powsim: Power Analysis and Sample Size Estimation for Bulk and Single Cell RNA-Seq Experiments

Beate Vieth *

March 10, 2017

Report issues on https://github.com/bvieth/powsim/issues

*vieth@bio.lmu.de

Contents

1 Installation guide

Powsim has a number of dependencies that need to be installed before hand (see also the README file on github).

```
ipak <- function(pkg, repository = c("CRAN", "Bioconductor",
    "github")) {
    new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
    if (length(new.pkg)) {
        if (repository == "CRAN") {
            install.packages(new.pkg, dependencies = TRUE)
        }
        if (repository == "Bioconductor") {
            source("https://bioconductor.org/biocLite.R")
            biocLite(new.pkg, dependencies = TRUE, ask = FALSE)
        }
        if (repository == "github") {
            devtools::install_github(pkg, build_vignettes = FALSE)
        }
    }
}

# CRAN PACKAGES
cranpackages <- c("gamlss.dist", "methods", "stats", "moments",
    "doParallel", "parallel", "reshape2", "dplyr", "tidyr", "data.table",
    "ggplot2", "ggthemes", "ggExtra", "cowplot", "scales", "fitdistrplus",
    "MASS", "pscl", "nonnest2", "cobs", "msir", "drc", "devtools",
    "XML")
# Note: To install msir, the dependency rgl needs to be
# installed (apt-get install libx11-dev mesa-common-dev
# libglu1-mesa-dev).  Note to install XML, xml2 config is
# needed (apt-get install libxml2-dev).
ipak(cranpackages, repository = "CRAN")

# BIOCONDUCTOR
biocpackages <- c("S4Vectors", "AnnotationDbi", "Biobase", "BiocParallel",
    "scater", "scran", "edgeR", "limma", "DESeq2", "baySeq",
    "NOISeq", "EBSeq", "DSS", "MAST", "ROTS", "IHW", "qvalue")
ipak(biocpackages, repository = "Bioconductor")

# GITHUB Note: to install scde cairo and x11 are needed
# (apt-get install libcairo2-dev, apt-get install libxt-dev).
githubpackages <- c("gu-mi/NBGOF", "hms-dbmi/scde", "nghiavtr/BPSC")
ipak(githubpackages, repository = "github")
devtools::install_github("kdkorthauer/scDD", build_vignettes = FALSE,
```

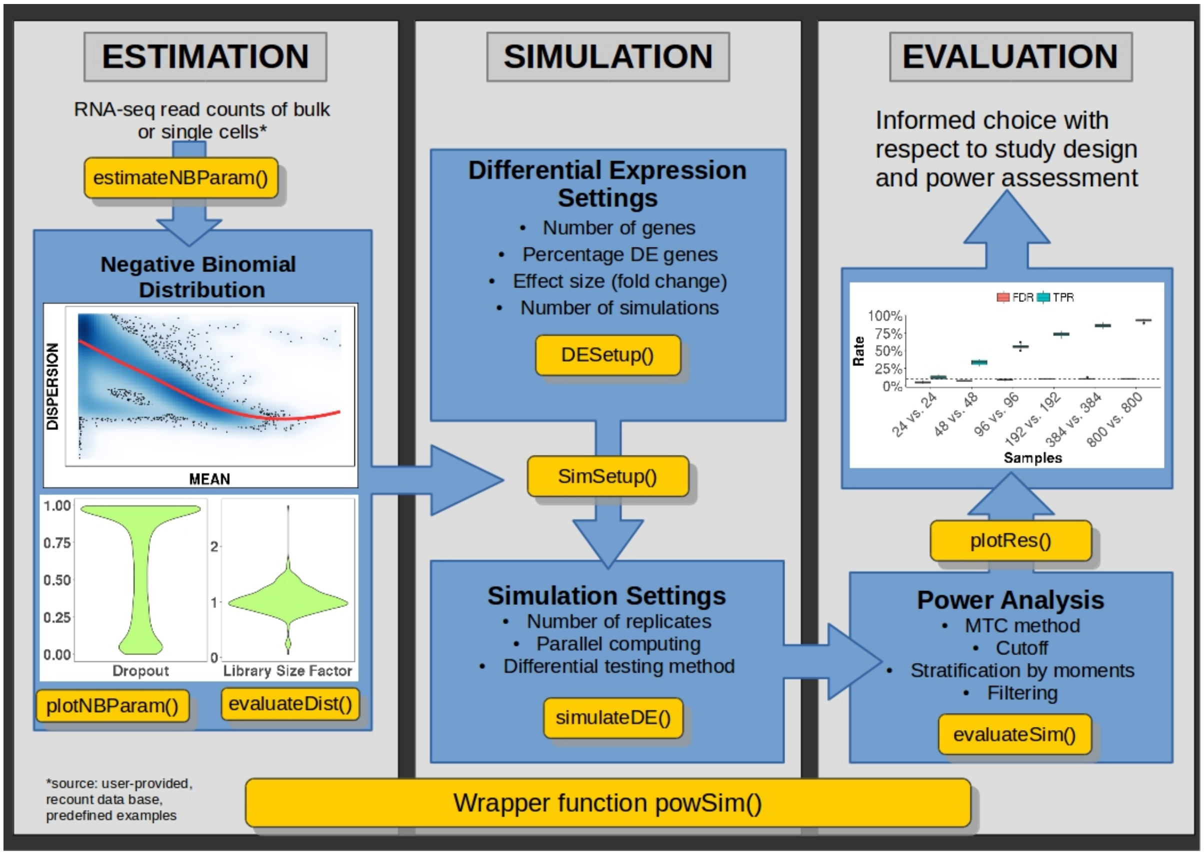

Figure 1: **Powsim schematic overview.** (A) Estimation: (B) Simulation: (C) Evaluation. Functions given in orange.

```
    ref = "develop")
```

2 Introduction

In this vignette, we illustrate the features of *powsim* by assessing the power to detect differential expression between two groups of embryonic stem cells cultured in standard 2i medium (E-MTAB-2600) [1].

3 powsim workflow

The basic workflow of *powsim* is illustrated in figure 1.

3.1 Parameter estimation

The parameters of the negative binomial distribution, i.e. mean and dispersion are estimated by the function `estimateNBParam`. In addition, the dropout probability, i.e. the fraction of zero counts per gene, is calculated. The user can choose between three estimation frameworks:

edgeR *edgeR* [2].
DESeq2 *DESeq2* [3].
MatchMoments Matching moments estimation of mean and dispersion based on normalized counts.

As described in edgeR and DESeq2, their frameworks are able to handle experimental setup information. For example, the user can specify batches and other characteristics of the samples.

The library size normalisation methods are trimmed mean of M-values for edgeR and median-of-ratios method for DESeq2 and MatchMoments for bulk RNA-seq data. For single cell RNA-seq data, the deconvolution method `computeSumFactors` in *scran* is used [4].

The estimates, sequencing depth and normalisation factors are plotted with `plotNBParam`.

With the following command, we estimate and plot the parameters for the embryonic stem cells cultured in standard 2i+lif medium (Kolodziejczyk) (figure 2). As expected for single cell RNA-seq, the variability (i.e. dispersion) and dropout rates are high. Furthermore, the dispersion strongly depends on the mean and does not level off with higher mean values.

```
# download count table
Kolodziejczyk.cnts <- getURL()
TwoiLIF.params = estimateNBParam(countData = Kolodziejczyk.cnts,
    cData = NULL, design = NULL, RNAseq = "singlecell", estFramework = "MatchMoments",
    sigma = 1.96)
plotNBParam(TwoiLif.params)
```

We have implemented a read count simulation framework assuming an underlying negative binomial distribution. To predict the dispersion given a random draw of mean expression value observed, we apply a locally weighted polynomial regression fit. To capture the variability of dispersion estimates observed, a local variability prediction band (sigma=1.96) is applied. For bulk RNA-seq experiments, dropouts are less probable but can still occur. To include this phenomenon we sample from the observed dropout rates for genes that have a mean expression value below 5% dropout probability determined by a decrease constrained B-splines regresssion of dropout rate against mean expression (`cobs` in *cobs*).

The resulting read count matrix has similar distributional characteristics as the original Kolodziejczyk data set (figure 3).

4 Simulations

For simulating differential expression between two groups, the number of genes, number of simulations, percentage of differential expression and effect size are set up with the function `DESetup`. The effect size is here defined as the log2 fold change which can be a constant, sampled from a vector or function. The uniform, normal and gamma distributions are possible options and illustrated in figure **??**. Depending on the settings, these distribution can be broader or narrower. If using this option, we recommend to choose a distribution that closely resembles previously observed or expected fold changes.

The distribution estimates and these settings are then combined to one object with `SimSetup`. This allows the user to assess power of multiple groupwise comparisons and different differential testing methods. The following command sets up simulations with 10,000 genes, 20% genes being DE, log fold change sample from a narrow gamma distribution and parameter estimates based on Kolodziejczyk data:

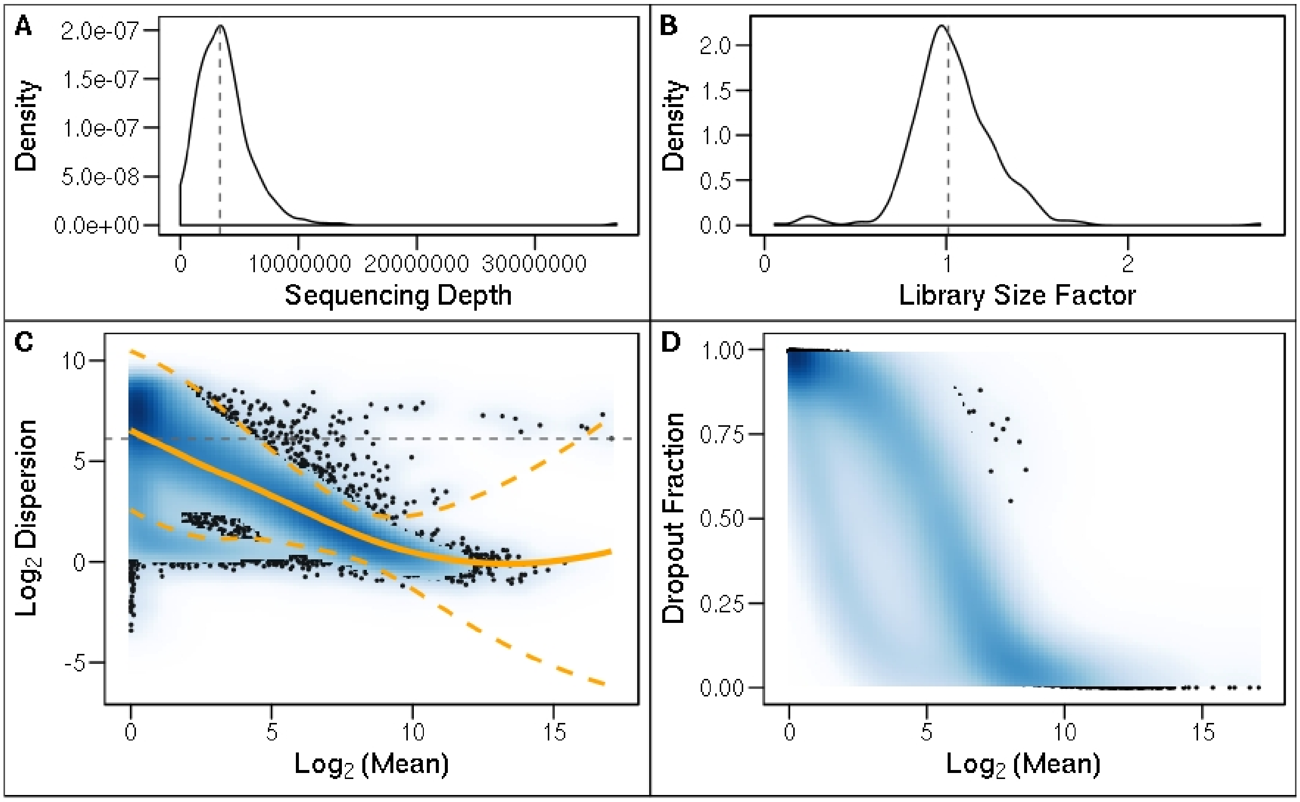

Figure 2: **Estimated parameters for Kolodziejczyk data set.** A) Sequencing depth per sample with median sequencing depth (grey dashed line). B) Library size normalisation factor per sample with median size factor (grey dashed line). C) Local polynomial regression fit between mean and dispersion estimates with variability band per gene (yellow). Common dispersion estimate (grey dashed line). D) Fraction of dropouts versus estimated mean expression per gene.

```
lfc.gamma = function(x) sample(c(-1, 1), size = x, replace = T) *
    rgamma(x, 3, 3)
de.opts = DESetup(ngenes = 10000, nsims = 25, p.DE = 0.2, LFC = lfc.gamma)
sim.opts = SimSetup(desetup = de.opts, params = TwoiLIF.params,
    size.factors = "given")
```

With the setup defined, the differential expression simulation is run with (simulateDE). For this, the user needs to set the following options:

Replicates The number of sample replicates per group (n1 and n2). These can be unbalanced.

DEmethod The differential testing method. The user can choose between methods in total. developed for bulk, xx developed for single cells. A detailed list can be found under point xx.

ncores A number of DE methods are able to run in parallel too speed up differential testing.

```
simDE = simulateDE(n1 = c(24, 48, 96, 192, 384, 800), n2 = c(24,
    48, 96, 192, 384, 800), sim.settings = sim.opts, ncores = 1,
    DEmethod = "MAST", verbose = T)
```

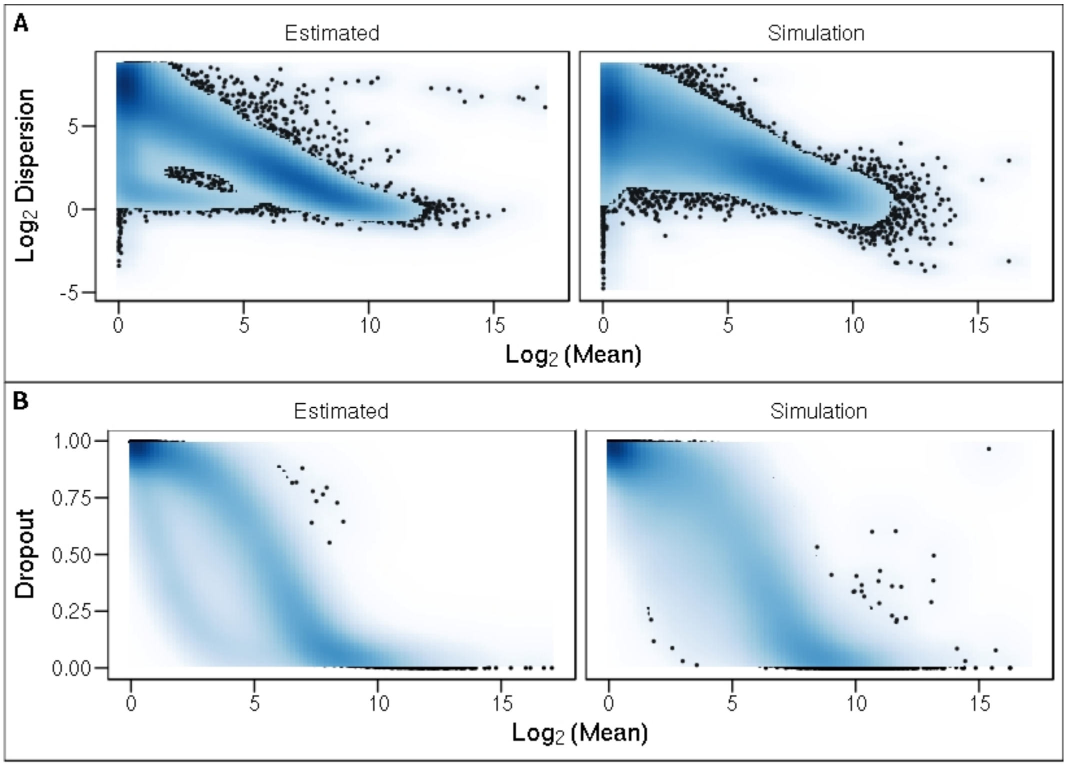

Figure 3: **Comparison of estimated and simulated read counts** (A) Dispersion versus Mean. (B) Dropout versus Mean.

4.1 Evaluation

The results of differential expression simulation are evaluated with `evaluateSim`. We have separated the evaluation from DE detection to allow the user to evaluate power in a comprehensive way as advocated by [5]. In this function, the proporations and error rates are estimated. The rates can be stratified by mean, dispersion or dropout. Furthermore, the user can choose between different multiple testing correction methods (see `p.adjust.methods`, `ihw` in *IHW* and `qvalue` in *qvalue*). Also, the genes can be filtered by mean, dispersion or dropout. To define biologically interesting genes, a cutoff for the log2 fold change with delta can be set.

With the following command we evaluate the marginal TPR and TPR conditional on the mean expression for the simulation based on Kolodziejczyk data.

```
evalDE = evaluateSim(simRes = simDE, alpha.type = "adjusted",
    MTC = "FDR", alpha.nominal = 0.1, stratify.by = "mean", filter.by = "none",
    strata.filtered = 0, target.by = "lfc", delta = 0)
```

The results of the evaluation can be plotted with `plotEvalRes`.

rate Marginal or Conditional Error Rates calculations. The conditional error rates are determined and calculated with `evaluateSim`. The number of genes per stratum are also summarised.

quick If this is set to *R*TRUE then only the TPR and FDR will be plotted.

With the following commands, the quick marginal and conditional power assessment for the Kolodziejczyk data is plotted.

```
plotEvalRes(evalRes = evalDE, rate = "marginal", quick = TRUE,
    annot = TRUE)

plotEvalRes(evalRes = evalDE, rate = "stratified", quick = TRUE,
    annot = TRUE)
```

5 Additional Functionalities

5.1 Evaluate Simulation Framework

It is important to validate the appropiateness of the chosen simulation framework. The function `evaluateDist` compares the theoretical fit of the poisson, negative binomial, zero-inflated poisson and zero-inflated negative binomial and Beta-Poisson distribution to the empirical RNA-seq read counts ([6], , [7], [8]). The evaluation is then summarized with the function `summariseDist` which chooses the best fitting distribution per gene based on goodness-of-fit statistics (Chi-square test), Akaike Information Criterium and comparing observed dropouts with zero count prediction of the models. Bulk RNA-seq experiments are usually conducted with a small number of samples. We therefore recommend to rely on the goodness-of-fit validation by [9]. To use this approach in `evaluateDist`, the user should allow for permutation simulations by setting the value of nsims to at least 100. If available, the computation can be run on multiple cores by setting the number of cores (ncores).

With the following command, we estimate and plot the parameters for the embryonic stem cells cultured in standard 2i lif medium (Kolodziejczyk).

```
TwoiLIF.distfit = evaluateDist(cnts = Kolodziejczyk.cnts, RNAseq = "singlecell",
    ncores = 1, nsims = 1, frac.genes = 1, min.meancount = 1,
    min.libsize = 1000)

plotEvalDist(evalDist = TwoiLIF.distfit, annot = F)
```

5.2 Negative Binomial Parameters

5.2.1 in silico Parameter Definition

We have also implemented the option to approximate the read count matrix simulation based on random distribution functions in *R*. The user then has to define the mean, dispersion, dropout and library size in `insilicoNBParam`. In the absence of suitable pilot studies, For example, a typical single cell RNA-seq experiment could be approximated with:

- mean
- dispersion
- library size

The same functionality can also be used for bulk RNA-seq.

5.2.2 Precalculated Parameter Estimates

We have included precalculated parameter estimates for bulk as well as single cell RNA-seq experiments. The data sets included represent different levels of mean, dispersion and dropout characteristics. The user can view these estimates with `plotNBParam` and use it as an input for `SimSetup`.

5.3 Simulation settings

By default, there is no difference in library sizes between the samples. If the user wishes for a more realistic, i.e. more variable distribution of read counts across samples, the library sizes can be sampled from observed, vector or function.

6 Wrapper Function

7 Session info

Here is the output of `sessionInfo` on the system on which this document was compiled:

- R version 3.3.2 (2016-10-31), `x86_64-pc-linux-gnu`
- Locale: `LC_CTYPE=en_GB.UTF-8`, `LC_NUMERIC=C`, `LC_TIME=de_DE.UTF-8`, `LC_COLLATE=en_GB.UTF-8`, `LC_MONETARY=de_DE.UTF-8`, `LC_MESSAGES=en_GB.UTF-8`, `LC_PAPER=de_DE.UTF-8`, `LC_NAME=C`, `LC_ADDRESS=C`, `LC_TELEPHONE=C`, `LC_MEASUREMENT=de_DE.UTF-8`, `LC_IDENTIFICATION=C`
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: cowplot 0.7.0, dplyr 0.5.0, ggplot2 2.2.1, knitr 1.15.1, tidyr 0.6.1, xtable 1.8-2
- Loaded via a namespace (and not attached): assertthat 0.1, BiocStyle 2.2.1, colorspace 1.3-2, DBI 0.6, evaluate 0.10, formatR 1.4, grid 3.3.2, gtable 0.2.0, highr 0.6, lazyeval 0.2.0, magrittr 1.5, munsell 0.4.3, plyr 1.8.4, R6 2.2.0, Rcpp 0.12.9, scales 0.4.1, stringi 1.1.2, stringr 1.2.0, tibble 1.2, tools 3.3.2

References

[1] Aleksandra A Kolodziejczyk, Jong Kyoung Kim, Jason C H Tsang, Tomislav Ilicic, Johan Henriksson, Kedar N Natarajan, Alex C Tuck, Xuefei Gao, Marc Bühler, Pentao Liu, John C Marioni, and Sarah A Teichmann. Single cell RNA-Sequencing of pluripotent states unlocks modular transcriptional variation. *Cell Stem Cell*, 17(4):471–485, 1 October 2015. doi:10.1016/j.stem.2015.09.011.

[2] Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. edger: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140, 1 January 2010. doi:10.1093/bioinformatics/btp616.

[3] Michael I Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol.*, 15(12):550, 2014. doi:10.1186/s13059-014-0550-8.

[4] Aaron T L Lun, Karsten Bach, and John C Marioni. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. *Genome Biol.*, 17:75, 27 April 2016. doi:10.1186/s13059-016-0947-7.

[5] Hao Wu, Chi Wang, and Zhijin Wu. PROPER: comprehensive power evaluation for differential expression using RNA-seq. *Bioinformatics*, 31(2):233–241, 15 January 2015. doi:10.1093/bioinformatics/btu640.

[6] A Colin Cameron and Pravin K Trivedi. *Regression Analysis of Count Data (Econometric Society Monographs)*. Cambridge University Press, 2 edition edition, 27 May 2013.

[7] Jong Kyoung Kim and John C Marioni. Inferring the kinetics of stochastic gene expression from single-cell RNA-sequencing data. *Genome Biol.*, 14(1):R7, 28 January 2013. doi:10.1186/gb-2013-14-1-r7.

[8] Mihails Delmans and Martin Hemberg. Discrete distributional differential expression (D3E)–a tool for gene expression analysis of single-cell RNA-seq data. *BMC Bioinformatics*, 17:110, 29 February 2016. doi:10.1186/s12859-016-0944-6.

[9] Gu Mi, Yanming Di, and Daniel W Schafer. Goodness-of-fit tests and model diagnostics for negative binomial regression of RNA sequencing data. *PLoS One*, 10(3):e0119254, 18 March 2015. doi:10.1371/journal.pone.0119254.