Recount Meta-Analysis

Jeff Leek and Margaret Taub

June 7, 2016

Contents

Analysis

Load libraries we will need

```
library('dplyr')
library('recount')
library('magrittr')
library('limma')
library('edgeR')
library('ffpe')
library('RSkittleBrewer')
library('SummarizedExperiment')
library('devtools')
trop = RSkittleBrewer::RSkittleBrewer('tropical')
```

Get data sets

We identify two projects consisting of samples from colon `SRP029880`,`SRP042228` and three that contain samples from blood `SRP059039`, `SRP059172`, `SRP062966`.

```
colon_proj <- c('SRP029880', 'SRP042228')
if(any(!file.exists(file.path(colon_proj, 'rse_gene.Rdata')))) {
    sapply(colon_proj, download_study)
}

blood_proj <- c('SRP059039', 'SRP059172', 'SRP062966')
if(any(!file.exists(file.path(blood_proj, 'rse_gene.Rdata')))) {
    sapply(blood_proj, download_study)
}
```

```
proj <- c(colon_proj,blood_proj)
```

Load the data

Now we load these data sets into R and calculate the number of genes and samples for each data set

```
dat <- lapply(proj, function(x) {
    load(file.path(x, 'rse_gene.Rdata'))
    return(rse_gene)
})
proj
```

```
## [1] "SRP029880" "SRP042228" "SRP059039" "SRP059172" "SRP062966"
```

```
sapply(dat, dim)
```

```
##       [,1]  [,2]  [,3]  [,4]  [,5]
## [1,] 23779 23779 23779 23779 23779
## [2,]    54   314   205   169   117
```

Load the metadata file

Now we load the metadata from the SRA samples

```
metadata <- all_metadata('sra')
```

```
## 2016-06-13 16:19:02 downloading the metadata to /var/folders/cx/n9s558kx6fb7jf5z_pgszgb80000gn/T//Rtm
```

Get additional geo data for these samples

Now we go through and collect geo information for the samples. We label them with their respective tissue and identify which samples are supposed to be normal.

```
if(!file.exists('charvec.Rdata')) {
    charvec <- vector('list', 5)
    dir.create('geoinfo', showWarnings = FALSE)
    for(i in 1:5){
      index <- match(colData(dat[[i]])$run, metadata$run)
      colData(dat[[i]])$geo <- metadata$geo_accession[index]
      info <- sapply(colData(dat[[i]])$geo, geo_info, destdir = 'geoinfo')
      charvec[[i]] <- sapply(info, geo_characteristics)
    }
    save(charvec, file = 'charvec.Rdata')
} else {
    load('charvec.Rdata')
}

## first data set - normals called 'normal-looking surrounding colonic epithelium'
colData(dat[[1]])$normal <- grepl('normal', unlist(charvec[1])[(1:54) * 2 - 1])
colData(dat[[1]])$tissue <- 'colon'

## second data set - normals called
colData(dat[[2]])$normal <- grepl('not ibd', tolower(unlist(charvec[[2]][5, ])))
colData(dat[[2]])$tissue <- 'colon'

## third data set  - normals called Control
colData(dat[[3]])$normal <- grepl('Control', unlist(charvec[[3]][2, ]))
colData(dat[[3]])$tissue <- 'blood'

## fourth data set  - normals called Control

colData(dat[[4]])$normal <- grepl('Control', unlist(charvec[[4]][1, ]))
colData(dat[[4]])$tissue <- 'blood'

## fifth data set - normals called healthy

colData(dat[[5]])$normal <- grepl('healthy', unlist(charvec[[5]][1, ]))
colData(dat[[5]])$tissue <- 'blood'
```

Merge the data sets

Now we merge the data sets into one ranged summarized experiment

```
mdat <- do.call(cbind, dat)
```

Get just the normals

Find out how many samples are normal in each study and subset to just the normal samples for further analysis.

```
table(colData(mdat)$normal, colData(mdat)$project)
```

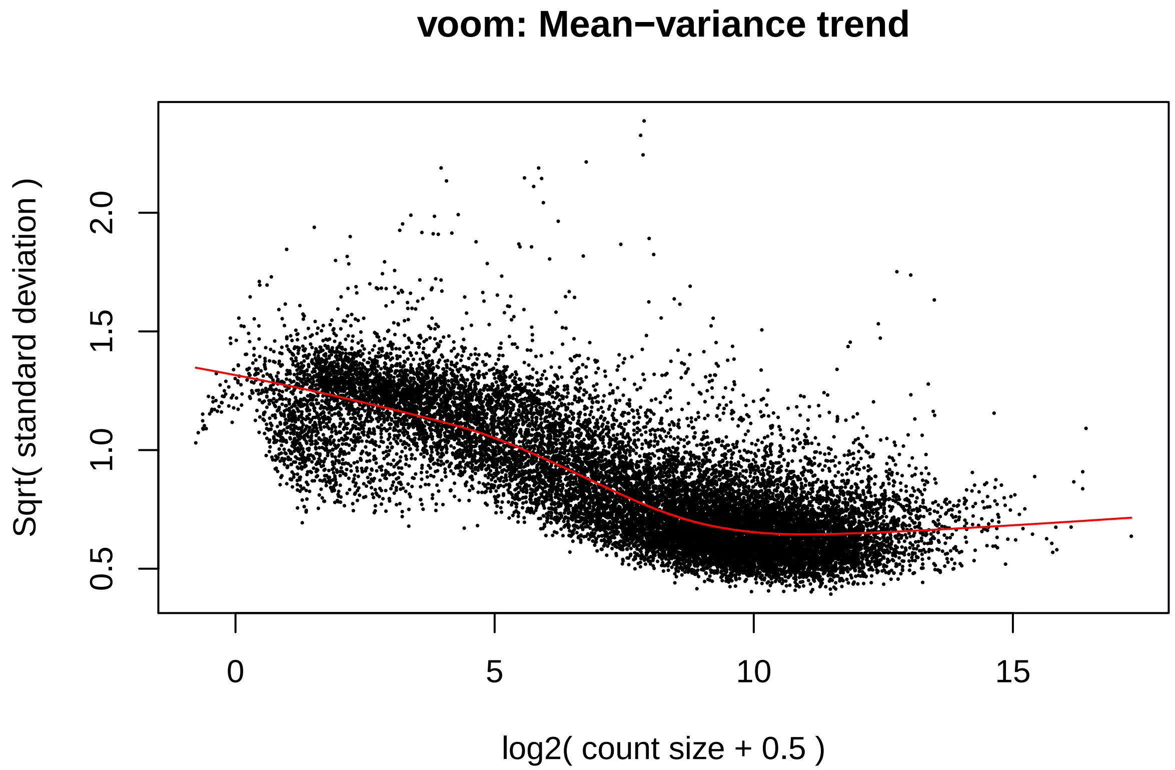
```
##
##         SRP029880 SRP042228 SRP059039 SRP059172 SRP062966
##   FALSE        35       273       181       122        99
##   TRUE         19        41        24        47        18
```

```
ndat <- mdat[, colData(mdat)$normal]
```

Do the analysis comparing tissue to

Here we do a differential expression analysis comparing blood to colon. We consider only the genes that have an average normalized count of at least 5 across the data set.

```
ndat <- scale_counts(ndat)
ndat_counts <- assays(ndat)[[1]]
keep <- rowMeans(ndat_counts) > 5
ndat_counts = ndat_counts[keep, ]
design <- model.matrix(~colData(ndat)$tissue)
dge <- DGEList(counts = ndat_counts)
dge <- calcNormFactors(dge)
v <- voom(dge, design, plot=TRUE)
```



```
fit <- lmFit(v, design)
fit <- eBayes(fit)
topTable(fit)
```

```
## Removing intercept from test coefficients
```

```
##             logFC    AveExpr         t      P.Value    adj.P.Val        B
## 7103    15.409396  0.4924358  79.43601 2.020597e-125 3.560898e-121 272.2428
## 10083   13.253543 -0.3745418  75.54867 3.368731e-122 2.968357e-118 265.4188
## 1909     7.783661 -2.5134139  75.11897 7.820641e-122 4.594105e-118 263.9306
## 56667   14.923360  0.6573342  73.53603 1.811337e-120 7.980298e-117 261.7541
## 25878   11.098638 -1.1683531  72.98350 5.507110e-120 1.941036e-116 260.6763
## 57381    8.703755 -1.8892183  70.64164 6.712078e-118 1.971449e-114 256.0142
## 133584   7.382474 -2.6339197  70.55688 8.008934e-118 2.016306e-114 255.2011
## 1441    -8.823323  8.1510433 -68.43403 7.140208e-116 1.143926e-112 254.6341
## 1015    14.208157  0.9677960  69.54431 6.710833e-117 1.314056e-113 254.2764
## 441094   6.910093 -2.8053456  69.62619 5.644920e-117 1.243505e-113 253.1423
```

GTEX analysis

Now we do the GTEX analysis comparing blood to colon.

```
## Download the GTEx data
if(!file.exists(file.path('SRP012682', 'rse_gene.Rdata'))) {
    download_study('SRP012682')
}
load(file.path('SRP012682', 'rse_gene.Rdata'))

gtex_metadata <- all_metadata('gtex')
```

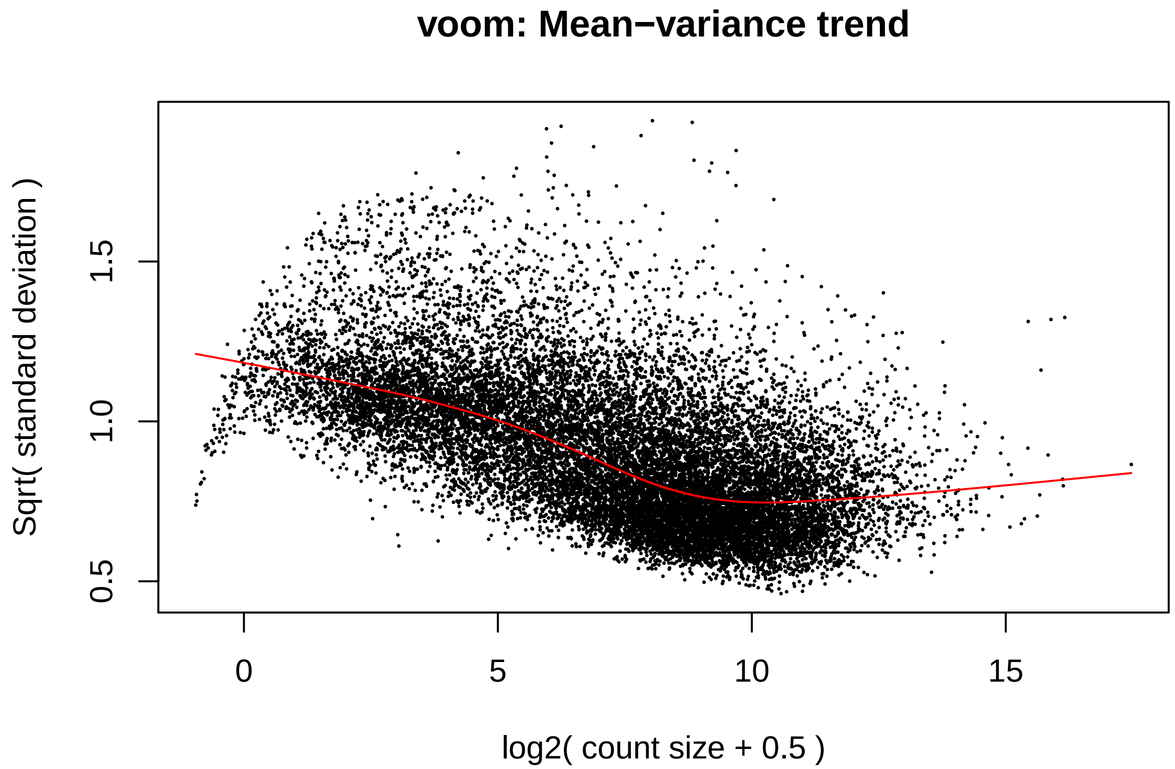
```
## 2016-06-13 16:19:31 downloading the metadata to /var/folders/cx/n9s558kx6fb7jf5z_pgszgb80000gn/T//Rtm
```

```
gtex_blood <- rse_gene[, subset(gtex_metadata, smtsd == 'Whole Blood')$run]
colData(gtex_blood)$tissue <- 'wholeblood'
gtex_colon <- rse_gene[, subset(gtex_metadata, smts == 'Colon')$run]
colData(gtex_colon)$tissue <- 'colon'

gtex_both <- do.call(cbind, list(gtex_blood, gtex_colon))
colData(gtex_both)$batch <- gtex_metadata[match(colData(gtex_both)$run,
    gtex_metadata$run), 'smgebtch']

gtex_both <- scale_counts(gtex_both)
gtex_both_counts <- assays(gtex_both)[[1]]
gtex_both_counts <- gtex_both_counts[keep, ]

design_gtex <- model.matrix(~colData(gtex_both)$tissue +
    colData(gtex_both)$batch)
dge_gtex <- DGEList(counts = gtex_both_counts)
dge_gtex <- calcNormFactors(dge_gtex)
v_gtex <- voom(dge_gtex, design_gtex, plot=TRUE)
```



```
fit_gtex <- lmFit(v_gtex, design_gtex)
fit_gtex <- eBayes(fit_gtex)
topTable(fit_gtex, coef = 2)
```

```
##            logFC  AveExpr         t P.Value adj.P.Val        B
## 6793    4.773601 6.779423  90.21488       0         0 874.9992
## 409     6.040177 7.723587  88.28947       0         0 861.2107
## 4542    7.490221 7.258075  84.59393       0         0 834.0351
## 4688    8.381934 6.579985  84.59066       0         0 833.9083
## 3936   7.980207 8.364888  84.02830       0         0 829.8006
## 1793  -5.244754 2.974639 -83.29415       0         0 823.9264
## 101    7.478347 6.054686  83.16901       0         0 823.2335
## 752    6.065599 7.218798  81.74749       0         0 812.4860
## 8514   4.617038 6.310878  80.81177       0         0 805.2694
## 51312  6.423430 7.944806  80.73634       0         0 804.6646
```

GTEX analysis

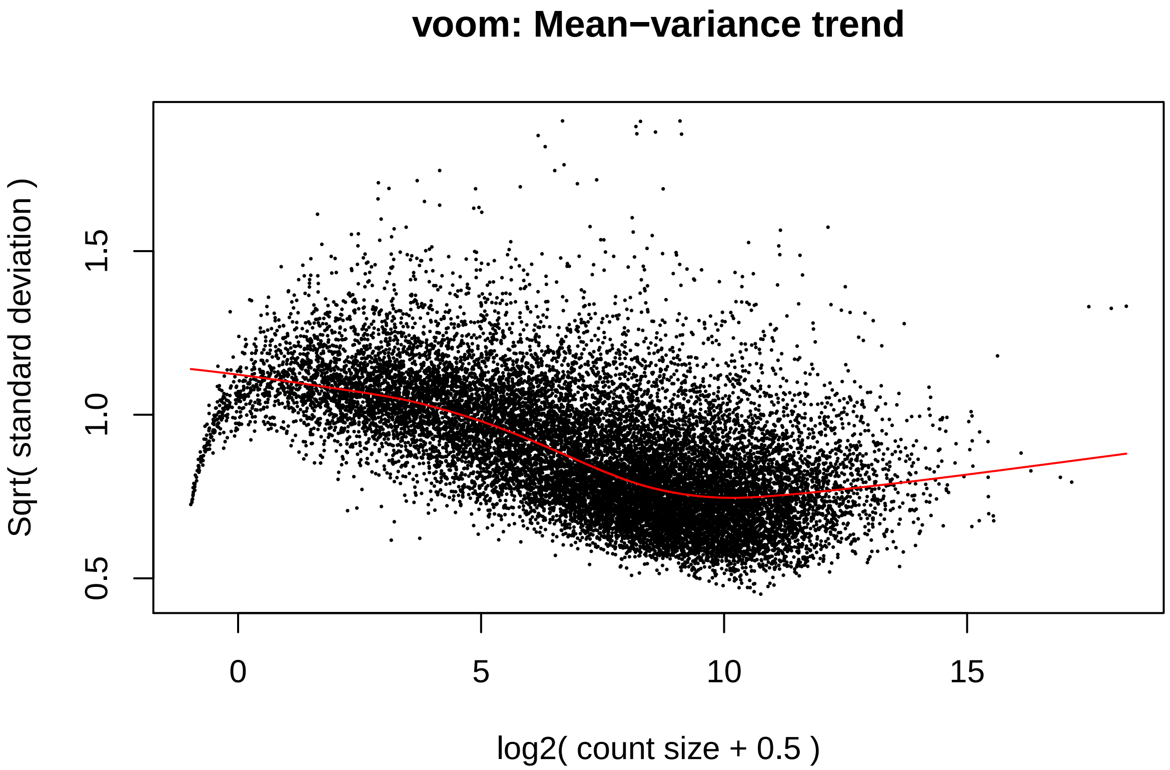
Now we do the GTEX analysis comparing blood to lung

```
gtex_lung <- rse_gene[, subset(gtex_metadata, smts=='Lung')$run]
colData(gtex_lung)$tissue <- 'lung'

gtex_both_lung <- do.call(cbind, list(gtex_blood, gtex_lung))
colData(gtex_both_lung)$batch <- gtex_metadata[
    match(
        colData(gtex_both_lung)$run,
        gtex_metadata$run
    ), 'smgebtch']

gtex_both_lung <- scale_counts(gtex_both_lung)
gtex_both_lung_counts <- assays(gtex_both_lung)[[1]]
gtex_both_lung_counts <- gtex_both_lung_counts[keep,]

design_gtex_lung <- model.matrix(~colData(gtex_both_lung)$tissue +
    colData(gtex_both_lung)$batch)
dge_gtex_lung <- DGEList(counts = gtex_both_lung_counts)
dge_gtex_lung <- calcNormFactors(dge_gtex_lung)
v_gtex_lung <- voom(dge_gtex_lung, design_gtex_lung, plot = TRUE)
```



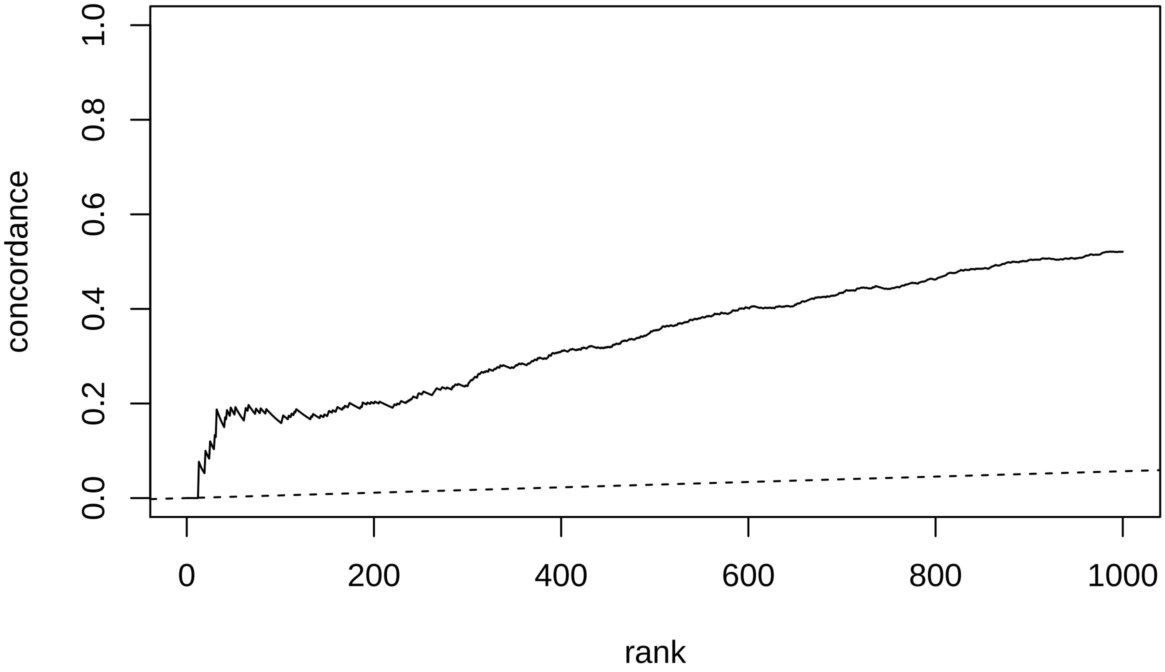
```
fit_gtex_lung <- lmFit(v_gtex_lung, design_gtex_lung)
fit_gtex_lung <- eBayes(fit_gtex_lung)
topTable(fit_gtex_lung, coef = 2)
```

```
##             logFC    AveExpr          t P.Value adj.P.Val        B
## 221395 -7.675466  3.4109146 -106.91629       0         0 996.3729
## 6943   -8.792146  1.6860275 -101.86151       0         0 964.1776
## 5754   -5.150188  2.7899033  -97.54954       0         0 935.9110
## 5420   -6.092697  4.0348220  -97.15255       0         0 933.2634
## 6909   -8.130718  2.9581907  -97.16373       0         0 933.1846
## 599    -2.835088  4.1595326  -96.50413       0         0 928.9989
## 10418  -6.396727  3.2092320  -94.48530       0         0 914.9644
## 9368    3.712945  6.1694693   94.43085       0         0 914.8108
## 207107 -8.226899 -0.5998609  -93.51892       0         0 908.0309
## 10160  -5.323208  3.0386054  -93.39156       0         0 907.3923
```

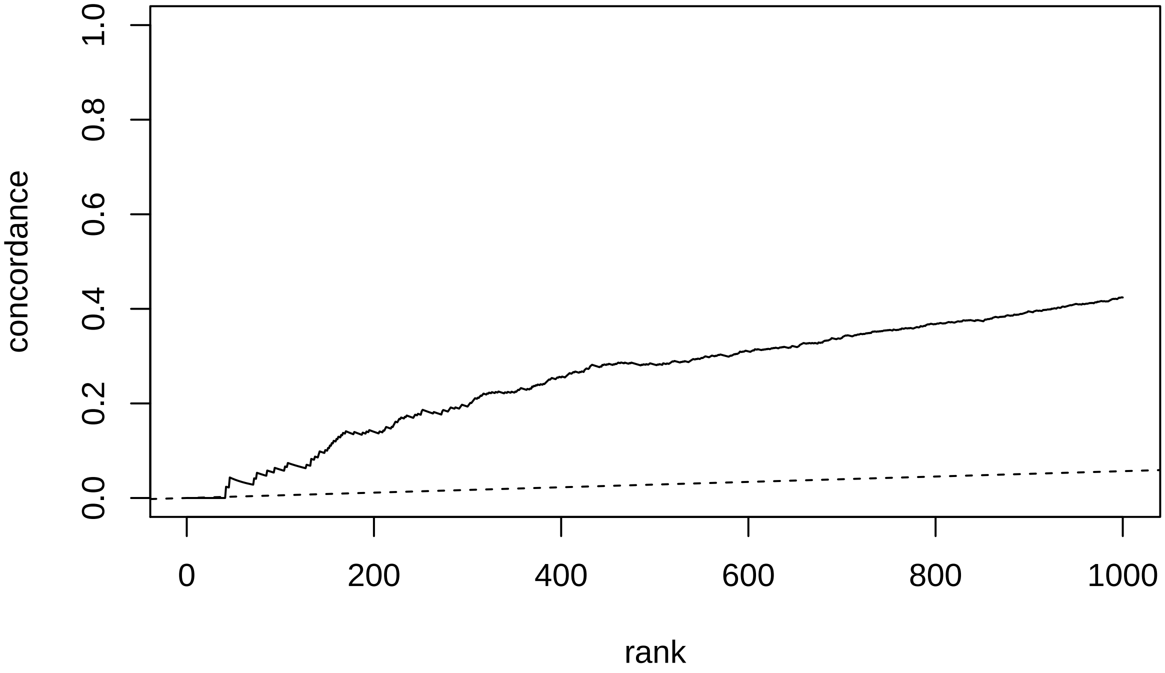
Compare CAT plots

Make CAT plots and compare different analyses.

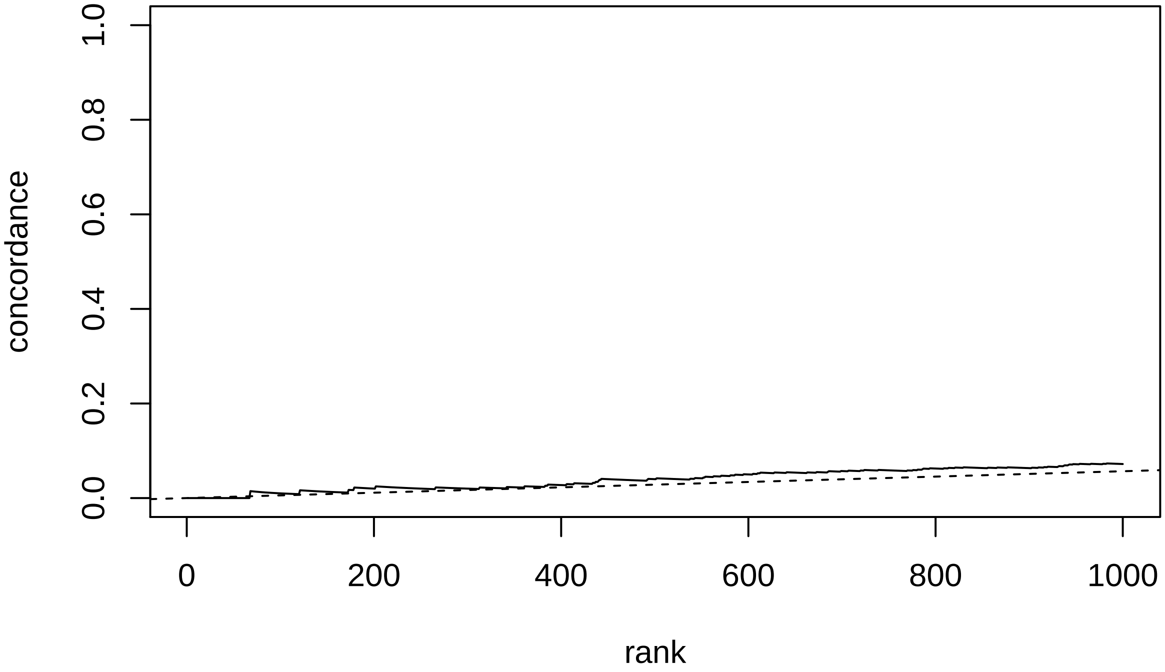
```
cat_sra_gtex <- CATplot(
    -rank(fit$coefficients[, 2]),
    -rank(-fit_gtex$coefficients[, 2]), maxrank = 1000, ylim = c(0,1))
```



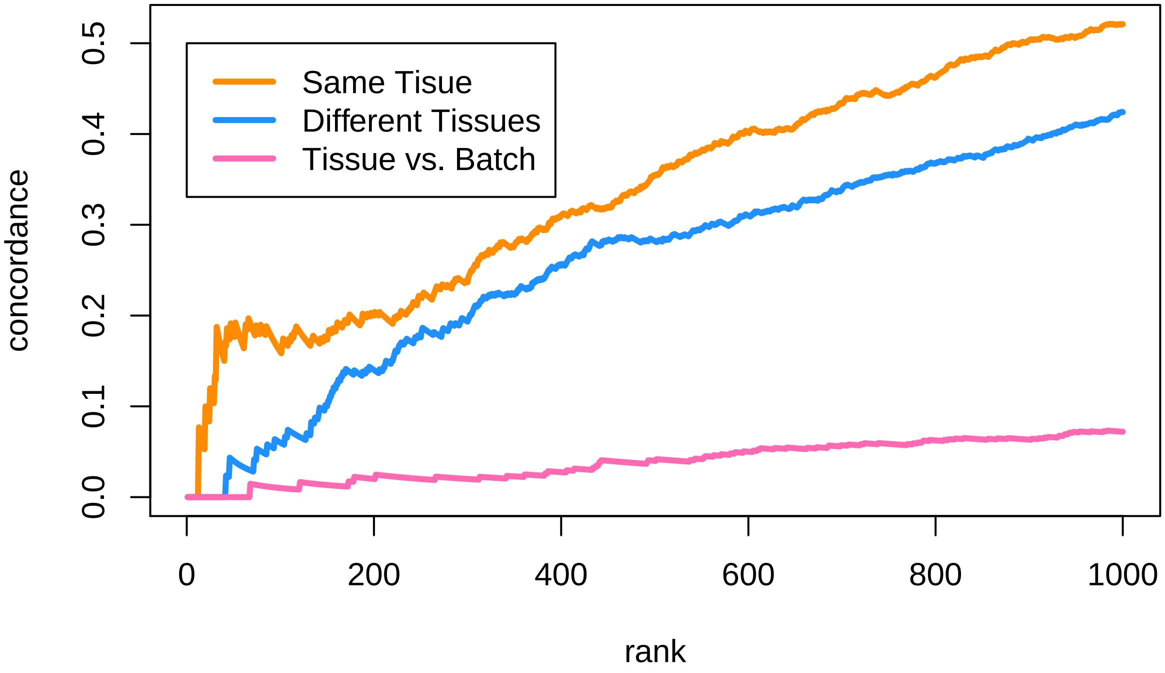
```
cat_sra_gtex_lung = CATplot(
    -rank(fit$coefficients[, 2]),
    -rank(-fit_gtex_lung$coefficients[, 2]), maxrank = 1000, ylim = c(0,1))
```



```
cat_sra_gtex_batch = CATplot(
    -rank(fit$coefficients[, 2]),
    -rank(-fit_gtex_lung$coefficients[, 3]), maxrank = 1000, ylim = c(0,1))
```



```r
plot(cat_sra_gtex, type = 'l', col = trop[1], lwd = 3)
lines(cat_sra_gtex_lung, type = 'l', col = trop[2], lwd = 3)
lines(cat_sra_gtex_batch, type = 'l', col = trop[3], lwd = 3)
legend(0, 0.5, legend=c('Same Tisue', 'Different Tissues',
    'Tissue vs. Batch'), col = trop[1:3], lwd = 3)
```



Reproducibility

This analysis report was made possible thanks to:

- R (R Core Team, 2016)
- *BiocStyle* (Oleś, Morgan, and Huber, 2016)
- *derfinder* (Collado-Torres, Nellore, Frazee, Wilks, et al., 2016)
- *devtools* (Wickham and Chang, 2016)
- *dplyr* (Wickham and Francois, 2015)
- *edgeR* (Robinson, McCarthy, and Smyth, 2010)
- *ffpe* (Waldron, L, Ogino, Shuji, Hoshida, Yujin, Shima, Kaori, et al., 2012)
- *knitcitations* (Boettiger, 2015)
- *magrittr* (Bache and Wickham, 2014)
- *recount* (Collado-Torres and Leek, 2016)
- *rmarkdown* (Allaire, Cheng, Xie, McPherson, et al., 2016)
- *RSkittleBrewer* (Frazee, 2016)
- *SummarizedExperiment* (Morgan, Obenchain, Hester, and Pagès, 2016)
- *limma* (Law, Chen, Shi, and Smyth, 2014)

Bibliography file

[1] J. Allaire, J. Cheng, Y. Xie, J. McPherson, et al. rmarkdown: Dynamic Documents for R. R package version 0.9.6. 2016. URL: https://CRAN.R-project.org/package=rmarkdown.

[2] S. M. Bache and H. Wickham. magrittr: A Forward-Pipe Operator for R. R package version 1.5. 2014. URL: https://CRAN.R-project.org/package=magrittr.

[3] C. Boettiger. knitcitations: Citations for 'Knitr' Markdown Files. R package version 1.0.7. 2015. URL: https://CRAN.R-project.org/package=knitcitations.

[4] L. Collado-Torres and J. T. Leek. recount: Explore and download data from the recount project. R package version 0.99.10. 2016. URL: https://github.com/leekgroup/recount.

[5] L. Collado-Torres, A. Nellore, A. C. Frazee, C. Wilks, et al. "Flexible expressed region analysis for RNA-seq with derfinder". In: bioRxiv (2016). DOI: 10.1101/015370. URL: http://biorxiv.org/content/early/2016/05/07/015370.

[6] A. Frazee. RSkittleBrewer: Fun with R Colors. R package version 1.1. 2016. URL: https://github.com/alyssafrazee/RSkittle

[7] C. Law, Y. Chen, W. Shi and G. Smyth. "Voom: precision weights unlock linear model analysis tools for RNA-seq read counts". In: Genome Biology 15 (2014), p. R29.

[8] M. Morgan, V. Obenchain, J. Hester and H. Pagès. SummarizedExperiment: SummarizedExperiment container. R package version 1.3.4. 2016.

[9] A. Oleś, M. Morgan and W. Huber. BiocStyle: Standard styles for vignettes and other Bioconductor documents. R package version 2.1.6. 2016. URL: https://github.com/Bioconductor/BiocStyle.

[10] R Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria, 2016. URL: https://www.R-project.org/.

[11] M. D. Robinson, D. J. McCarthy and G. K. Smyth. "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data". In: Bioinformatics 26 (2010), pp. -1.

[12] Waldron, L, Ogino, Shuji, Hoshida, Yujin, Shima, Kaori, et al. "Expression profiling of archival tumors for long-term health studies". In: Clinical Cancer Research 18.22 (2012). PMID: 23136189, pp. 6136–6146. DOI: 10.1158/1078-0432.CCR-12-1915.

[13] H. Wickham and W. Chang. devtools: Tools to Make Developing R Packages Easier. R package version 1.11.1. 2016. URL: https://CRAN.R-project.org/package=devtools.

[14] H. Wickham and R. Francois. dplyr: A Grammar of Data Manipulation. R package version 0.4.3. 2015. URL: https://CRAN.R-project.org/package=dplyr.

```r
## Time spent creating this report:
diff(c(timestart, Sys.time()))
```

```
## Time difference of 13.37282 mins
```

```
## Date this report was generated
message(Sys.time())
```

```
## 2016-06-13 16:32:23
```

```
## Reproducibility info
options(width = 120)
devtools::session_info()
```

```
## Session info -------------------------------------------------------------------------------------------

##  setting  value
##  version  R version 3.3.0 RC (2016-05-01 r70572)
##  system   x86_64, darwin13.4.0
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  tz       America/New_York
##  date     2016-06-13

## Packages -----------------------------------------------------------------------------------------------

##  package               * version  date       source
##  affy                    1.51.0   2016-05-27 Bioconductor
##  affyio                  1.43.0   2016-05-27 Bioconductor
##  annotate                1.51.0   2016-05-05 Bioconductor
##  AnnotationDbi           1.35.3   2016-05-27 Bioconductor
##  assertthat              0.1      2013-12-06 CRAN (R 3.3.0)
##  base64                  2.0      2016-05-10 CRAN (R 3.3.0)
##  beanplot                1.2      2014-09-19 CRAN (R 3.3.0)
##  bibtex                  0.4.0    2014-12-31 CRAN (R 3.3.0)
##  Biobase               * 2.33.0   2016-05-05 Bioconductor
##  BiocGenerics          * 0.19.1   2016-06-11 Bioconductor
##  BiocInstaller           1.23.4   2016-05-27 Bioconductor
##  BiocParallel            1.7.2    2016-05-20 Bioconductor
##  BiocStyle             * 2.1.6    2016-06-11 Bioconductor
##  biomaRt                 2.29.2   2016-05-30 Bioconductor
##  Biostrings              2.41.2   2016-06-08 Bioconductor
##  bitops                  1.0-6    2013-08-17 CRAN (R 3.3.0)
##  bumphunter              1.13.0   2016-05-05 Bioconductor
##  chron                   2.3-47   2015-06-24 CRAN (R 3.3.0)
##  codetools               0.2-14   2015-07-15 CRAN (R 3.3.0)
##  colorout              * 1.1-2    2016-05-05 Github (jalvesaq/colorout@6538970)
##  colorspace              1.2-6    2015-03-11 CRAN (R 3.3.0)
##  data.table              1.9.6    2015-09-19 CRAN (R 3.3.0)
##  DBI                     0.4-1    2016-05-08 CRAN (R 3.3.0)
##  devtools              * 1.11.1   2016-04-21 CRAN (R 3.3.0)
##  digest                  0.6.9    2016-01-08 CRAN (R 3.3.0)
##  doRNG                   1.6      2014-03-07 CRAN (R 3.3.0)
##  dplyr                 * 0.4.3    2015-09-01 CRAN (R 3.3.0)
##  edgeR                 * 3.15.0   2016-05-27 Bioconductor
##  evaluate                0.9      2016-04-29 CRAN (R 3.3.0)
##  ffpe                  * 1.17.0   2016-05-27 Bioconductor
##  foreach                 1.4.3    2015-10-13 CRAN (R 3.3.0)
##  formatR                 1.4      2016-05-09 CRAN (R 3.3.0)
##  genefilter             1.55.2   2016-05-27 Bioconductor
##  GenomeInfoDb         * 1.9.1    2016-05-13 Bioconductor
##  GenomicAlignments      1.9.2    2016-06-13 Bioconductor
##  GenomicFeatures        1.25.12  2016-05-21 Bioconductor
##  GenomicRanges        * 1.25.4   2016-06-10 Bioconductor
##  GEOquery               2.39.3   2016-05-20 Bioconductor
##  htmltools              0.3.5    2016-03-21 CRAN (R 3.3.0)
##  httr                   1.1.0    2016-01-28 CRAN (R 3.3.0)
##  illuminaio             0.15.0   2016-05-27 Bioconductor
##  IRanges              * 2.7.6    2016-06-10 Bioconductor
##  iterators              1.0.8    2015-10-13 CRAN (R 3.3.0)
##  KernSmooth             2.23-15  2015-06-29 CRAN (R 3.3.0)
##  knitcitations        * 1.0.7    2015-10-28 CRAN (R 3.3.0)
##  knitr                  1.13     2016-05-09 CRAN (R 3.3.0)
##  lattice                0.20-33  2015-07-14 CRAN (R 3.3.0)
##  limma                * 3.29.7   2016-06-13 Bioconductor
##  locfit                 1.5-9.1  2013-04-20 CRAN (R 3.3.0)
##  lubridate              1.5.6    2016-04-06 CRAN (R 3.3.0)
##  lumi                   2.25.0   2016-05-27 Bioconductor
##  magrittr             * 1.5      2014-11-22 CRAN (R 3.3.0)
##  MASS                   7.3-45   2016-04-21 CRAN (R 3.3.0)
##  Matrix                 1.2-6    2016-05-02 CRAN (R 3.3.0)
##  matrixStats            0.50.2   2016-04-24 CRAN (R 3.3.0)
##  mclust                 5.2      2016-03-31 CRAN (R 3.3.0)
##  memoise                1.0.0    2016-01-29 CRAN (R 3.3.0)
##  methylumi              2.19.3   2016-06-03 Bioconductor
##  mgcv                   1.8-12   2016-03-03 CRAN (R 3.3.0)
##  minfi                  1.19.2   2016-05-27 Bioconductor
##  multtest               2.29.0   2016-05-27 Bioconductor
##  nleqslv                3.0.1    2016-05-02 CRAN (R 3.3.0)
##  nlme                   3.1-128  2016-05-10 CRAN (R 3.3.0)
##  nor1mix                1.2-1    2015-07-27 CRAN (R 3.3.0)
##  openssl                0.9.3    2016-05-04 CRAN (R 3.3.0)
##  pkgmaker               0.22     2014-05-14 CRAN (R 3.3.0)
##  plyr                   1.8.3    2015-06-12 CRAN (R 3.3.0)
##  preprocessCore         1.35.0   2016-05-27 Bioconductor
##  quadprog               1.5-5    2013-04-17 CRAN (R 3.3.0)
##  R6                     2.1.2    2016-01-26 CRAN (R 3.3.0)
##  RColorBrewer           1.1-2    2014-12-07 CRAN (R 3.3.0)
##  Rcpp                   0.12.5   2016-05-14 CRAN (R 3.3.0)
##  RCurl                  1.95-4.8 2016-03-01 CRAN (R 3.3.0)
##  recount              * 0.99.10  2016-06-12 Github (leekgroup/recount@7a7ea73)
##  RefManageR             0.10.13  2016-04-04 CRAN (R 3.3.0)
##  registry               0.3      2015-07-08 CRAN (R 3.3.0)
##  reshape                0.8.5    2014-04-23 CRAN (R 3.3.0)
##  RJSONIO                1.3-0    2014-07-28 CRAN (R 3.3.0)
##  rmarkdown            * 0.9.6    2016-05-01 CRAN (R 3.3.0)
##  rngtools               1.2.4    2014-03-06 CRAN (R 3.3.0)
##  Rsamtools              1.25.0   2016-05-05 Bioconductor
##  RSkittleBrewer       * 1.1      2016-06-13 Github (alyssafrazee/RSkittleBrewer@230d1d0)
##  RSQLite                1.0.0    2014-10-25 CRAN (R 3.3.0)
##  rstudioapi             0.5      2016-01-24 CRAN (R 3.3.0)
##  rtracklayer            1.33.5   2016-06-13 Bioconductor
##  S4Vectors            * 0.11.4   2016-06-11 Bioconductor
##  sfsmisc                1.1-0    2016-02-23 CRAN (R 3.3.0)
##  siggenes               1.47.0   2016-05-27 Bioconductor
##  stringi                1.0-1    2015-10-22 CRAN (R 3.3.0)
##  stringr                1.0.0    2015-04-30 CRAN (R 3.3.0)
##  SummarizedExperiment * 1.3.4    2016-06-10 Bioconductor
##  survival               2.39-4   2016-05-11 CRAN (R 3.3.0)
##  TTR                  * 0.23-1   2016-03-21 CRAN (R 3.3.0)
##  withr                  1.0.1    2016-02-04 CRAN (R 3.3.0)
##  XML                    3.98-1.4 2016-03-01 CRAN (R 3.3.0)
##  xtable                 1.8-2    2016-02-05 CRAN (R 3.3.0)
##  xts                    0.9-7    2014-01-02 CRAN (R 3.3.0)
##  XVector                0.13.0   2016-05-05 Bioconductor
##  yaml                   2.1.13   2014-06-12 CRAN (R 3.3.0)
##  zlibbioc               1.19.0   2016-05-05 Bioconductor
##  zoo                    1.7-13   2016-05-03 CRAN (R 3.3.0)
```