

# Reanalysis Johnson2007 - figures and code

## *immediate*

2020-11-17

The `sessionInfo()` is provided at the end of the document.

```
rm(list = ls())

library(GGally)
library(ggplot2)
library(limma)
library(sva)
library(qvalue)
library(randRotation)
```

File from <https://github.com/ous-uiو-bioinfo-core/batch-adjust-warning-figures/tree/master/reanalysis/Johnson2007/data> See also Nygaard, V., Rodland, E. A. & Hovig, E. Methods that remove batch effects while retaining group differences may lead to exaggerated confidence in downstream analyses. Biostatistics kxv027 (2015). doi:10.1093/biostatistics/kxv027

```
edata <- read.table(file = "dataExample2.txt", header = TRUE,
                     sep = "\t", dec = ".", stringsAsFactors = FALSE)
```

File from <https://github.com/ous-uiو-bioinfo-core/batch-adjust-warning-figures/tree/master/reanalysis/Johnson2007/data> See also Nygaard, V., Rodland, E. A. & Hovig, E. Methods that remove batch effects while retaining group differences may lead to exaggerated confidence in downstream analyses. Biostatistics kxv027 (2015). doi:10.1093/biostatistics/kxv027

```
pdata <- read.table(file = "sampleInfoExample2.txt", header = TRUE,
                     sep = "\t", dec = ".", stringsAsFactors = TRUE,
                     row.names = 1)
```

```
samps <- which(pdata$Type != "WT")

pdata <- pdata[samps,]
edata <- edata[,samps]
pdata$Type <- droplevels(pdata$Type)
pdata$Batch <- as.factor(pdata$Batch)

all.equal(rownames(pdata), colnames(edata))
## [1] TRUE
```

```
# flooring to 1
edata[edata<1] <- 1
# take out data with too many low/missing values.
negativeprobesfilter <- rowSums(edata>1) >= (0.9*ncol(edata))
```

## Reanalysis Johnson2007 - figures and code

```
edata <- edata[negativeprobesfilter,]
# quantilenormalize
edata.quan <- normalizeBetweenArrays(log2(edata), method="quantile")

# for debugging
debug = FALSE
if(debug) edata <- edata[1:1000,]

##### ComBat - "p ComBat" values

mod.com <- model.matrix(~Type, pdata)
edata.com <- ComBat(edata.quan,
                      batch = pdata$Batch,
                      mod = mod.com)
## Found3batches
## Adjusting for 1 covariate(s) or covariate level(s)
## Standardizing Data across genes
## Fitting L/S model and finding priors
## Finding parametric adjustments
## Adjusting the Data

mod.fit <- model.matrix(~Type, pdata)
fit1 <- lmFit(edata.com, design = mod.fit)
fit1 <- eBayes(fit1)

ps.com <- topTable(fit1, number = Inf, sort.by = "none")$P.Value
## Removing intercept from test coefficients
fdr.com <- topTable(fit1, number = Inf, sort.by = "none")$adj.P.Val
## Removing intercept from test coefficients

sum(fdr.com<0.05)
## [1] 649
sum(qvalue(ps.com)$qvalues < 0.05)
## [1] 814

##### limma batch as covariate - "p Limma (+batch)" values

mod.fit <- model.matrix(~Type + Batch, pdata)
fit2 <- lmFit(edata.quan, design = mod.fit)
fit2 <- eBayes(fit2)

ps.lim <- topTable(fit2, coef = 2, number = Inf, sort.by = "none")$P.Value
fdr.lim <- topTable(fit2, coef = 2, number = Inf, sort.by = "none")$adj.P.Val

sum(fdr.lim<0.05)
## [1] 271
sum(qvalue(ps.lim)$qvalues < 0.05)
## [1] 392
```

## Reanalysis Johnson2007 - figures and code

```
#### ComBat with random rotations - "p ComBat - 2000 rot." values

mod.fit <- model.matrix(~Type, pdata)

rr1 <- initBatchRandrot(edata.quan, mod.fit, 2, pdata$Batch)
## Initialising batch "1"
## Initialising batch "2"
## Initialising batch "3"

statistic <- function(Y, batch, mod){
  edata.com <- sva::ComBat(Y,
    batch = batch,
    mod = mod, mean.only = FALSE)

  fit1 <- limma::lmFit(edata.com, design = mod)
  fit1 <- limma::eBayes(fit1)

  abs(limma::topTable(fit1, number = Inf, sort.by = "none")$t)
}

rs1 <- rotateStat(rr1, R = 2000, statistic = statistic, pdata$Batch,
  mod.fit, parallel = TRUE)

ps.rot <- pFdr(rs1)
fdr.rot <- p.adjust(ps.rot, "BH")

sum(fdr.rot<0.05)
## [1] 420
sum(qvalue(ps.rot)$qvalues < 0.05)
## [1] 515

ps <- cbind(ps.com, ps.lim, ps.rot = ps.rot[,1])
colnames(ps) <- c("p ComBat", "p Limma (+batch)", "p ComBat - 2000 rot.")

apply(ps, 2, function(i)sum(qvalue::qvalue(i)$qvalues < 0.05))
##           p ComBat      p Limma (+batch) p ComBat - 2000 rot.
##                  814                  392                  515
apply(ps, 2, function(i)sum(p.adjust(i, "BH") < 0.05))
##           p ComBat      p Limma (+batch) p ComBat - 2000 rot.
##                  649                  271                  420

#### p-vals scatterplot

df1 <- data.frame(ps)
colnames(df1) <- colnames(ps)

F1 <- function(...){
  ggally_points(..., size = 0.7) +
    scale_y_log10() +
    scale_x_log10() +
```

## Reanalysis Johnson2007 - figures and code

```

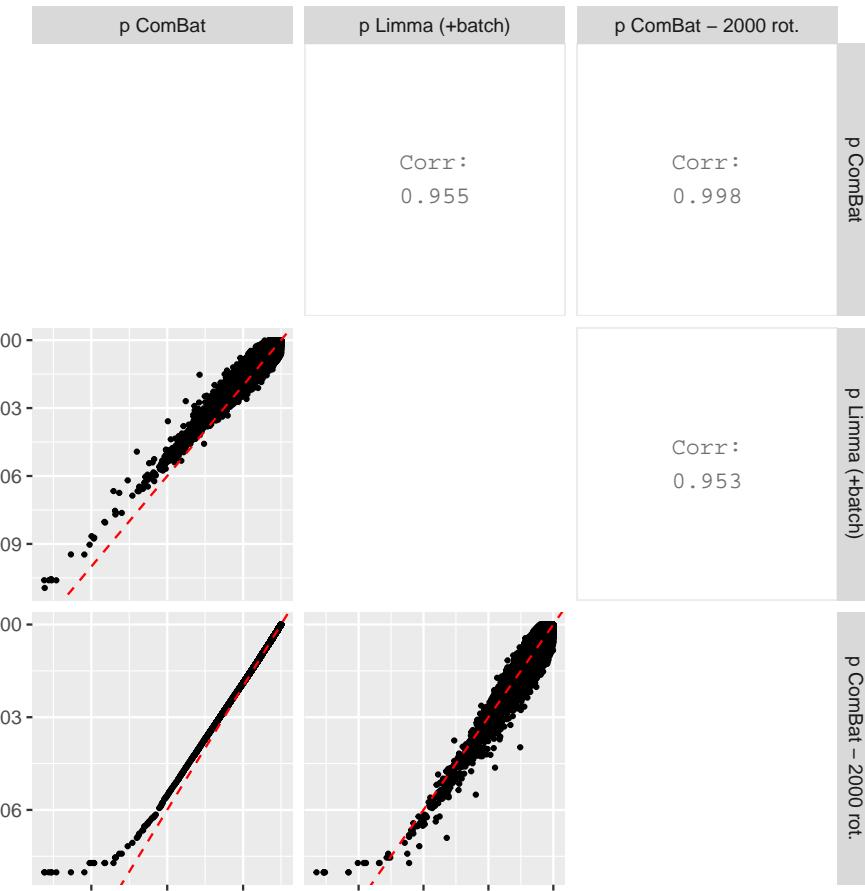
        geom_abline(slope = 1, intercept = 0, lty = 2, lwd = 0.5, col = "red")
    }
lower.pan <- list(continuous = F1, combo = "facethist", discrete = "facetbar",
na = "na")

my.cor <- function(...)
  ggally_statistic(
    text_fn =
      function(x,y)formatC(cor(log(x),log(y)), digits = 3, format = "f"),
    title = "Corr",
    sep = ":\n",...)

upper.pan <- list(continuous = my.cor, combo = "box_no_facet",
discrete = "count", na = "na")

ggpairs(df1, lower = lower.pan, upper = upper.pan, diag = NULL)

```

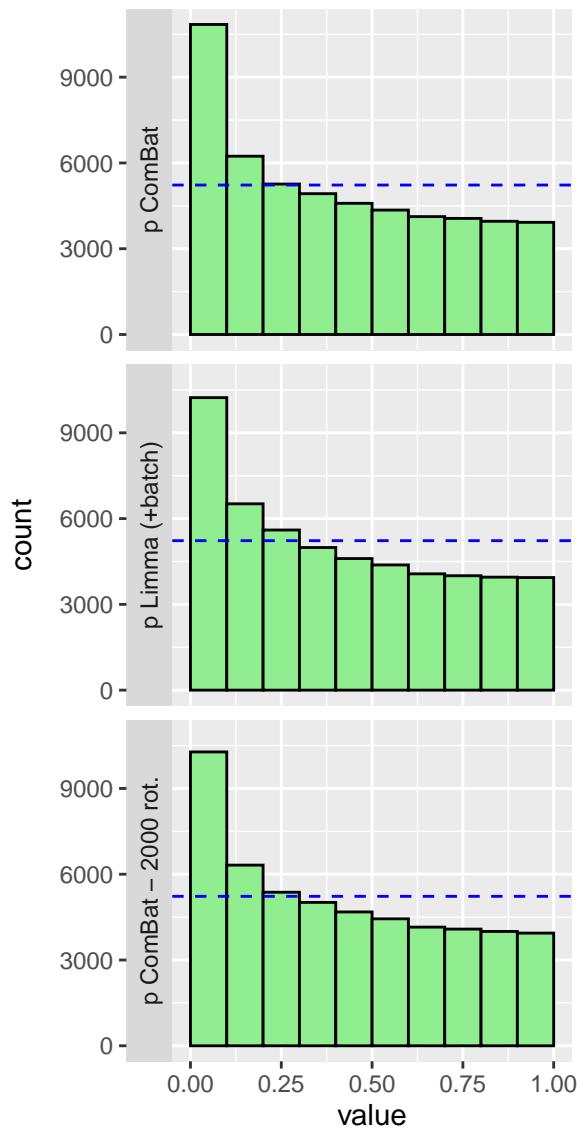


```
##### Histograms
```

```
df2 <- reshape2::melt(ps)
```

## Reanalysis Johnson2007 - figures and code

```
ggplot(df2, aes(x=value))+  
  geom_histogram(colour="black", fill="lightgreen", binwidth =0.1, boundary=0)+  
  facet_grid(Var2 ~ ., switch = "y") +  
  geom_abline(slope = 0, intercept = nrow(ps)/10, lty = 2, col = "blue") +  
  theme(axis.title.y = element_text(vjust=+3.3))
```



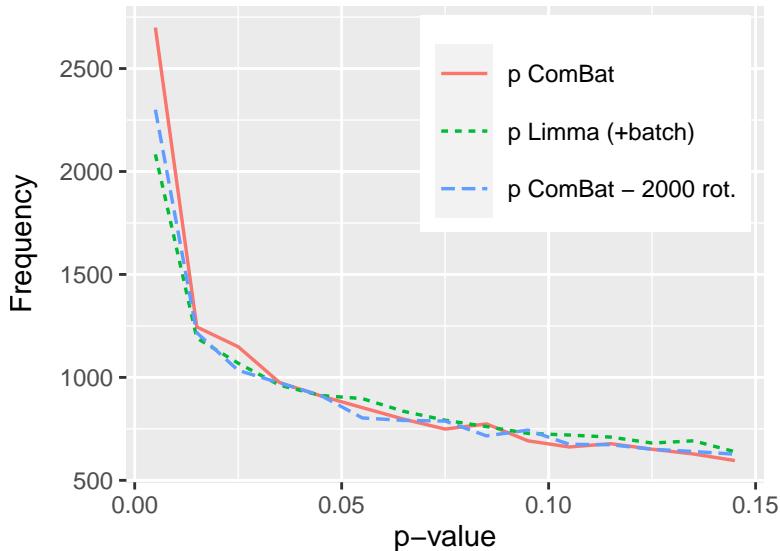
```
## Histograms 2  
  
ind <- 1:15  
h.com <- hist(ps.com, breaks = 100, plot = FALSE)  
h.lim <- hist(ps.lim, breaks = 100, plot = FALSE)  
h.rot <- hist(ps.rot, breaks = 100, plot = FALSE)  
  
maxcount <- max(h.com$counts, h.lim$counts, h.rot$counts)
```

## Reanalysis Johnson2007 - figures and code

```
lab <- factor(rep(colnames(ps), rep(length(ind), 3)), levels = colnames(ps))

df1 <- data.frame(
  mids = c(h.com$mids[ind], h.lim$mids[ind], h.rot$mids[ind]),
  counts = c(h.com$counts[ind], h.lim$counts[ind], h.rot$counts[ind]),
  lab = lab)

ggplot(df1, aes(x = mids, y = counts, colour = lab, lty = lab))+
  geom_line(lwd = 0.6) +
  xlab("p-value") + ylab("Frequency") +
  theme(axis.title.y = element_text(vjust=+3.3)) +
  theme(legend.justification=c(1,1), legend.position=c(0.98, 0.98),
        legend.title = element_blank()) +
  theme(legend.key.size = unit(1.5,"line"))
```



## 1 Session Info

```
sessionInfo()
## R Under development (unstable) (2020-11-14 r79432)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19041)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=German_Austria.1252  LC_CTYPE=German_Austria.1252    LC_MONETARY=German_Austria.1252
## [4] LC_NUMERIC=C                      LC_TIME=German_Austria.1252
##
## attached base packages:
## [1] stats      graphics   grDevices utils      datasets   methods    base
##
```

## Reanalysis Johnson2007 - figures and code

```
## other attached packages:
## [1] qvalue_2.23.0      randRotation_1.3.1   sva_3.39.0      BiocParallel_1.25.1  genefilter_1.73.0
## [6] mgcv_1.8-33        nlme_3.1-150       limma_3.47.0     GGally_2.0.0       ggplot2_3.3.2
## [11] BiocStyle_2.19.0
##
## loaded via a namespace (and not attached):
## [1] Rcpp_1.0.5          locfit_1.5-9.4    lattice_0.20-41  snow_0.4-3       digest_0.6.27
## [6] R6_2.5.0            plyr_1.8.6         stats4_4.1.0     RSQLite_2.2.1     evaluate_0.14
## [11] httr_1.4.2          pillar_1.4.6       Rdpack_2.1       rlang_0.4.8       annotate_1.69.0
## [16] blob_1.2.1          S4Vectors_0.29.3  Matrix_1.2-18    rmarkdown_2.5     labeling_0.4.2
## [21] splines_4.1.0       stringr_1.4.0     tinytex_0.27     bit_4.0.4        munsell_0.5.0
## [26] compiler_4.1.0     xfun_0.19         pkgconfig_2.0.3  BiocGenerics_0.37.0 htmltools_0.5.0
## [31] tibble_3.0.4        bookdown_0.21    edgeR_3.33.0     IRanges_2.25.2   matrixStats_0.57
## [36] XML_3.99-0.5       reshape_0.8.8     crayon_1.3.4     withr_2.3.0      rbibutils_1.4
## [41] grid_4.1.0          xtable_1.8-4     gtable_0.3.0     lifecycle_0.2.0   DBI_1.1.0
## [46] magrittr_1.5        scales_1.1.1     stringi_1.5.3   farver_2.0.3     reshape2_1.4.4
## [51] xml2_1.3.2          ellipsis_0.3.1   vctrs_0.3.4      RColorBrewer_1.1-2 tools_4.1.0
## [56] bit64_4.0.5         Biobase_2.51.0   glue_1.4.2      parallel_4.1.0   survival_3.2-7
## [61] yaml_2.2.1          AnnotationDbi_1.53.0 colorspace_2.0-0  BiocManager_1.30.10 gbRd_0.4-11
## [66] memoise_1.1.0       knitr_1.30
```