dgeAnalysis v1.5.1

User Manual

Leids Universitair
Medisch Centrum

SASC
Sequencing Analysis Support Core

Authors
Tom Kuipers
Hailiang (Leon) Mei
Davy Cats

10-09-2022

1. Preface

With the development of next-generation sequencing techniques, RNA-Seq is becoming a more commonly applied method in various fields, e.g., cancer genomics, cell biology, etc. RNA-Seq gives great insights into a biological species at a specific moment. To find out which genes (or transcripts) are playing an important function at a specific moment or under a specific condition, differential gene expression analysis is used. With the use of this type of analysis, research between different conditions can be conducted.

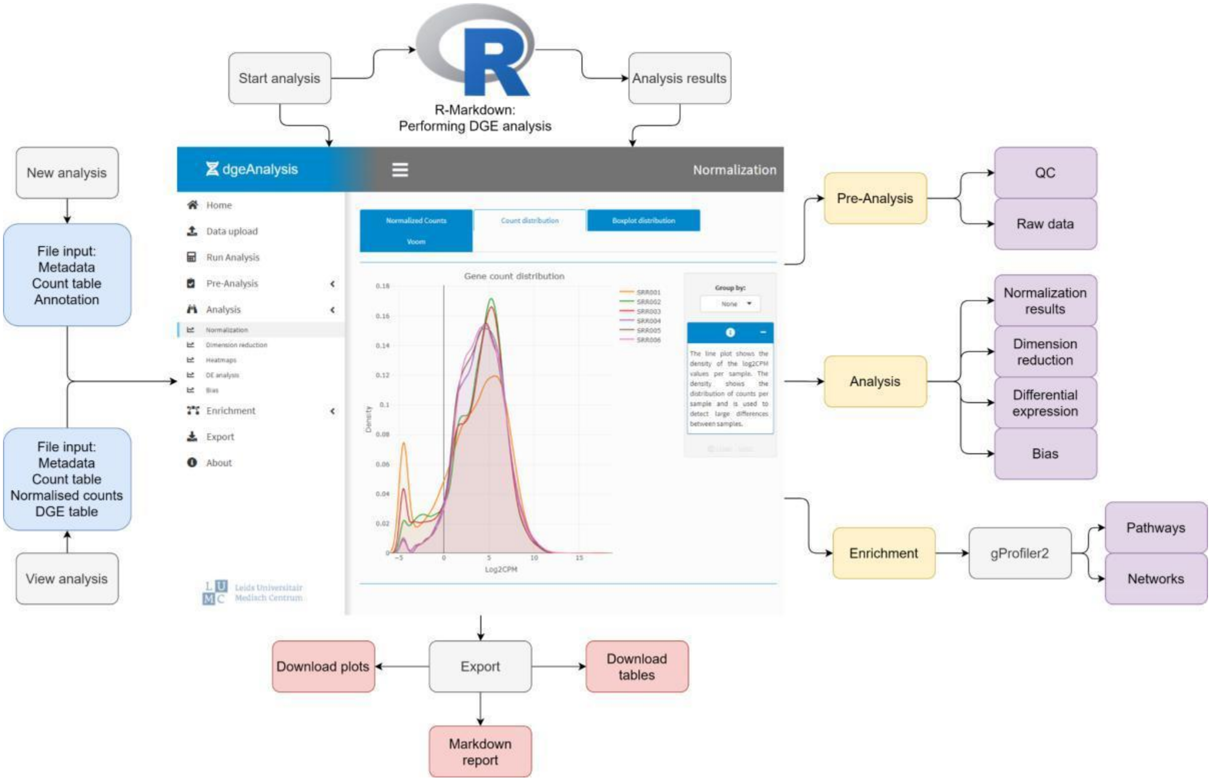
This application is created using primarily R code, together with a variation of R packages including Shiny, edgeR, limma, and DESeq2. The application is built upon a shiny based framework and so can be used in any R environment. Using R-markdown for back-end analysis, reproducibility can be ensured. The R-markdown files contain all steps taken to perform an analysis, together with all used input variables and dependencies, including version numbers. The front-end images are all created with the use of ggplot2 & plotly to generate interactive plots, with click and hover options.

Using the application for differential gene analysis has proven to be very helpful in many projects. Next to checking data quality, it also provides a platform to carry out a DGE analysis quickly. This makes it an easy-to-use platform capable of performing a variety of analysis possibilities.

The package is available from GitHub:

[GitHub - LUMC/dgeAnalysis: R package with Shiny application for DGE analysis]

A quick view of the application:



The line plot shows the density of the log2CPM values per sample. The density shows the distribution of counts per sample and is used to detect large differences between samples.

2. Start dgeAnalysis

2.1. RStudio

To use the *dgeAnalysis* package has a number of requirements before installing the package in R.

1. R v4.1+ The R Project for Statistical Computing
2. RStudio v2022.02.2+48 https://rstudio.com/
3. A web browser (pref. Google Chrome or Mozilla Firefox)

If installing on Windows, make sure RTools and pandoc is installed:
R 4.0 and up:
https://cran.r-project.org/bin/windows/Rtools/rtools40-i686.exe
https://cran.r-project.org/bin/windows/Rtools/rtools40-x86_64.exe
Pandoc:
https://pandoc.org/installing.html

If installing on Linux, make sure these libraries are installed. This can be done by running this in the Linux terminal:

```
sudo apt-get update && apt-get install \
   build-essential \
   gdebi-core \
   pandoc \
   pandoc-citeproc \
   libcairo2-dev \
   libjpeg-dev \
   libxt-dev \
   libssl-dev \
   libcurl4-gnutls-dev \
   libxml2-dev \
   libcurl4-openssl-dev \
   -y
```

With R and RStudio installed, the *dgeAnalysis* package can be installed. First, open RStudio and type the following steps in the **Console**:

1. Install the “devtools” package (if not already installed)

```
install.package("devtools")
```

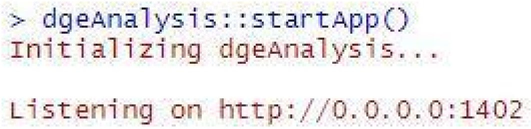
2. Install the *dgeAnalysis* package. Sometimes installing the package can cause problems. These are often related to other packages failing to install. Check

```
library("devtools")
devtools::install_github("LUMC/dgeAnalysis")
```

3. Launch the application

```
library("dgeAnalysis")
dgeAnalysis::startApp()
```

Running the command `dgeAnalysis::startApp()` first initializes the application, before it is opened in the default web browser:



```
> dgeAnalysis::startApp()
Initializing dgeAnalysis...

Listening on http://0.0.0.0:1402
```

2.2. Docker image

This method is specifically for Linux or macOS users (A browser (pref. Google Chrome or Mozilla Firefox is required)

Make sure Docker is installed: https://docs.docker.com/get-docker/

To start the docker image, follow these steps:

1. Open a terminal
2. Load docker image

```
docker load < dgeAnalysis.tar.gz
```

3. Run docker image

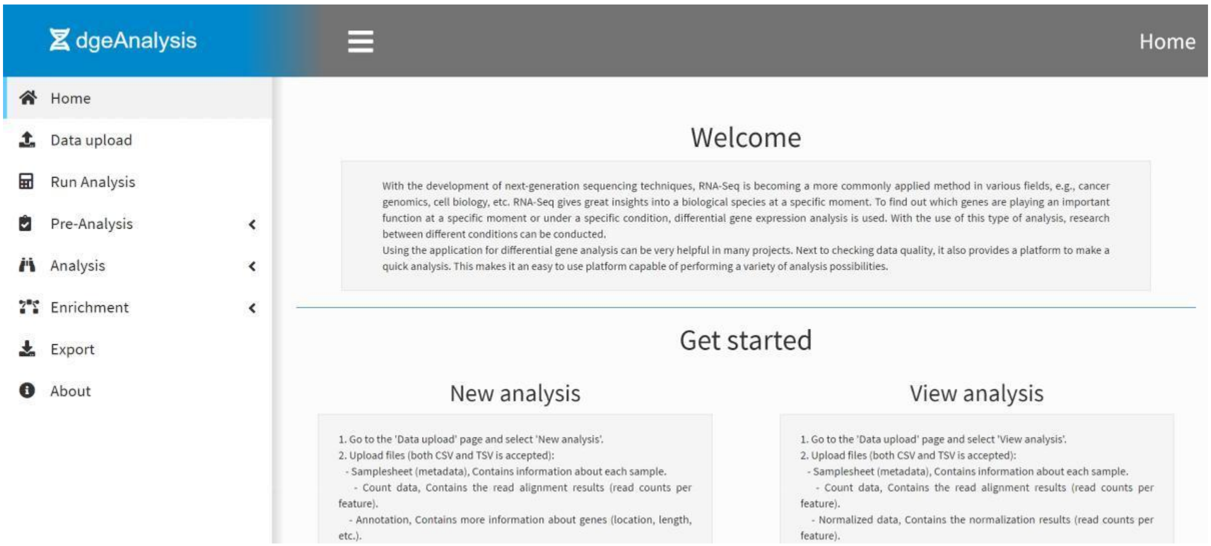
```
docker run -p 1402:1402 dge_analysis
```

4. If a browser isn't opened automatically, click or copy the link in the generated output.

```
Listening on http://0.0.0.0:1402
```

2.3. Application successfully started

Once all packages are loaded, the application will look like this:



Welcome

With the development of next-generation sequencing techniques, RNA-Seq is becoming a more commonly applied method in various fields, e.g., cancer genomics, cell biology, etc. RNA-Seq gives great insights into a biological species at a specific moment. To find out which genes are playing an important function at a specific moment or under a specific condition, differential gene expression analysis is used. With the use of this type of analysis, research between different conditions can be conducted.
Using the application for differential gene analysis can be very helpful in many projects. Next to checking data quality, it also provides a platform to make a quick analysis. This makes it an easy to use platform capable of performing a variety of analysis possibilities.

Get started

New analysis

1. Go to the 'Data upload' page and select 'New analysis'.
2. Upload files (both CSV and TSV is accepted):
 - Samplesheet (metadata), Contains information about each sample.
 - Count data, Contains the read alignment results (read counts per feature).
 - Annotation, Contains more information about genes (location, length, etc.).

View analysis

1. Go to the 'Data upload' page and select 'View analysis'.
2. Upload files (both CSV and TSV is accepted):
 - Samplesheet (metadata), Contains information about each sample.
 - Count data, Contains the read alignment results (read counts per feature).
 - Normalized data, Contains the normalization results (read counts per feature).

The application is now ready to use!

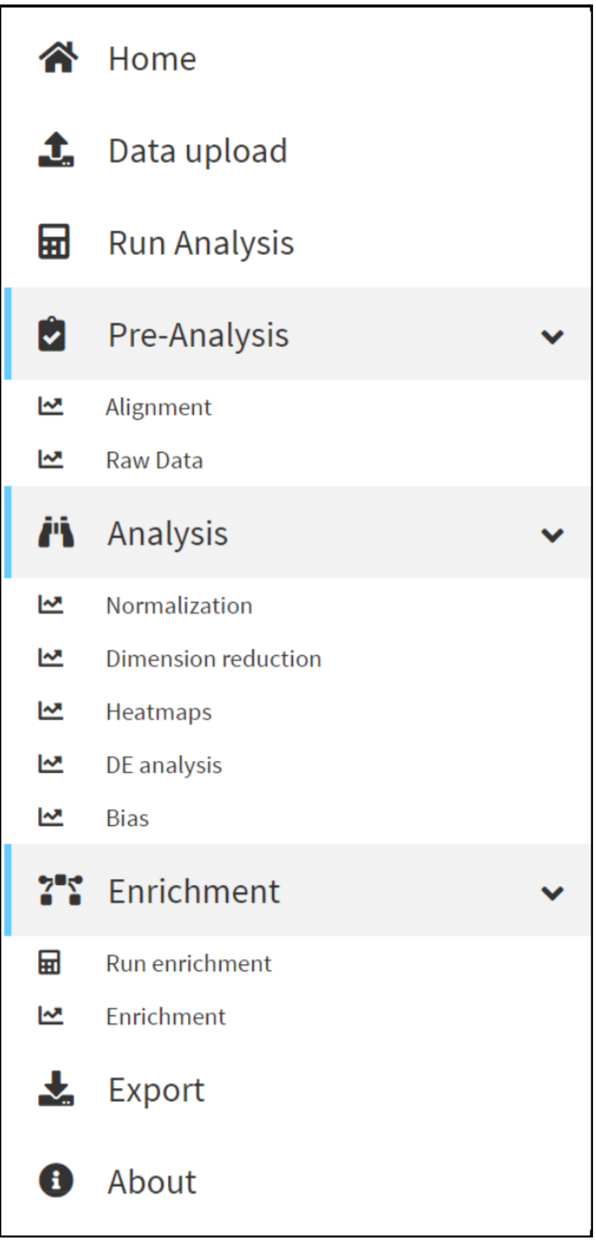
2.4. Example data

On the GitHub page of *dgeAnalysis* there is a folder called “exampleData”. This folder includes a zip file with a sample sheet, count table, and an annotation file. These files can be viewed to check the data format and perform a test analysis. Note: The test data is of no use, so the results should be treated as such.

3. Procedures

Once the application is loaded and started in a web browser, it's ready to use. There are multiple pages in the application, which will be explained here. Navigation through the application is easy using the navigation bar on the left of the screen:

- **Home**
  The home page contains a short explanation of the application and a quick startup guide.
- **Data upload**
  Here all raw data can be uploaded and viewed to check that data is loaded correctly.
- **Run Analysis**
  Change settings and run an analysis with the uploaded data.
- **Pre-Analysis**
  View distribution of raw data and perform quality checks based on raw data.
- **Analysis**
  View the distribution of normalized data, PCA, and the differential gene expression analysis results.
- **Enrichment**
  Perform enrichment analysis on the analyzed data, with the use of multiple databases.
- **Export**
  Download generated tables or a markdown report containing all used analysis steps that have been running on the background.
- **About**
  Check application information and information about the current session (like package versions, etc.).



3.1. Perform a new analysis

3.1.1. Upload data

Before starting an analysis, load all data in the “Data upload” page. Here there is an option to start a new analysis. To start a new analysis, there are three file upload options:

1. Metadata/sample sheet
   - This contains the sample information.
2. Count table
   - This contains the read alignment results (read counts per feature).
3. Annotation (optional)
   - This contains more information about genes (location, length, etc.).

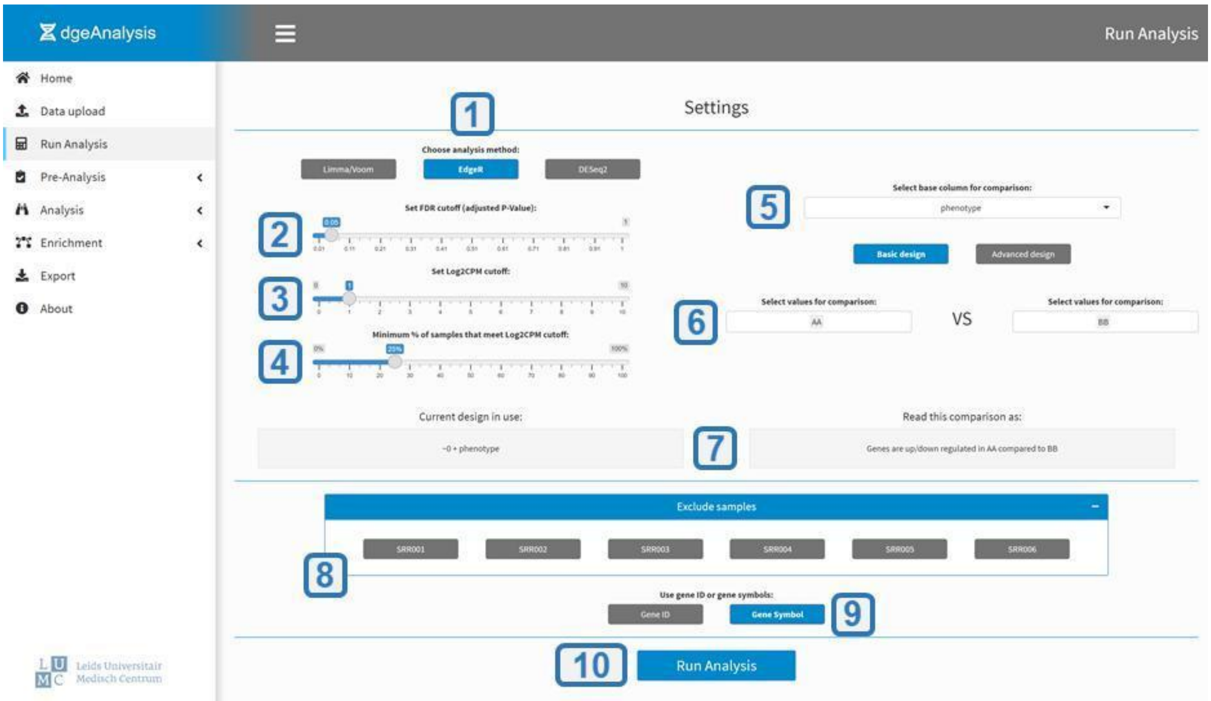
Where the metadata and count table are mandatory, the annotation is optional. This is because they are not needed to perform the analysis itself. It is recommended to do upload an annotation as well because this can add more insight into the data. To perform gene set enrichment, an annotation is required. All uploaded data is shown in table format. This can be used as a control check to see if all data is loaded correctly.

3.1.2. Check Pre-Analysis

Before starting the actual analysis, it’s a good idea to first check the distribution of raw data. This can be done in the tabs below “Pre-Analysis”. Check how the alignment results to see how many reads actually are aligned and how the reads are distributed across the biggest genes. Next, check how the raw data is distributed, possible outliers can already be detected here.

3.1.3. Run the analysis

Navigate to the “Run analysis” page to start a new analysis.



Choose analysis method: EdgeR

Set FDR cutoff (adjusted P-Value): 0.05

Set Log2CPM cutoff: 1

Minimum % of samples that meet Log2CPM cutoff: 25%

Select base column for comparison: phenotype

Select values for comparison: AA

Select values for comparison: BB

Current design in use: ~0 + phenotype

Read this comparison as: Genes are up/down regulated in AA compared to BB

Use gene ID or gene symbols: Gene Symbol

(1) *Choose between Limma/Voom, EdgeR, or DESeq2 as a method to run the analysis with.*
(2) *Change the p-value cutoff (default=“0.05”) to define differentially expressed genes.*

(3) Change the Log2CPM cutoff (default="1") to set a CPM filter for low expressed genes.
(4) The minimum number of samples in % that meet the Log2CPM value at (3) (default=25%).
(5) Set the base column to start the comparison and select a design type (Basic or advanced).
(6) Select values to set a comparison in the inputs (based on selected columns).
(7) The text shows the design matrix (left) and the comparison (right).
(8) Select samples to be excluded from the analysis.
(9) Set if genes should be shown in plots/tables with ID or symbol.
(10) With the button "Run Analysis" the analysis is started.

There are three analysis methods: Limma, EdgeR, and DESeq2. The method selected, will be the method used to perform the differential gene expression analysis. There are also two values that can be used in the analysis: p-value & Log2CPM. For the Log2CPM, there are two values: One to set the CPM cutoff value and one to decide how many samples (in %) should meet this value. When a gene doesn't match these values across samples it will be filtered out, this to remove low expressed genes. The p-value is used to define the differential expressed genes.

To set a comparison, a column is selected from the sample sheet. Based on this column, it's possible to create an analysis with a basic- or advanced design. The design of the analysis categorizes the samples based on the metadata input. The exact values to compare can be chosen based on the selected design columns and determine which samples are compared with each other.

When it's necessary to exclude certain samples from the analysis, then these samples can be selected. All selected samples will be excluded during the analysis, so they don't have any effect on the results. This is very useful in case a sample is a clear outlier.

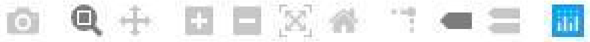
If an annotation file is uploaded, there is an extra option to show results with the gene IDs (Ensembl IDs) or with gene symbols (gene names). To start the analysis with all uploaded files and settings, there is a button with "Run Analysis" to perform the analysis.

3.1.4. Check the analysis results

After the analysis has finished, view all results from top to bottom in the tabs below "Analysis". The normalization tabs are the same as the raw data tabs. Now the data can be compared to see the difference before and after normalization. Check the PCA on possible outliers as well. If these plots show a clear outlier, it might be worth removing them from the analysis. This means rerunning the analysis in "Run Analysis" and selecting the samples that need to be excluded. The DE analysis tab contains plots that visualize the actual analysis results based on the set comparison.

3.1.5. Export results

Downloading a plot can be done from the plot itself. Hover with your mouse somewhere over a plot and see this taskbar opening in the top-right corner:



Click on the camera icon on the left to download a plot.

To export tables, navigate to the page "Export". Here, choose a table in the select menu and download the results as CSV or TSV by clicking on one of the two buttons.

!! IMPORTANT !!

Always download the R Markdown report after ANY analysis. The R Markdown report contains all information about the current analysis, including all used settings, intermediate steps, and important results. This HTML file is essential to ensure reproducibility!

3.1.6. Enrichment

The enrichment analysis is using the gProfiler2 package to determine enriched pathways. This function uses a list of genes (gene IDs or gene names) to find enriched terms. It's not required to perform a DGE analysis first. If a DGE analysis has been performed previously, results can be filtered using the in-app filtering options. With these filtering options, a gene list can be made for genes with a specific log2FC or p-value. Also, selecting only (up and/or down) regulated genes can be performed here.

There are five databases available that can be used to search for enriched terms:

1. GO
2. KEGG
3. Reactome
4. WikiPathways
5. Human phenotype ontology

3.2. View an analysis

To view an analysis, these files should be uploaded for the full use of the application:

1. Metadata/sample sheet
    - This contains the sample information.
2. Count table
    - This contains the read alignment results (read counts per feature).
3. Normalized counts
    - This contains the normalization results of the counts per feature.
4. DE table
    - This contains all the analysis results.

The metadata is required, but the other three files are only needed to view specific parts of the application: A count table to view raw data and QC, Normalized counts to view normalization results, and PCA and DE table to view differential expression results.

4. Install packages manually

If there is trouble installing some packages, copy the line of a specific packages into the R/RStudio console to install the package manually:

```
## Shiny environment
if (!require("shiny")) install.packages("shiny")
if (!require("shinydashboard")) install.packages("shinydashboard")
if (!require("shinyWidgets")) install.packages("shinyWidgets")
if (!require("shinycssloaders")) install.packages("shinycssloaders")
if (!require("shinyjs")) install.packages("shinyjs")

## Differential expression analysis
if (!require("BiocManager")) install.packages("BiocManager")
if (!require("knitr")) install.packages("knitr")
if (!require("SummarizedExperiment")) BiocManager::install("SummarizedExperiment")
if (!require("edgeR")) BiocManager::install("edgeR")
if (!require("limma")) BiocManager::install("limma")
if (!require("DESeq2")) BiocManager::install("DESeq2")
if (!require("tidyr")) install.packages("tidyr")
if (!require("scales")) install.packages("scales")
if (!require("broom")) install.packages("broom")
if (!require("plotly")) install.packages("plotly")
if (!require("Rtsne")) install.packages("Rtsne")
if (!require("rmarkdown")) install.packages("rmarkdown")


## Pathway analysis
if (!require("gprofiler2")) install.packages("gprofiler2")
if (!require("igraph")) install.packages("igraph")
```