

# Additional plots for: Independent filtering increases power for detecting differentially expressed genes, Bourgon et al., PNAS (2010)

*Richard Bourgon and Wolfgang Huber*

May 2, 2023

## Contents

1	Introduction . . . . .	1
2	Data preparation . . . . .	1
3	Filtering volcano plot . . . . .	2
4	Rejection count plots . . . . .	3
4.1	Across $p$ -value cutoffs . . . . .	3
4.2	Across filtering fractions . . . . .	4

## 1 Introduction

---

This vignette illustrates use of some functions in the *genefilter* package that provide useful diagnostics for independent filtering [1]:

- `kappa_p` and `kappa_t`
- `filtered_p` and `filtered_R`
- `filter_volcano`
- `rejection_plot`

## 2 Data preparation

---

Load the ALL data set and the *genefilter* package:

```
library("genefilter")
library("ALL")
data("ALL")
```

Reduce to just two conditions, then take a small subset of arrays from these, with 3 arrays per condition:

```

bcell <- grep("^B", as.character(ALL$BT))
moltyp <- which(as.character(ALL$mol.biol) %in%
               c("NEG", "BCR/ABL"))
ALL_bcrneg <- ALL[, intersect(bcell, moltyp)]
ALL_bcrneg$mol.biol <- factor(ALL_bcrneg$mol.biol)
n1 <- n2 <- 3
set.seed(1969)
use <- unlist(tapply(1:ncol(ALL_bcrneg),
                    ALL_bcrneg$mol.biol, sample, n1))
subsample <- ALL_bcrneg[,use]

```

We now use functions from *genefilter* to compute overall standard deviation filter statistics as well as standard two-sample  $t$  and related statistics.

```

S <- rowSds( exprs( subsample ) )
temp <- rowttests( subsample, subsample$mol.biol )
d <- temp$dm
p <- temp$p.value
t <- temp$statistic

```

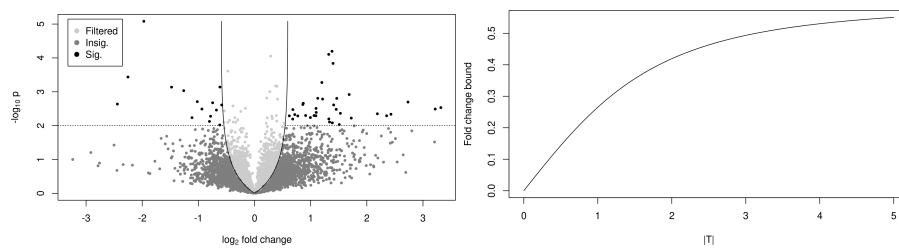
### 3 Filtering volcano plot

Filtering on overall standard deviation and then using a standard  $t$ -statistic induces a lower bound of fold change, albeit one which varies somewhat with the significance of the  $t$ -statistic. The `filter_volcano` function allows you to visualize this effect.

The output is shown in the left panel of Fig. 1.

The `kappa_p` and `kappa_t` functions, used to make the volcano plot, compute the fold change bound multiplier as a function of either a  $t$ -test  $p$ -value or the  $t$ -statistic itself. The actual induced bound on the fold change is  $\kappa$  times the filter's cutoff on the overall standard deviation. Note that fold change bounds for values of  $|T|$  which are close to 0 are not of practical interest because we will not reject the null hypothesis with test statistics in this range.

The plot is shown in the right panel of Fig. 1.



**Figure 1:** Left panel: plot produced by the `filter_volcano` function. Right panel: graph of the `kappa_t` function.

## 4 Rejection count plots

### 4.1 Across $p$ -value cutoffs

The `filtered_p` function permits easy simultaneous calculation of unadjusted or adjusted  $p$ -values over a range of filtering thresholds ( $\theta$ ). Here, we return to the full “BCR/ABL” versus “NEG” data set, and compute adjusted  $p$ -values using the method of Benjamini and Hochberg, for a range of different filter stringencies.

```
table(ALL_bcrneg$mol.biol)
```

```
##
## BCR/ABL    NEG
##      37     42
```

```
S2 <- rowVars(exprs(ALL_bcrneg))
p2 <- rowttests(ALL_bcrneg, "mol.biol")$p.value
theta <- seq(0, .5, .1)
p_bh <- filtered_p(S2, p2, theta, method="BH")
```

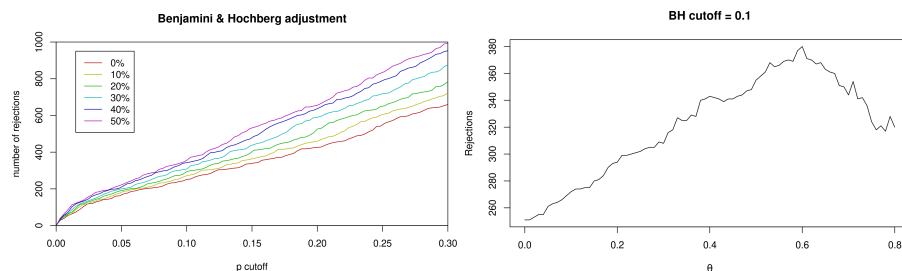
```
head(p_bh)
```

```
##           0%          10%          20%          30%          40%          50%
## [1,] 0.9185626 0.8943104 0.8624798 0.8278077          NA          NA
## [2,] 0.9585758 0.9460504 0.9304104 0.9059466 0.8874485 0.8709793
## [3,] 0.7022442          NA          NA          NA          NA          NA
## [4,] 0.9806216 0.9747555 0.9680574 0.9567131          NA          NA
## [5,] 0.9506087 0.9349386 0.9123998 0.8836386          NA          NA
## [6,] 0.6339004 0.5896890 0.5440851 0.4951371 0.4497915 0.4102711
```

The `rejection_plot` function takes sets of  $p$ -values corresponding to different filtering choices — in the columns of a matrix or in a list — and shows how rejection count ( $R$ ) relates to the choice of cutoff for the  $p$ -values. For these data, over a reasonable range of FDR cutoffs, increased filtering corresponds to increased rejections.

```
rejection_plot(p_bh, at="sample",
               xlim=c(0, .3), ylim=c(0,1000),
               main="Benjamini & Hochberg adjustment")
```

The plot is shown in the left panel of Fig. 2.



**Figure 2:** Left panel: plot produced by the `rejection_plot` function. Right panel: graph of `theta`.

## 4.2 Across filtering fractions

If we select a fixed cutoff for the adjusted  $p$ -values, we can also look more closely at the relationship between the fraction of null hypotheses filtered and the total number of discoveries. The `filtered_R` function wraps `filtered_p` and just returns rejection counts. It requires a  $p$ -value cutoff.

```
theta <- seq(0, .80, .01)
R_BH <- filtered_R(alpha=.10, S2, p2, theta, method="BH")
```

```
head(R_BH)

## 0% 1% 2% 3% 4% 5%
## 251 251 253 255 255 261
```

Because overfiltering (or use of a filter which is inappropriate for the application domain) discards both false and true null hypotheses, very large values of  $\theta$  reduce power in this example:

```
plot(theta, R_BH, type="l",
      xlab=expression(theta), ylab="Rejections",
      main="BH cutoff = 0.1")
```

The plot is shown in the right panel of Fig. 2.

## Session information

- R version 4.3.0 RC (2023-04-18 r84287), x86\_64-pc-linux-gnu
- Locale: LC\_CTYPE=en\_US.UTF-8, LC\_NUMERIC=C, LC\_TIME=en\_GB, LC\_COLLATE=C, LC\_MONETARY=en\_US.UTF-8, LC\_MESSAGES=en\_US.UTF-8, LC\_PAPER=en\_US.UTF-8, LC\_NAME=C, LC\_ADDRESS=C, LC\_TELEPHONE=C, LC\_MEASUREMENT=en\_US.UTF-8, LC\_IDENTIFICATION=C
- Time zone: America/New\_York
- TZcode source: system (glibc)
- Running under: Ubuntu 22.04.2 LTS
- Matrix products: default
- BLAS: /home/biocbuild/bbs-3.18-bioc/R/lib/libRblas.so
- LAPACK: /usr/lib/x86\_64-linux-gnu/lapack/liblapack.so.3.10.0
- Base packages: base, datasets, grDevices, graphics, methods, stats, utils
- Other packages: ALL 1.43.0, Biobase 2.61.0, BiocGenerics 0.47.0, BiocStyle 2.29.0, class 7.3-21, genefilter 1.83.1, knitr 1.42
- Loaded via a namespace (and not attached): AnnotationDbi 1.63.1, BiocManager 1.30.20, Biostings 2.69.0, DBI 1.1.3, GenomInfoDb 1.37.0, GenomInfoDbData 1.2.10, IRanges 2.35.0, KEGGREST 1.41.0, Matrix 1.5-4, MatrixGenerics 1.13.0, R6 2.5.1, RCurl 1.98-1.12, RSQLite 2.3.1, Rcpp 1.0.10, S4Vectors 0.39.0, XML 3.99-0.14, XVector 0.41.0, annotate 1.79.0, bit 4.0.5, bit64 4.0.5, bitops 1.0-7, blob 1.2.4, bookdown 0.33, bslib 0.4.2, cachem 1.0.8,

cli 3.6.1, codetools 0.2-19, compiler 4.3.0, crayon 1.5.2, digest 0.6.31, evaluate 0.20, fastmap 1.1.1, grid 4.3.0, highr 0.10, htmltools 0.5.5, httr 1.4.5, jquerylib 0.1.4, jsonlite 1.8.4, lattice 0.21-8, magick 2.7.4, magrittr 2.0.3, matrixStats 0.63.0, memoise 2.0.1, png 0.1-8, rlang 1.1.1, rmarkdown 2.21, sass 0.4.5, splines 4.3.0, stats4 4.3.0, survival 3.5-5, tools 4.3.0, vctrs 0.6.2, xfun 0.39, xtable 1.8-4, yaml 2.3.7, zlibbioc 1.47.0

## References

- [1] Richard Bourgon, Robert Gentleman and Wolfgang Huber. Independent filtering increases power for detecting differentially expressed genes.