**CompareDEtools (v1.0)**

- *User’s Manual* –

Maintainer: BuKyung Baik <[back829@unist.ac.kr](mailto:back829@unist.ac.kr)>

Author: Sora Yoon<[yoonsora1@unist.ac.kr](mailto:yoonsora1@unist.ac.kr)> &

BuKyung Baik <[back829@unist.ac.kr](mailto:back829@unist.ac.kr)>

Last updated: 2019. 6. 05.

**1. Introduction**

CompareDEtools is an R package that has been developed to compare the performance of 14 existing differential expression (DE) analysis tools based on the simulation or real RNA-seq data.

The comparable methods include baySeq, DESeq, DESeq2, edgeR, limma, PoissonSeq, ROTS and SAMSeq. With this package, the users can simulate RNA-seq data for various conditions such as the presence of ourliers, the size of gene expression levels and dispersions. The simulation dataset is generated based on the mean and dispersion values obtained from real data such as TCGA KIRC, Bottomly or SEQC dataset. The users can generate performance boxplots, heatmaps and PCA plots for the analysis. The installation and usage of comareDEtools is described below.

1. **Package Installation**
2. Open R program.
3. Type following commands in R console

|  |
| --- |
| * + - install.packages('devtools')     - library(devtools)     - install.packages('BiocManager')     - BiocManager::install(c('baySeq','Biobase','compcodeR','DESeq','DESeq2','edgeR,','impute','limma','ROTS'))     - install.packages('gplots','gtools','ggplot2','PoissonSeq','reshape','RColorBrewer','ROCR','samr','SimSeq','statmod','XML')     - install\_github('unistbig/compareDEtools') |

1. **Run**

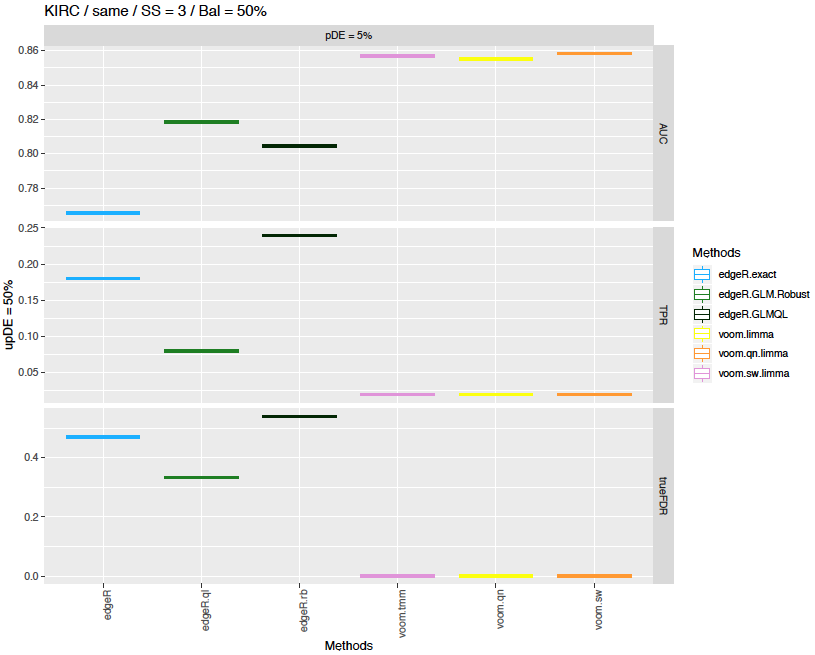
The whole process consists of four simple steps.

1. Load DE R package.
2. Generate simulation data
3. Run simulation
4. Draw result plots

***Toy Example1 Synthetic data analysis***

|  |
| --- |
| # Load library  >> library(compareDEtools)  # Assign directories to store simulation datasets, analysis results and result plots  >> dataset.dir='~/Dataset/'  >> analysis.dir='~/Analysis/'  >> figure.dir='~/Fig/'  # Assign comparison methods  >> AnalysisMethods=c('edgeR','edgeR.ql','edgeR.rb','voom.tmm','voom.qn','voom.sw')  #Generate RNA-seq simulation dataset  >> GenerateSyntheticSimulation(working.dir=dataset.dir, data.types='KIRC', rep=1, nsample=c(3), nvar=1000, nDE=c(50), fraction.upregulated = 0.5, disp.Types = 'same', modes=c('D'))  #Run DE methods for the simulation datasets  >> runSimulationAnalysis(working.dir=dataset.dir, output.dir=analysis.dir, real=FALSE, data.types='KIRC', rep=1, nsample=c(3), nDE=c(50), fraction.upregulated=0.5, disp.Types='same', modes=c('D'), AnalysisMethods = AnalysisMethods, para=list())  # Draw boxplots for comparing the performances of DE methods  >> performance\_plot(working.dir=analysis.dir,figure.dir=figure.dir,fixedfold=F,simul.data='KIRC', rep=1, nsample=c(3), nvar=1000, nDE=c(50), fraction.upregulated = 0.5, disp.Type = 'same', mode='D', AnalysisMethods=AnalysisMethods, rowType = c('AUC','TPR','trueFDR')) |

When you check the figure directory (figure.dir), the resulting plot will be displayed as below.



Now, let’s take a look at each function used above.

**GenerateSyntheticSimulation**

This function generates RNA-seq simulation datasets with given parameters. It takes ten arguments as follows.

* working.dir: A directory where simulation datasets are stored (type: character)
* data.types: Data types that determines the scale of gene expression and dispersion. There are four possible values such as,
  + KIRC : for large gene expression and dispersion
  + Bottomly : for small gene expression and dispersion
  + mBdK : for small gene expression and large dispersion
  + mKdB : for large gene expression and small dispersion

Here, dispersion indicates gene expression dispersion among samples.

(type: character vector)

* fixedfold: True or False. If set False (default), the gene expression fold change between two samples will be set as + random number with exponential distribution. Here, is determined based on the sample size *N*.

If set True, DE genes are split three groups and fixed fold changes (0.625, 1.15, 1.3) will be assigned. Those fixed values are given to compare DE analysis tools under conditions similar to SEQC or KIRC dataset and cannot be modified by users. (type: logical)

* rep: The number of datasets generated for each condition (type: integer)
* nsample: The number of samples in test and control groups. (type: numeric vector)
* nvar: The number of genes (type: integer)
* nDE: The number of DE genes (type: numeric vector)
* fraction.upregulated: The proportion of upregulated DE genes among total DE genes. (type: numeric vector)
* disp.types: ‘same’ or ‘different’. It determines whether gene expression dispersion between test and control groups are same or not. (type: character vector)
* modes: It determines the type of simulation. The possible options are,
  + ‘D’ for basic simulation (no outliers added)
  + ‘R’ for adding 5% of random outliers
  + ‘OS’ for adding outlier samples to each sample group
  + ‘DL’ for decreasing KIRC simulation dispersion 22.5 times to make it comparable to that of SEQC data.

**runSimulationAnalysis**

This function runs DE analysis using given methods. It takes fourteen arguments as follows.

* working.dir: Directory where RNA-seq simulation datasets are saved. (type: character)
* output.dir: Directory where DE analysis results are saved. (type: character)
* real: It indicates whether DE analysis is run with real dataset or synthetic dataset. (type: logical)
* fpc: TRUE or FALSE. If set TRUE (default), the dataset is generated with samples from single sample group for calculating false positive counts, and vice versa. (type: logical)
* data.types: Data types that determines the scale of gene expression and dispersion. Four possible values are available if real parameter is FALSE.,
  + KIRC : for large gene expression and dispersion
  + Bottomly : for small gene expression and dispersion
  + mBdK : for small gene expression and large dispersion
  + mKdB : for large gene expression and small dispersion

Three reference RNA-seq data are available if real parameter is TRUE

* + KIRC
  + Bottomly
  + SEQC
* fixedfold: True or False. If set False (default), the gene expression fold change between two samples will be set as + random number with exponential distribution. Here, is determined based on the sample size *N*.

If set True, DE genes are split three groups and fixed fold changes (0.625, 1.15, 1.3) will be assigned. Those fixed values are given to compare DE analysis tools under conditions similar to SEQC or KIRC dataset and cannot be modified by users. (type: logical)

* rep: the number of iterations of DE analysis (type: integer)
* nsample: The number of samples in test and control groups. (type: numeric vector)
* nDE: The number of DE genes in the simulation dataset. This parameter is for the synthetic simulation dataset. (type: integer)
* fraction.upregulated: The proportion of upregulated DE genes among total DE genes. This parameter is for the synthetic simulation dataset. (type: numeric vector)
* disp.types: ‘same’ or ‘different’. It determines whether gene expression dispersion between test and control groups are same or not. This parameter is for the synthetic simulation dataset. (type: character)
* modes: It determines the type of simulation. The possible options are,
  + ‘D’ for basic simulation (no outliers added)
  + ‘R’ for adding 5% of random outliers
  + ‘OS’ for adding outlier samples to each sample group
  + ‘DL’ for decreasing KIRC simulation dispersion 22.5 times to make it comparable to that of SEQC data.
* AnalysisMethods: A character vector indicating the list of DE analysis methods to execute. The possible methods are,
  + BaySeq
  + BaySeq.qn
  + DESeq.pc
  + DEseq2
  + edgeR
  + edgeR.ql
  + edgeR.rb
  + PoissonSeq
  + ROTS
  + SAMseq
  + voom.qn
  + voom.sw
  + voom.tmm
* para: A list parameter indicating the parameters required to run each DE analysis methods. It contains lists for each method, and in each list, the parameters for corresponding DE analysis method are included. The analysis methods not in the para list will be run with default parameters. (e.g. para=list(ROTS=list(transformation=FALSE, normalize=FALSE)) )

**performance\_plot**

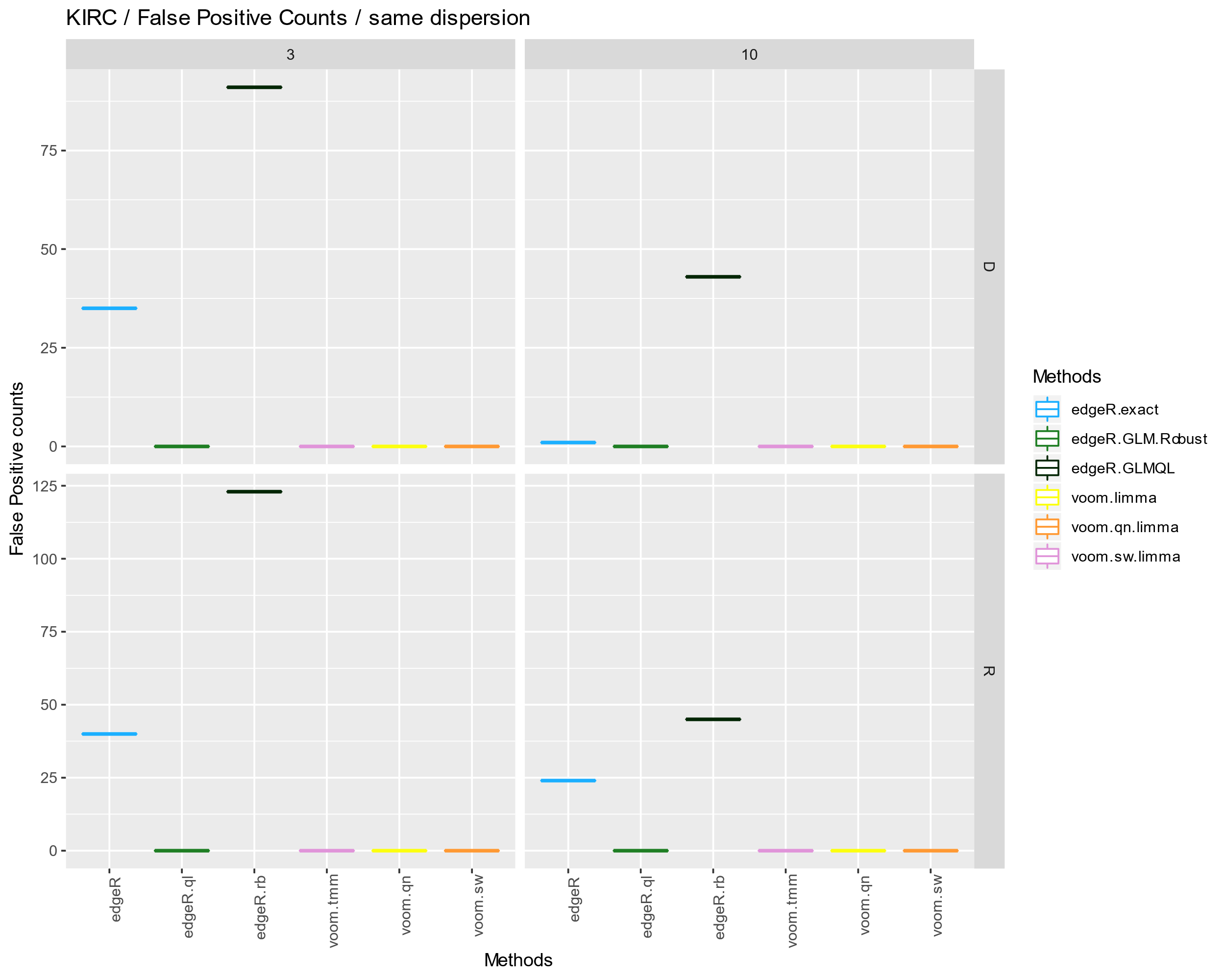
This function generates performance plot based on the DE analysis result. It takes thirteen arguments as follows.

* working.dir: Directory where DE analysis results are saved. (type: character)
* output.dir: Directory where resulting plots will be saved. (type: character)
* fixedfold: same as in the ‘runSimulationAnalysis’ function. (type: logical)
* simul.data: Simulated data types that determines the scale of gene expression and dispersion. Four possible values are available.
  + KIRC : for large gene expression and dispersion
  + Bottomly : for small gene expression and dispersion
  + mBdK : for small gene expression and large dispersion. Synthetic dataset is generated with mean parameters from Bottomly and dispersion parameters from KIRC.
  + mKdB : for large gene expression and small dispersion. Synthetic dataset is generated with mean parameters from KIRC and dispersion parameters from Bottomly.
* rep: the number of iterations of DE analysis (type: integer)
* nsample: The number of samples in test and control groups. (type: numeric vector)
* nDE: The number of DE genes in the simulation dataset. (type: integer)
* fraction.upregulated: The proportion of upregulated DE genes among total DE genes. This parameter is for the synthetic simulation dataset. (type: numeric vector)
* disp.types: ‘same’ or ‘different’. It determines whether gene expression dispersion between test and control groups are same or not. This parameter is for the synthetic simulation dataset. (type: character)
* modes: It determines the type of simulation. The possible options are,
  + ‘D’ for basic simulation (no outliers added)
  + ‘R’ for adding 5% of random outliers
  + ‘OS’ for adding outlier samples to each sample group
  + ‘DL’ for decreasing KIRC simulation dispersion 22.5 times to make it comparable to that of SEQC data.
* rowType: The type of results shown in performance plot. Subset of c(‘AUC’, ‘TPR’, ‘trueFDR’) are available. (type: character vector)
* AnalysisMethods: A character vector indicating the list of DE analysis methods to execute. The possible methods are,
  + BaySeq
  + BaySeq.qn
  + DESeq.pc
  + DEseq2
  + edgeR
  + edgeR.ql
  + edgeR.rb
  + PoissonSeq
  + ROTS
  + SAMseq
  + voom.qn
  + voom.sw
  + voom.tmm

***Toy Example2 Synthetic data false positive analysis***

|  |
| --- |
| # Load library  >> library(compareDEtools)  # Assign directories to store simulation datasets, analysis results and result plots  >> dataset.dir='~/Dataset/'  >> analysis.dir='~/Analysis/'  >> figure.dir='~/Fig/'  # Assign comparison methods  >> AnalysisMethods=c('edgeR','edgeR.ql','edgeR.rb','voom.tmm','voom.qn','voom.sw')  #Generate RNA-seq simulation dataset  >> GenerateSyntheticSimulation(working.dir=dataset.dir, data.types='KIRC', rep=10, nsample=c(3, 10), nvar=10000, nDE=0, fraction.upregulated = 0.5, disp.Types = 'same', modes=c('D','R'))  #Run DE methods for the simulation datasets  >> runSimulationAnalysis(working.dir=dataset.dir, output.dir=analysis.dir, real=FALSE, data.types='KIRC', rep=10, nsample=c(3, 10), fpc=True, nDE=c(0), disp.Types='same', modes=c('D','R'), AnalysisMethods = AnalysisMethods, para=list())  # Draw boxplots for comparing the performances of DE methods  >> fpc\_performance\_plot(working.dir=analysis.dir,figure.dir=figure.dir,simul.data='KIRC', rep=1, nsample=c(3, 10), disp.Type = 'same', modes=c('D','R'), AnalysisMethods=AnalysisMethods) |

When you check the figure directory (figure.dir), the resulting plot will be displayed as below.



Now, let’s take a look at each function used above.

**fpc\_performance\_plot**

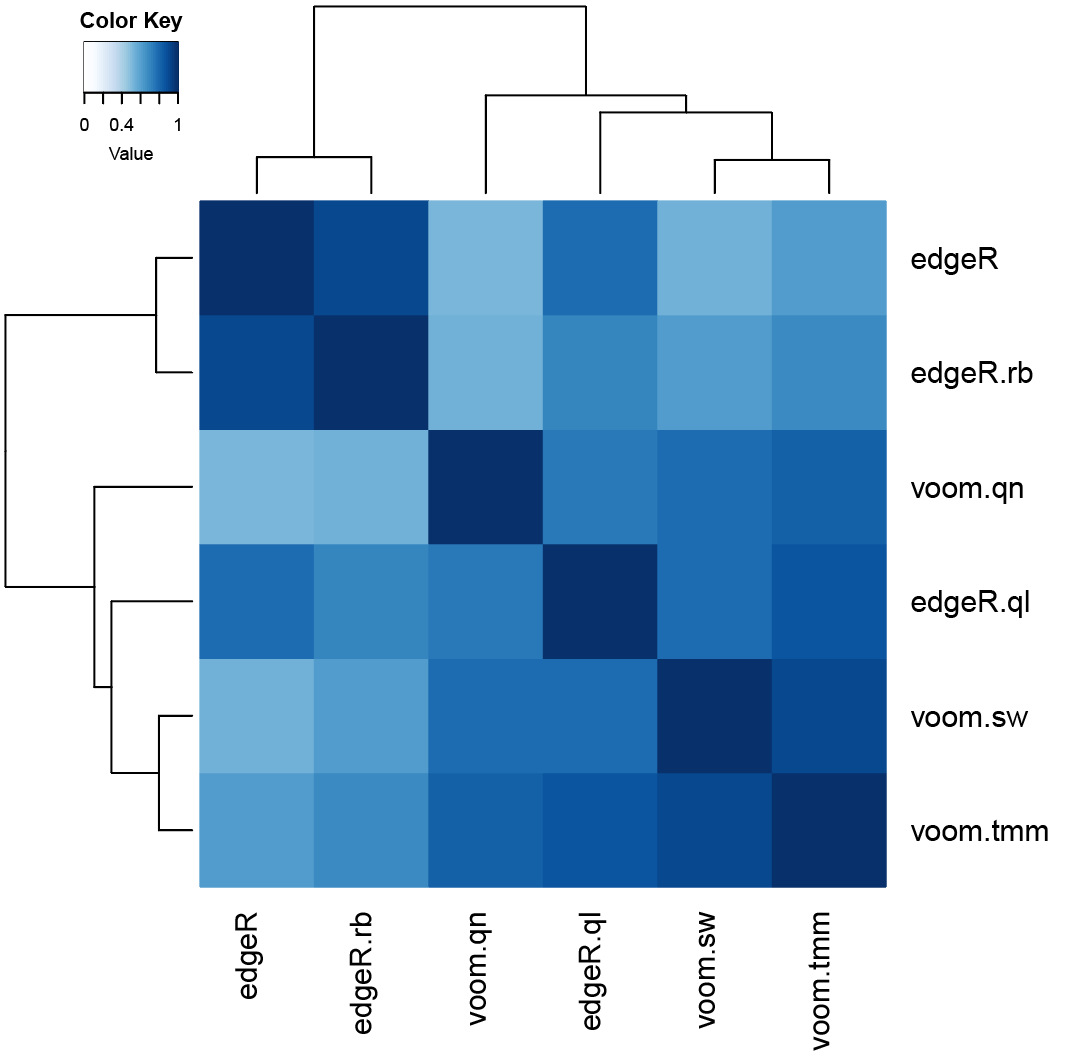
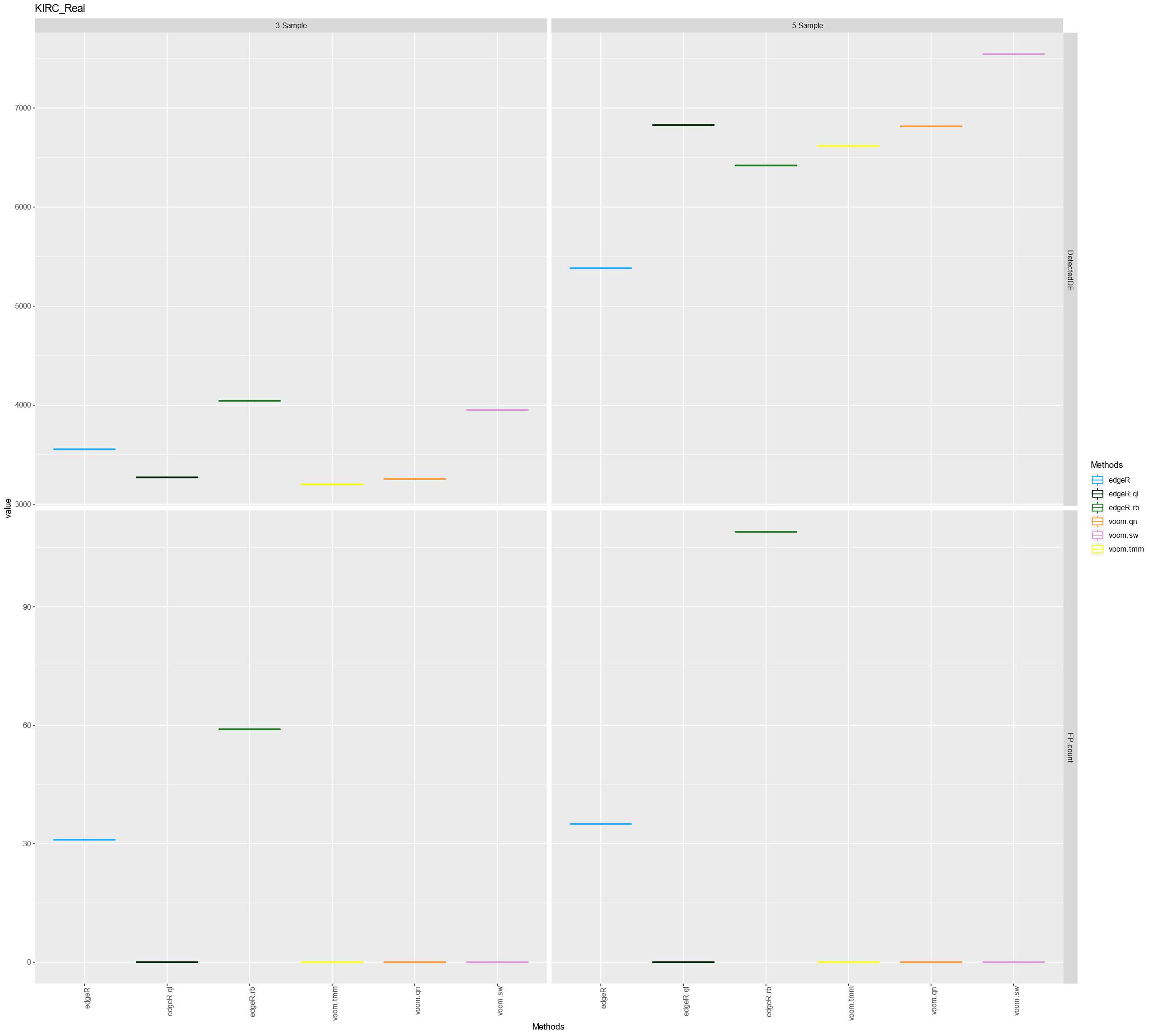
This function generates performance plot based on the DE analysis result. It takes thirteen arguments as follows.

* working.dir: Directory where DE analysis results are saved. (type: character)
* output.dir: Directory where resulting plots will be saved. (type: character)
* simul.data: Simulated data types that determines the scale of gene expression and dispersion. Four possible values are available.
  + KIRC : for large gene expression and dispersion
  + Bottomly : for small gene expression and dispersion
  + mBdK : for small gene expression and large dispersion. Synthetic dataset is generated with mean parameters from Bottomly and dispersion parameters from KIRC.
  + mKdB : for large gene expression and small dispersion. Synthetic dataset is generated with mean parameters from KIRC and dispersion parameters from Bottomly.
* rep: the number of iterations of DE analysis (type: integer)
* nsample: The number of samples in test and control groups. (type: numeric vector)
* disp.Type: ‘same’ or ‘different’. It determines whether gene expression dispersion between test and control groups are same or not. (type: character)
* modes: It determines the type of simulation. The possible options are,
  + ‘D’ for basic simulation (no outliers added)
  + ‘R’ for adding 5% of random outliers
  + ‘OS’ for adding outlier samples to each sample group
  + ‘DL’ for decreasing KIRC simulation dispersion 22.5 times to make it comparable to that of SEQC data.
* AnalysisMethods: A character vector indicating the list of DE analysis methods to execute. The possible methods are,
  + BaySeq
  + BaySeq.qn
  + DESeq.pc
  + DEseq2
  + edgeR
  + edgeR.ql
  + edgeR.rb
  + PoissonSeq
  + ROTS
  + SAMseq
  + voom.qn
  + voom.sw
  + voom.tmm

***Toy Example3 Real data analysis***

|  |
| --- |
| # Load library  >> library(compareDEtools)  # Assign directories to store simulation datasets, analysis results and result plots  >> dataset.dir='~/Dataset/'  >> analysis.dir='~/Analysis/'  >> figure.dir='~/Fig/'  # Assign comparison methods  >> AnalysisMethods=c('edgeR','edgeR.ql','edgeR.rb','voom.tmm','voom.qn','voom.sw')  #Generate RNA-seq simulation dataset  >> GenerateRealSimulation(working.dir=dataset.dir, data.types='KIRC', rep=1, nsample=c(3,5))  >> GenerateRealSimulation(working.dir=dataset.dir, data.types='KIRC', fpc=TRUE, rep=1, nsample=c(3,5))  #Run DE methods for the simulation datasets  >> runSimulationAnalysis(working.dir=dataset.dir, output.dir=analysis.dir, real=TRUE, data.types='KIRC', rep=1, nsample=c(3,5), AnalysisMethods = AnalysisMethods, para=list())  >> runSimulationAnalysis(working.dir=dataset.dir, output.dir=analysis.dir, real=TRUE, fpc=TRUE, data.types='KIRC', rep=1, nsample=c(3,5), AnalysisMethods = AnalysisMethods, para=list())  # Draw boxplots for comparing the performances of DE methods  >> performance\_realdata\_plot(working.dir=analysis.dir,figure.dir=figure.dir,simul.data='KIRC', rep=1, nsample=c(3,5), AnalysisMethods=AnalysisMethods, rowType = c("DetectedDE","FP.count"))  # Draw correlation heatmap for comparing the performances of DE methods  >>correlation\_heatmap(working.dir=analysis.dir, figure.dir=figure.dir,simul.data='KIRC', nsample=5, topgenes=5000, AnalysisMethods=AnalysisMethods, rep=1) |

When you check the figure directory (figure.dir), the resulting plot will be displayed as below.



Now, let’s take a look at each function used above.

**GenerateRealdataSimulation**

Unlike ‘GenerateSyntheticSimulation’ function, this function generates dataset consisting of a sample subset of the real RNA-seq data. It takes five arguments as follows.

* working.dir: A directory where simulation datasets are stored (type: character)
* fpc: TRUE or FALSE. If set TRUE (default), the dataset is generated with samples from single sample group for calculating false positive counts, and vice versa. (type: logical)
* data.types: The reference RNA-seq data. One of ‘KIRC’, ‘Bottomly’ and ‘SEQC’ is possible. (type: character vector)
* rep: The number of datasets generated for each condition (type: integer)
* nsample: The number of samples in test and control groups. (type: numeric vector)

**performance\_realdata\_plot**

This function generates performance plot based on the DE analysis result. It takes thirteen arguments as follows.

* working.dir: Directory where DE analysis results are saved. (type: character)
* output.dir: Directory where resulting plots will be saved. (type: character)
* simul.data: Simulated types of reference RNA-seq data.
  + KIRC
  + Bottomly
  + SEQC
* rep: the number of iterations of DE analysis (type: integer)
* nsample: The number of samples in test and control groups. (type: numeric vector)
* rowType: The type of results shown in performance plot. Subset of c(‘DetectedDE’, ‘FP.count’) are available for ‘KIRC and ‘Bottomly’ datasets. Subset of c(‘AUC’, ‘TPR’, ‘trueFDR’) are available for SEQC datasets. (type: character vector)
* AnalysisMethods: A character vector indicating the list of DE analysis methods to execute. The possible methods are,
  + BaySeq
  + DESeq.pc
  + DEseq2
  + edgeR
  + edgeR.ql
  + edgeR.rb
  + PoissonSeq
  + BaySeq.qn
  + ROTS
  + SAMseq
  + voom.qn
  + voom.sw
  + voom.tmm

**correlation\_heatmap**

This function generates correlation heatmap based on the DE analysis result. It takes thirteen arguments as follows.

* working.dir: Directory where DE analysis results are saved. (type: character)
* output.dir: Directory where resulting plots will be saved. (type: character)
* simul.data: Simulated types of reference RNA-seq data. Only two reference data are available.
  + KIRC
  + Bottomly
* rep: the number of iterations of DE analysis (type: integer)
* nsample: The number of samples in test and control groups. (type: numeric vector)
* topgenes: The number of top significant genes selected by q-values from each DE methods. Gene rankings are calculated with the union of top significant genes from DE methods and similarity matrix is generated based on the rankings.
* AnalysisMethods: A character vector indicating the list of DE analysis methods to execute. The possible methods are,
  + BaySeq
  + BaySeq.qn
  + DESeq.pc
  + DEseq2
  + edgeR
  + edgeR.ql
  + edgeR.rb
  + PoissonSeq
  + ROTS
  + SAMseq
  + voom.qn
  + voom.sw
  + voom.tmm