

random data - figures and code

Peter Hettegger

2020-11-16

For reproducibility, the `sessionInfo()` is provided at the end of the document.

```
rm(list = ls())

set.seed(0)

library(randRotation)
library(GGally)
## Loading required package: ggplot2
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg      ggplot2
library(ggplot2)

pdata <- data.frame(batch = as.factor(rep(c(1:3), c(10,10,10))),
                    group = as.factor(rep(c("Control", "Case"), c(5,5))))

features <- 2000

# Matrix with random gene expression data
edata <- matrix(rnorm(features * nrow(pdata)), features)
rownames(edata) <- paste("feature", 1:nrow(edata))
```

```
mod1 <- model.matrix(~group, pdata)

##### Raw data (no batch effect) - "p0" values

fit0 <- limma::lmFit(edata, mod1)
fit0 <- limma::eBayes(fit0)
tt0 <- limma::topTable(fit0, 2, Inf, sort.by = "none", adjust.method = "none")
p.0 <- tt0$P.Value
```

```
##### add batch effect #####

## additive and multiplicative batch effect
edata <- t( t(edata)*rep(c(0.8,1,1.2), c(10,10,10)) + rep(c(-2,0,2), c(10,10,10)) )

#####
```

```
##### ComBat - "p ComBat" values
edata.combat <- sva::ComBat(edata, pdata$batch, mod1)
## Found3batches
## Adjusting for1covariate(s) or covariate level(s)
```

random data - figures and code

```
## Standardizing Data across genes
## Fitting L/S model and finding priors
## Finding parametric adjustments
## Adjusting the Data
fit1 <- limma::lmFit(edata.combat, mod1)
fit1 <- limma::eBayes(fit1)
tt1 <- limma::topTable(fit1, 2, Inf, sort.by = "none", adjust.method = "none")
p.com <- tt1$P.Value

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

mod2 <- model.matrix(~group+batch, pdata)
fit2 <- limma::lmFit(edata, mod2)
fit2 <- limma::eBayes(fit2)
tt2 <- limma::topTable(fit2, 2, Inf, sort.by = "none", adjust.method = "none")
p.lim <- tt2$P.Value

##### ComBat with random rotation - "p ComBat - 100 rot." values

rr1 <- initBatchRandrot(edata, mod1, 2, pdata$batch)
## Initialising batch "1"
## Initialising batch "2"
## Initialising batch "3"

statistic <- function(Y, mod, batch){
  Y <- sva::ComBat(Y, batch, mod)
  fit1 <- limma::lmFit(Y, mod)
  fit1 <- limma::eBayes(fit1)
  abs(fit1$t[,2])
}

rs1 <- rotateStat(rr1, 100, statistic, mod1, pdata$batch)

p.rot <- pFdr(rs1, pooled = TRUE)[,1]

ps <- cbind(p.0, p.com, p.lim, p.rot)
colnames(ps) <- c("p0", "p ComBat", "p Limma (+batch)", "p ComBat - 100 rot.")

#### p-vals scatterplot

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

F1 <- function(...){
  ggally_points(..., size = 0.7)+
    scale_y_log10()+
    scale_x_log10()+
    geom_abline(slope = 1, intercept = 0, lty = 2, lwd = 0.5, col = "red")
}
```

random data - figures and code

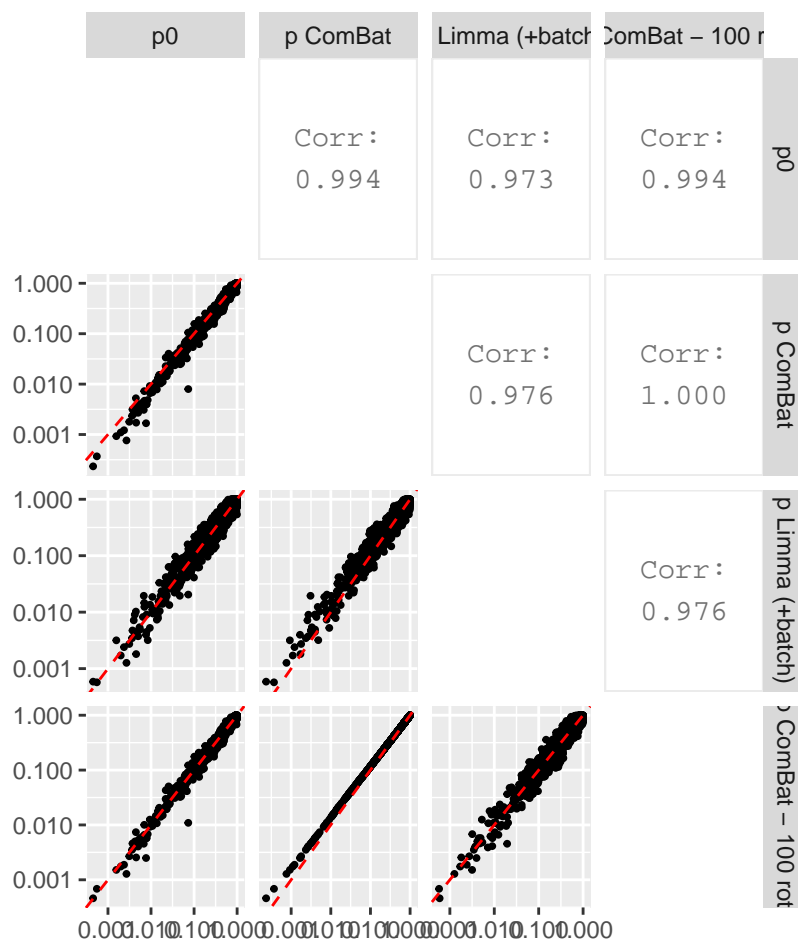
```

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

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

```



Histograms

```

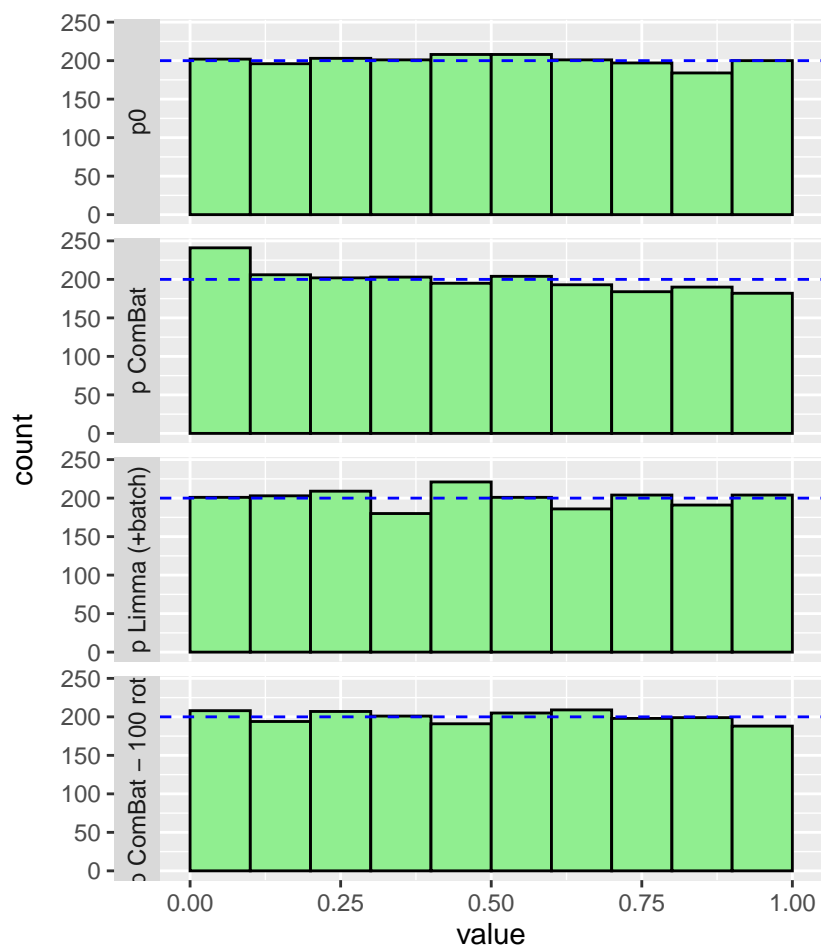
df2 <- reshape2::melt(ps)

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 = 200, lty = 2, col = "blue")+

```

random data - figures and code

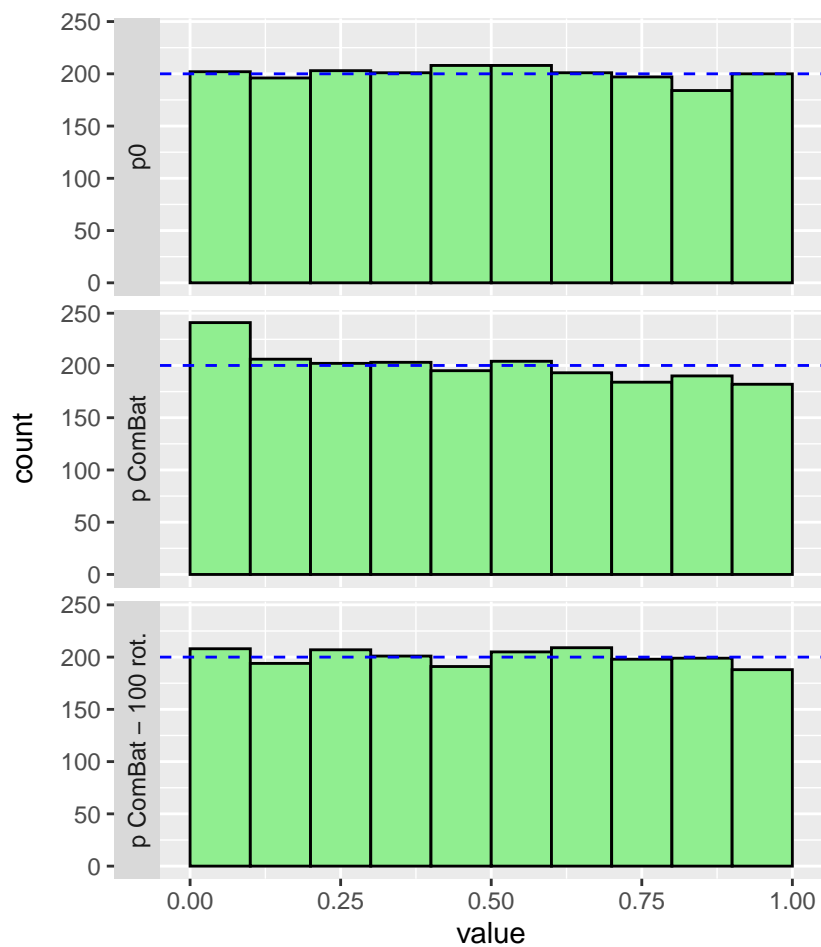
```
theme(axis.title.y = element_text(vjust=+3.3))
```



```
df2 <- reshape2::melt(ps[, -3])

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 = 200, lty = 2, col = "blue")+
  theme(axis.title.y = element_text(vjust=+3.3))
```

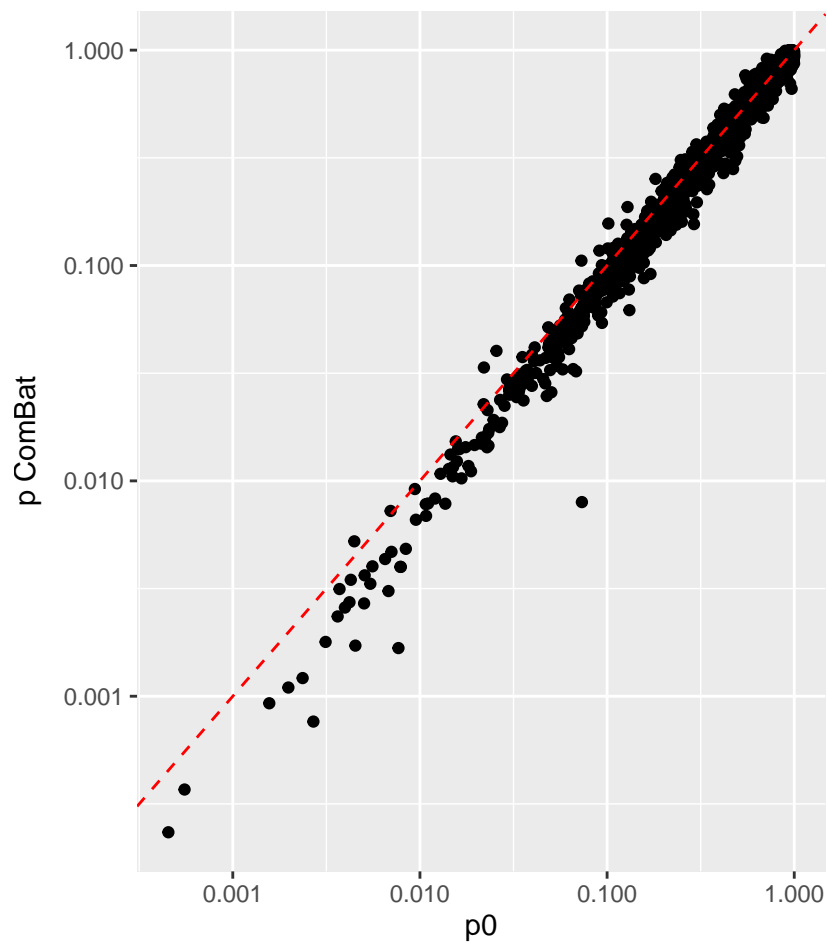
random data - figures and code



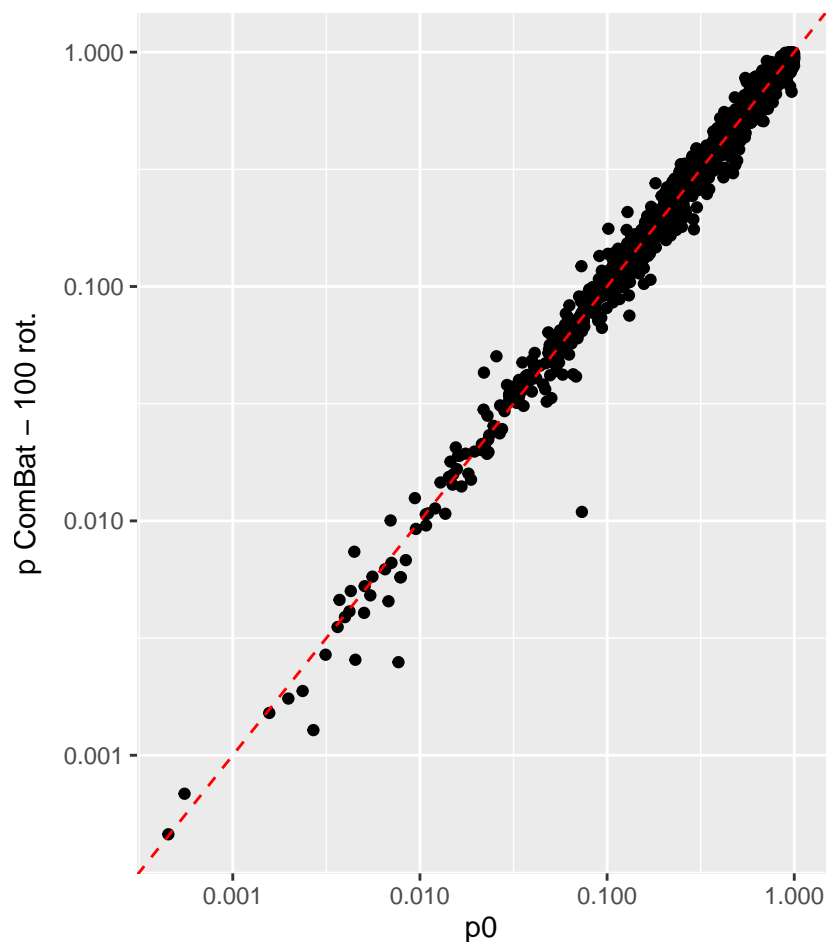
Scatterplots

```
ggplot(data = df1, aes(x = `p0`, y = `p ComBat`)) +  
  geom_point() +  
  scale_y_log10()+  
  scale_x_log10()+  
  theme(axis.title.y = element_text(vjust=+3.3))+  
  geom_abline(slope = 1, intercept = 0, lty = 2, lwd = 0.5, col = "red")
```

random data - figures and code



```
ggplot(df1, aes(x = `p0`, y = `p ComBat - 100 rot.`)) +  
  geom_point() +  
  scale_y_log10()+  
  scale_x_log10()+  
  theme(axis.title.y = element_text(vjust=+3.3))+  
  geom_abline(slope = 1, intercept = 0, lty = 2, lwd = 0.5, col = "red")
```



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
## [3] LC_MONETARY=German_Austria.1252 LC_NUMERIC=C
## [5] LC_TIME=German_Austria.1252
##
## attached base packages:
## [1] stats    graphics  grDevices utils      datasets  methods   base
##
## other attached packages:
## [1] GGally_2.0.0      ggplot2_3.3.2      randRotation_1.3.1 BiocStyle_2.19.0
```

random data - figures and code

```
##
## loaded via a namespace (and not attached):
## [1] Rcpp_1.0.5          locfit_1.5-9.4      lattice_0.20-41
## [4] digest_0.6.27       R6_2.5.0            plyr_1.8.6
## [7] stats4_4.1.0        RSQLite_2.2.1       evaluate_0.14
## [10] sva_3.39.0          httr_1.4.2          pillar_1.4.6
## [13] Rdpack_2.1          rlang_0.4.8         annotate_1.69.0
## [16] blob_1.2.1          S4Vectors_0.29.3    Matrix_1.2-18
## [19] rmarkdown_2.5       labeling_0.4.2       splines_4.1.0
## [22] BiocParallel_1.25.1 stringr_1.4.0        bit_4.0.4
## [25] munsell_0.5.0       compiler_4.1.0      xfun_0.19
## [28] pkgconfig_2.0.3     BiocGenerics_0.37.0 mgcv_1.8-33
## [31] htmltools_0.5.0     tibble_3.0.4        bookdown_0.21
## [34] edgeR_3.33.0        IRanges_2.25.2      matrixStats_0.57.0
## [37] XML_3.99-0.5        reshape_0.8.8       crayon_1.3.4
## [40] withr_2.3.0         rbibutils_1.4        grid_4.1.0
## [43] nlme_3.1-150        xtable_1.8-4        gtable_0.3.0
## [46] lifecycle_0.2.0     DBI_1.1.0           magrittr_1.5
## [49] scales_1.1.1        stringi_1.5.3       reshape2_1.4.4
## [52] farver_2.0.3        genefilter_1.73.0    limma_3.47.0
## [55] xml2_1.3.2          ellipsis_0.3.1      vctrs_0.3.4
## [58] RColorBrewer_1.1-2  tools_4.1.0         bit64_4.0.5
## [61] Biobase_2.51.0      glue_1.4.2          parallel_4.1.0
## [64] survival_3.2-7      yaml_2.2.1          AnnotationDbi_1.53.0
## [67] colorspace_2.0-0    BiocManager_1.30.10 gbRd_0.4-11
## [70] memoise_1.1.0       knitr_1.30
```