env ayup_dbs:v0.1.0-alpha",
      format(Sys.time(), format = "%Y-%m-%d %H:%M (%Z)")
    ),
    cfg_prog$version
  )
```

```
Warning: Removed 350 rows containing non-finite values (stat_binhex).
```

```
  export_as(
    p,
    file = file.path(cfg_prog$paths$grh, "fig_07_06.pdf"),
    file_graph_alone = file.path(cfg_prog$paths$grh, "fig_07_06_af.pdf")
  )
```

```
[log] output saved as: ../tlg/graph/fig_07_06.pdf
```

```
[log] output saved as: ../tlg/graph/fig_07_06_af.pdf (annot. free)
```

```r
show_slate(p)
```

5.5 dds_4 - DE: Month 3, mRNA

```r
ctrl <- cfg_prog$rna
ngs_assay <- "mrna"

dds_4_fit <- besd_mae |>
  (\(mae) mae[ , , ngs_assay])() |>
  filter_for_depth(ngs_assay, ctrl$depth_threshold[[ngs_assay]]) |>
  filter_for_visit("Month 3") |>
  filter_for_low_expr(ngs_assay, ctrl$cpm_threshold[[ngs_assay]]) |>
  (\(mae) DESeq2::DESeqDataSetFromMatrix(
    countData = mae[[ngs_assay]],
    colData = colData(x = mae),
    design = stats::formula(~ diabcd)
  ) )() |>
  DESeq2::DESeq(
    parallel = TRUE,
    BPPARAM = BiocParallel::bpparam()
  )
```

```
Warning: 'experiments' dropped; see 'metadata'
```

```
harmonizing input:
  removing 282 sampleMap rows not in names(experiments)
```

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates: 2 workers
```

```
mean-dispersion relationship

final dispersion estimates, fitting model and testing: 2 workers

-- replacing outliers and refitting for 21 genes
-- DESeq argument 'minReplicatesForReplace' = 7
-- original counts are preserved in counts(dds)

estimating dispersions

fitting model and testing
```

```
dds_4_est <-
  dds_4_fit |>
  de_by_ctrs(
    ctrs = list(
      c("diabcd", test = "T2D", ref = "NGT"),
      c("diabcd", test = "T2D", ref = "IGT"),
      c("diabcd", test = "IGT", ref = "NGT")
    )
  ) |>
  Reduce(rbind, x = _)
```

5.5.1 Fig 07 07

```
pval <- c(0.05, 0.001)
log_adj <- pretty(dds_4_est$log_padj)
brk <- setNames(
  c(log_adj, -1 * log10(pval)),
  c(log_adj, paste0("padj=", pval))
)

dds_4_gg <-
  dds_4_est |>
  ggplot(aes(log2FoldChange, log_padj, fill = log10(..count..))) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex() +
  scale_fill_gradient(low = "black", high = "gray90") +
  scale_y_continuous(breaks = brk) +
  facet_grid(. ~ ctrs) +
  theme_minimal() +
  theme(
    text = element_text(size = 7),
    title =  element_text(size = 7),
    legend.position = "bottom",
    legend.key.height = unit(.5, "lines"),
    legend.key.width = unit(3, "lines"),
    legend.text.align = 0,
    panel.grid.minor = element_blank()
  )

p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 7.7",
      strwrap(
        "Volcano plot - Response Size and Significance of Differential
        Expression among all mRNA at Month 3", width = 80
      ),
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_figure(dds_4_gg, height = inches(3), width = inches(6)) |>
  add_footer(
    c(
      "Program t2d_07_mir / Env ayup_dbs:v0.1.0-alpha",
      format(Sys.time(), format = "%Y-%m-%d %H:%M (%Z)")
    ),
    cfg_prog$version
  )

export_as(
  p,
  file = file.path(cfg_prog$paths$grh, "fig_07_07.pdf"),
  file_graph_alone = file.path(cfg_prog$paths$grh, "fig_07_07_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_07_07.pdf
```

```
[log] output saved as: ../tlg/graph/fig_07_07_af.pdf (annot. free)
```

```
show_slate(p)
```

5.5.2 Fig 07 08

```
dds_4_est$label <- gsub("ENSG0+", "", dds_4_est$feature)
head(dds_4_est)

pval <- c(0.05, 0.001)
log_adj <- pretty(dds_4_est$log_padj)
brk <- setNames(
  c(log_adj, -1 * log10(pval)),
  c(log_adj, paste0("padj=", pval))
)

dds_4_gg <- ggplot(dds_4_est, mapping = aes(x = log2FoldChange, y = log_padj)) +
  geom_vline(xintercept = c(-1, 1), lty = 2) +
  xlab("Log2-fold-change") +
  ylab(expression(-1 %*% log10(padj))) +
  stat_bin_hex(mapping = aes(fill = log10(..count..)), color = "transparent") +
  geom_point(
    data = subset(dds_4_est, padj < 0.001 & abs(log2FoldChange) > 2),
    size = 1,
    shape = 3,
    color = "royalblue"
  ) +
  geom_text(
    data = subset(dds_4_est, padj < 0.001 & abs(log2FoldChange) > 2),
    aes(label = label),
    vjust = -0.5,
    size = 2,
    color = "royalblue",
    show.legend = FALSE
  ) +
  scale_fill_gradient(low = "black", high = "gray90") +
  scale_color_manual(values = c("#FF5003", "#003F7D")) +
  scale_y_continuous(breaks = brk) +
  facet_wrap(. ~ ctrs, ncol = 3) +
  coord_cartesian(clip = "off") +
  theme_minimal() +
  theme(
    text = element_text(size = 7),
    title =  element_text(size = 7),
    legend.position = "bottom",
    legend.key.height = unit(.5, "lines"),
    legend.key.width = unit(2, "lines"),
    legend.text.align = 0,
    panel.grid.minor = element_blank()
  )

p <- clean_slate() |>
  add_header(c("FCA Collin", "UMB BESD"), c("Confidential", "Draft")) |>
  add_title(
    c(
      "Figure 7.8",
      strwrap(
        "Volcano plot - Response Size and Significance of Differential
        Expression among mRNA at Month 3", width = 80
      ),
      "Analysis Set: Full Analysis Set"
    )
  ) |>
  add_figure(dds_4_gg, height = inches(3), width = inches(6)) |>
  add_footer(
    c(
      "Program t2d_07_mir / Env ayup_dbs:v0.1.0-alpha",
      format(Sys.time(), format = "%Y-%m-%d %H:%M (%Z)")
    ),
    cfg_prog$version
  )

export_as(
  p,
  file = file.path(cfg_prog$paths$grh, "fig_07_08.pdf"),
  file_graph_alone = file.path(cfg_prog$paths$grh, "fig_07_08_af.pdf")
)
```

```
[log] output saved as: ../tlg/graph/fig_07_08.pdf

[log] output saved as: ../tlg/graph/fig_07_08_af.pdf (annot. free)
```

```
show_slate(p)
```

5.6 Session Informations

```
sessioninfo::session_info()
```

6 WGCNA

Program 11

Francois Collin
2022-07-01

```
cfg_prog <- yaml::read_yaml("_prog.yml")
```

```
devtools::load_all("src/pkg/dbs.data")
```

```
i Loading dbs.data
```

```
devtools::load_all("src/pkg/latarnia.utils")
```

```
i Loading latarnia.utils
```

```
Loading required package: grid
```

```
Loading required package: shiny
```

```
knitr::opts_chunk$set(results = cfg_prog$knitr$results)

source("R/ngs.R")
source("R/inches.R")
```

The analysis was realized with R (v4.2.0, R Core Team 2022). Note the use of the additional packages downloaded from the RStudio package manager repository (freeze date 2022-05-04, repos):

- DESeq2 for differential expression analysis and variance stabilizing transformation (version 1.36.0, Love, Huber, and Anders 2014)

6.0.1 Materials and Methods

6.0.1.1 Helper functions

```r
# Help for data pre-processing
filter_for_depth <- function(mae, assay, depth_threshold) {
  mae[, colSums(mae[[ngs_assay]]) > depth_threshold, ]
}

filter_for_visit <- function(mae, visit) {
  mae[, colData(mae)$avisit == visit, ]
}

filter_for_low_expr <- function(mae, assay, cpm_threshold,
                                frac_cols = cfg_prog$rna$cpm_threshold$fraccol
) {
  # Genes expressed at least cpm_threshold in frac_cols columns
  mae[
    rowSums(cpm(mae[[assay]]) > cpm_threshold) >
      ncol(mae[[assay]]) * frac_cols,
    ,
  ]
}
```

6.0.1.2 Data preparation

```r
adsl <- dbs.data::adsl
advs <- dbs.data::advs
adlb <- dbs.data::adlb

#' Subjid and Visit to Sample
#'
subjvis_to_spl <- function(df) paste0(df$subjid, "v", df$avisitn)

ads <- adlb |>
  rbind(advs) |>
  subset(select = -c(ct, dtype, param, base, basetype, chg, pchg)) |>
  tidyr::pivot_wider(names_from = "paramcd", values_from = "aval") |>
  merge(adsl, y = _, by = "subjid") |>
  (\(df) S4Vectors::DataFrame(df, row.names = subjvis_to_spl(df)))() |>
  subset(select = c(subjid, diabcd, diab, avisitn, avisit, GLU120)) |>
  (\(df) {
    assertthat::assert_that(all(table(subjvis_to_spl(df)) == 1))
    df
  })()

ads
```

```r
rna <- list(
  mrna = dbs.data::mrna_raw,
  premirna = dbs.data::premirna_raw,
  mirna = dbs.data::mirna_raw
)

rna[c("premirna", "mirna")] <- lapply(
  X = rna[c("premirna", "mirna")],
  FUN = format_mirna
)

# Rows represent genes
rna <- lapply(X = rna, FUN = function(x) y <- x[rowSums(x) > 0, ])
rna <- lapply(X = rna, as.matrix)
assertthat::assert_that(all(colnames(rna$premirna) == colnames(rna$mirna)))
rna$allmirna <- rbind(rna$premirna, rna$mirna)

library(testthat)
test_that("rna features discriminated in noexpr, expr", {
  lapply(X = rna, FUN = \(x) expect_true(all(rowSums(x) > 0)))
})
```

```r
library(MultiAssayExperiment)
```

```
Loading required package: SummarizedExperiment

Loading required package: MatrixGenerics

Loading required package: matrixStats


Attaching package: 'MatrixGenerics'
```

```
The following objects are masked from 'package:matrixStats':

    colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
    colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
    colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
    colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
    colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
    colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
    colWeightedMeans, colWeightedMedians, colWeightedSds,
    colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
    rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
    rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
    rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
    rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
    rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
    rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
    rowWeightedSds, rowWeightedVars

Loading required package: GenomicRanges

Loading required package: stats4

Loading required package: BiocGenerics


Attaching package: 'BiocGenerics'

The following objects are masked from 'package:stats':

    IQR, mad, sd, var, xtabs

The following objects are masked from 'package:base':

    anyDuplicated, append, as.data.frame, basename, cbind, colnames,
    dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
    grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
    order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
    rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
    union, unique, unsplit, which.max, which.min

Loading required package: S4Vectors
```

```
Attaching package: 'S4Vectors'

The following objects are masked from 'package:base':

    expand.grid, I, unname

Loading required package: IRanges

Loading required package: GenomeInfoDb

Loading required package: Biobase

Welcome to Bioconductor

    Vignettes contain introductory material; view with
    'browseVignettes()'. To cite Bioconductor, see
    'citation("Biobase")', and for packages 'citation("pkgname")'.


Attaching package: 'Biobase'

The following object is masked from 'package:MatrixGenerics':

    rowMedians

The following objects are masked from 'package:matrixStats':

    anyMissing, rowMedians
```

```
#' (Sample-)Map Arrays
#'
#' Use the column names of `x` to deduce the `primary` and `colnames`.
#' This is used to generate the sample mapping between colData and Experiments.
#'
#' @param x (`dataframe`).
#'
#' @note In our case, primary and colnames are equivalent, colnames could
#' be different from primary names when a biological sample has different
```

```r
#' names in the biological assays (e.g. machine constraint, technical
#' repetitions).
#'
#' @seealso [MultiAssayExperiment::listToMap()]
#' @examples
#' \dontrun{
#'   lapply(rna, map_arrays)
#'   MultiAssayExperiment::listToMap(lapply(rna, map_arrays))
#' }
#'
map_arrays <- function(x) {
  y <- data.frame(colname = colnames(x))
  y$primary <- y$colname
  y
}

besd_mae <- MultiAssayExperiment(
  experiments = ExperimentList(rna),
  colData = ads,
  sampleMap = listToMap(lapply(rna, map_arrays))
)

besd_mae
```

```r
#' # https://horvath.genetics.ucla.edu/html/CoexpressionNetwork/Rpackages/WGCNA/Tutorials/
#' # WGCNA: gene in rows, sample in cols

ctrl <- cfg_prog$rna
ngs_assay <- "allmirna"
dta <- besd_mae |>
  (\(mae) mae[ , , ngs_assay])() |>
  filter_for_depth(ngs_assay, ctrl$depth_threshold[[ngs_assay]]) |>
  filter_for_visit("Baseline") |>
  filter_for_low_expr(ngs_assay, ctrl$cpm_threshold[[ngs_assay]]) |>
  (\(mae) {
    mae[[ngs_assay]] <-
      DESeq2::DESeqDataSetFromMatrix(
        countData = mae[[ngs_assay]],
        colData = colData(x = mae),
        design = stats::formula(~ diabcd)
      ) |>
      DESeq2::varianceStabilizingTransformation(blind = FALSE) |>
      SummarizedExperiment::assay()
    mae
  })()
```

```
Warning: 'experiments' dropped; see 'metadata'
```

```
harmonizing input:
  removing 282 sampleMap rows not in names(experiments)
```

```
-- note: fitType='parametric', but the dispersion trend was not well captured by the
   function: y = a/x + b, and a local regression fit was automatically substituted.
   specify fitType='local' or 'mean' to avoid this message next time.
```

```
#' WGCNA Expression data.
#'
#' A matrix or data frame in which columns are genes and rows ar samples.
#'
dta_expr <- t(dta[[ngs_assay]])
assert_that(WGCNA::goodSamplesGenes(dta_expr, verbose = 0)$allOK)
```

6.0.2 Sample clustering

```
spl_clust <- hclust(dist(dta_expr), method = "average")

# Clustering Dendrogram of samples based on their Euclidean distance
theme_custom <- function(...) {
    theme_dendro() +
    theme(
      plot.margin = unit(c(0, 0, 0, 0), "null"),
      panel.spacing = unit(c(0, 0, 0, 0), "null")
    ) +
    theme(...)
}

library(ggdendro)
```

```r
library(ggplot2)
dend <- spl_clust |> as.dendrogram()
lim <- c(0, nobs(dend))
ddata <- ggdendro::dendro_data(dend, type = "rectangle")

p1 <- ggplot(ggdendro::segment(ddata)) +
  geom_segment(aes(x = x, y = y, xend = xend, yend = yend)) +
  coord_flip(clip = "off") +
  scale_y_reverse(expand = expansion(0)) +
  scale_x_continuous(
    breaks = seq_len(nobs(dend)),
    labels = ggdendro::label(ddata)$label,
    lim = lim,
    position = "top"
  ) +
  theme_custom(axis.text.y = element_text())

dta_traits <- colData(dta)
dta_traits$x <- rownames(dta_traits)
dta_traits$x <- factor(dta_traits$x, levels = labels(dend))
dta_traits <- as.data.frame(dta_traits)

p2 <- ggplot(
  dta_traits,
  aes(
    xmin = 0, xmax = 1,
    ymin = as.numeric(x) - 1/2, ymax = as.numeric(x) + 1/2,
    fill = diabcd
  )
) +
  geom_rect(col = "white") +
  scale_y_continuous(breaks = 1:34) +
  scale_x_continuous(expand = expansion(0)) +
  scale_fill_viridis_d(option = "C", direction = -1, begin = .2, end = .8) +
  coord_cartesian(ylim = lim) +
  theme_custom(legend.position = "top")

p4 <- ggplot(
  dta_traits,
  aes(
    xmin = 0, xmax = 1,
    ymin = as.numeric(x) - 1/2, ymax = as.numeric(x) + 1/2,
    fill = GLU120
  )
) +
  geom_rect(col = "white") +
  scale_y_continuous(breaks = 1:34) +
  scale_x_continuous(expand = expansion(0)) +
  scale_fill_viridis_c() +
  coord_cartesian(ylim = lim) +
  theme_custom(legend.position = "top")

p <- egg::ggarrange(
  p1, p2, p4,
  nrow = 1,
  widths = c(2, 1, 1),
  padding = unit(0, "lines"),
  draw = FALSE
)
p <- clean_slate() |>
  add_title(c(
    "Figure xxx",
    wrap_long_lines(
      "Clustering dendrogram of samples based on their Euclidean distance
      and trait heatmap"
    )
  )) |>
  add_figure(p, height = inches(5))
```

```
show_slate(p)
```

6.0.3 Soft threshold determination

```
library(WGCNA)
```

```
Loading required package: dynamicTreeCut

Loading required package: fastcluster


Attaching package: 'fastcluster'
```

```
The following object is masked from 'package:stats':

    hclust


Attaching package: 'WGCNA'

The following object is masked from 'package:IRanges':

    cor

The following object is masked from 'package:S4Vectors':

    cor

The following object is masked from 'package:stats':

    cor
```

```r
enableWGCNAThreads()

sft <- pickSoftThreshold(
  dta_expr,
  powerVector = c(c(1:10), seq(from = 12, to=20, by=2)),
  verbose = 0
)

library(ggplot2)
pwr <- cfg_prog$rna$wgcna$sft$allmirna
p1 <- ggplot(
  sft$fitIndices,
  aes(
    Power,
    - sign(slope) * SFT.R.sq,
    label = Power,
    shape = Power == pwr
  )
) +
  geom_hline(yintercept = .9, lty = 2) +
  geom_hline(yintercept = c(0, 1), lty = 1) +
  geom_line() +
  geom_point(shape = 3) +
  geom_label(
    data = sft$fitIndices |> subset(Power == 5),
    vjust = 0, nudge_y = .025
  ) +
  geom_segment(
    data = sft$fitIndices |> subset(Power == 5),
    aes(xend = Power, yend = 0)
  ) +
  geom_segment(
    data = sft$fitIndices |> subset(Power == 5),
    aes(xend = -Inf, yend = - sign(slope) * SFT.R.sq)
  ) +
  geom_point(show.legend = FALSE, fill = "white") +
  scale_shape_manual(values = c("TRUE" = 21, "FALSE" = 3)) +
  coord_cartesian(ylim = c(0, 1)) +
  xlab("Soft Threshold (power)") +
  ylab("Scale Free Topology Model Fit") +
  ggtitle("Scale independence") +
  theme_minimal() +
  theme(panel.grid.minor = element_blank())

p2 <- ggplot(
  sft$fitIndices,
  aes(Power, mean.k., label = Power, shape = Power == pwr)) +
  geom_hline(yintercept = .9) +
  geom_line() +
  geom_segment(
    data = sft$fitIndices |> subset(Power == 5),
    aes(xend = Power, yend = 0)
  ) +
  geom_segment(
    data = sft$fitIndices |> subset(Power == 5),
    aes(xend = -Inf, yend = mean.k.)
  ) +
  geom_label(
    data = sft$fitIndices |> subset(Power == 5),
    vjust = 0, hjust = 0, nudge_y = 1
  ) +
  geom_point(fill = "white", show.legend = FALSE) +
  scale_shape_manual(values = c("TRUE" = 21, "FALSE" = 3)) +
  xlab("Soft Threshold (power)") +
  ylab("Mean connectivity") +
  ggtitle("Mean connectivity") +
  theme_minimal() +
  theme(panel.grid.minor = element_blank())

p <- egg::ggarrange(p1, p2, nrow = 1, draw = FALSE)
p <- clean_slate() |>
  add_title(c(
    "Figure xxx",
    "Soft threshold determination"
  )) |>
  add_figure(p, width = inches(6), height = inches(3))
```

```
show_slate(p)
```

6.0.4 Gene clustering

```
pwr <- cfg_prog$rna$wgcna$sft$allmirna
adjc <- adjacency(dta_expr, power = pwr)

# Turn adjacency into topological overlap
tom <- TOMsimilarity(adjc)
diss_tom <- 1 - tom

gene_hc <- hclust(as.dist(diss_tom), method = "average")

# We like large modules, so we set the minimum module size relatively high:
minModuleSize <- 30

# Module identification using dynamic tree cut:
dyn_mods <- cutreeDynamic(
  dendro = gene_hc,
  distM = diss_tom,
  deepSplit = 2,
  pamRespectsDendro = FALSE,
  minClusterSize = minModuleSize
)
table(dyn_mods)
```

```r
# Convert numeric lables into colors
dyn_col <- labels2colors(dyn_mods)
table(dyn_col)
```

```r
# Plot the dendrogram and colors underneath
plotDendroAndColors(
  gene_hc,
  dyn_col,
  "Dynamic Tree Cut",
  dendroLabels = FALSE, hang = 0.03,
  addGuide = TRUE, guideHang = 0.05,
  main = "Gene dendrogram and module colors"
)
```

6.0.4.1 Legacy

6.0.4.2 Data preparation

6.1 Session Informations

```r
sessioninfo::session_info()
```

References

Committee for Medicinal Products for Human Use (CHMP). 2015. "Guideline on Adjustment for Baseline Covariates in Clinical Trials." European Medicines Agency. https://www.ema.europa.eu/en/documents/scientific-guideline/guideline-adjustment-baseline-covariates-clinical-trials_en.pdf.

COMMITTEE FOR PROPRIETARY MEDICINAL PRODUCTS. 2003. "POINTS TO CONSIDER ON ADJUSTMENT FOR BASELINE COVARIATES." The European Agency for the Evaluation of Medicinal Products Evaluation of Medicines for Human Use. https://www.ema.europa.eu/en/documents/scientific-guideline/points-consider-adjustment-baseline-covariates_en.pdf.

Dündar, Friederike, Luce Skrabanek, and Paul Zumbo. 2018. "Introduction to Differential Gene Expression Analysis Using RNA-Seq." Internet.

Fox, John, and Sanford Weisberg. 2019. *An R Companion to Applied Regression*. Third. Thousand Oaks CA: Sage. https://socialsciences.mcmaster.ca/jfox/Books/Companion/.

Gohel, David. 2022. *Flextable: Functions for Tabular Reporting*.

Josse, Julie, and François Husson. 2016. "missMDA: A Package for Handling Missing Values in Multivariate Data Analysis." *Journal of Statistical Software* 70 (1): 1–31. https://doi.org/10.18637/jss.v070.i01.

Lenth, Russell V. 2022. *Emmeans: Estimated Marginal Means, Aka Least-Squares Means*. https://github.com/rvlenth/emmeans.

Liu, Guanghan F, Kaifeng Lu, Robin Mogg, Madhuja Mallick, and Devan V Mehrotra. 2009. "Should Baseline Be a Covariate or Dependent Variable in Analyses of Change from Baseline in Clinical Trials?" *Statistics in Medicine* 28 (20): 2509–30. https://doi.org/10.1002/sim.3639.

Love, Michael I., Wolfgang Huber, and Simon Anders. 2014. "Moderated Estimation of Fold Change and Dispersion for RNA-Seq Data with DESeq2." *Genome Biology* 15: 550. https://doi.org/10.1186/s13059-014-0550-8.

O'Connell, Nathaniel S, Lin Dai, Yunyun Jiang, Jaime L Speiser, Ralph Ward, Wei Wei, Rachel Carroll, and Mulugeta Gebregziabher. 2017. "Methods for Analysis of Pre-Post Data in Clinical Research: A Comparison of Five Common Methods." *Journal of Biometrics & Biostatistics* 8 (1): 1. https://doi.org/10.4172/2155-6180.1000334.

R Core Team. 2022. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Ramos, Marcel, Lucas Schiffer, Angela Re, Rimsha Azhar, Azfar Basunia, Carmen Rodriguez Cabrera, Tiffany Chan, et al. 2017. "Software for the Integration of Multi-Omics Experiments in Bioconductor." *Cancer Research* 77(21); e39-42.

Van Breukelen, G. 2006. "ANCOVA Versus Change from Baseline: More Power Inrandomized Studies, More Bias in Nonrandomized Studies." *Journal of ClinicalEpidemiology*, 59–920. https://doi.org/10.1016/j.jclinepi.2006.02.007.

Vickers, Andrew J. 2001. "The Use of Percentage Change from Baseline as an Outcome in a Controlled Trial Is Statistically Inefficient: A Simulation Study." *BMC Medical Research Methodology* 1 (1): 1–4. https://doi.org/10.1186/1471-2288-1-6.

Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. https://ggplot2.tidyverse.org.

Wickham, Hadley, and Maximilian Girlich. 2022. *Tidyr: Tidy Messy Dataversion*.

Zhang, Shiyuan, James Paul, Manyat Nantha-Aree, Norman Buckley, Uswa Shahzad, Ji Cheng, Justin DeBeer, et al. 2014. "Empirical Comparison of Four Baseline Covariate Adjustment Methods in Analysis of Continuous Outcomes in Randomized Controlled Trials." *Clinical Epidemiology* 6: 227. https://doi.org/10.2147/clep.s56554.