# Contents

**The START app allows users to visualize RNA-seq data starting with count data.**

- *Explore* the app's features with the example data set pre-loaded by clicking on the tabs above.
- *Upload* your data in the "Input Data" tab.

# Instructions

The app is hosted on the website: https://kcvi.shinyapps.io/START/

Code can be found on github: https://github.com/jminnier/STARTapp

To run this app locally on your machine, download R or RStudio and run the following commands once to set up the environment:

```
install.packages(c("reshape2","ggplot2","ggthemes","gplots","ggvis","dplyr","tidyr","DT",
                   "RColorBrewer","pheatmap","shinyBS","plotly","markdown","NMF","scales","heatmaply"))
## try http:// if https:// URLs are not supported
source("https://bioconductor.org/biocLite.R")
biocLite(c("limma","edgeR"))
```

You may now run the shiny app with just one command in R:

```
shiny::runGitHub("STARTapp", "jminnier")
```

## Input Data

You may use this app by

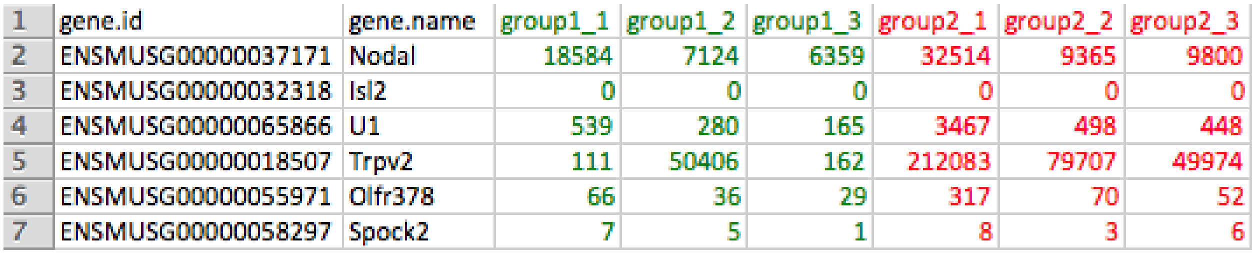
1. Exploring the pre-loaded example data set. This is a pre-loaded mouse RNA-seq example for exploring the app's features.
2. Upload your own data that is either
    i. Count data (or log2-expression data)
    ii. Analyzed data = expression data + p-values and fold changes.
3. Uploading an .RData file containing your data that was previously downloaded from a START app session.

### Data Format

- Must be a .CSV *comma-separated-value* file (you may export from Excel).
- File must have a header row.
- First/Left-hand column(s) must be gene identifiers.
- Format expression column names as `GROUPNAME_REPLICATE#`, e.g. `Treat_1, Treat_2, Treat_3, Control_1, Control_2, High_1, High_2`

### Count or Expression Data

- Each row denotes a gene, each column denotes a sample.



| 1 | gene.id | gene.name | group1_1 | group1_2 | group1_3 | group2_1 | group2_2 | group2_3 |
|---|---|---|---|---|---|---|---|---|
| 2 | ENSMUSG00000037171 | Nodal | 18584 | 7124 | 6359 | 32514 | 9365 | 9800 |
| 3 | ENSMUSG00000032318 | Isl2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | ENSMUSG00000065866 | U1 | 539 | 280 | 165 | 3467 | 498 | 448 |
| 5 | ENSMUSG00000018507 | Trpv2 | 111 | 50406 | 162 | 212083 | 79707 | 49974 |
| 6 | ENSMUSG00000055971 | Olfr378 | 66 | 36 | 29 | 317 | 70 | 52 |
| 7 | ENSMUSG00000058297 | Spock2 | 7 | 5 | 1 | 8 | 3 | 6 |

Figure 1:

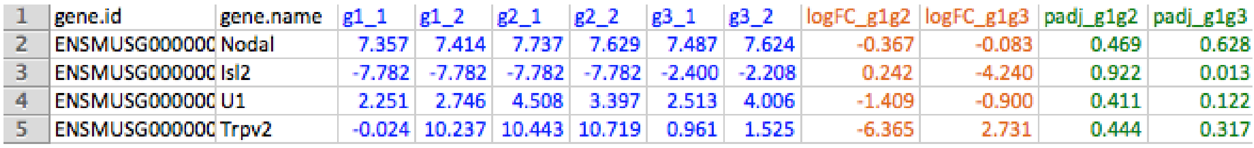
Count data contains read counts for each gene for each sample, along with gene identifiers.

Analysis: When raw counts are uploaded, the data is then analyzed by the app. The app uses the voom method from the 'limma' Bioconductor package to transform the raw counts into logged and normalized intensity values. These values are then analyzed via linear regression where gene intensity is regressed on the group factor. P-values from all pairwise regression tests for group effect are computed and Benjamini-Hochberg false discovery rate adjusted p-values are computed for each pairwise comparison. The "log2cpm" values are the log2-counts-per-million values. The "log2cpm_voom" values are the normalized logcpm values from the voom method. Both methods use an offset of 0.5, which means 0.5 is added to all count values before normalizing (in the case of voom) and log transforming so that 0 counts have non infinite values.

Example file: https://github.com/jminnier/STARTapp/blob/master/data/examplecounts_short.csv

### Analyzed Data

- Each row denotes a gene, each column denotes a sample.
- Additional columns provide Fold Changes and P-values



| 1 | gene.id | gene.name | g1_1 | g1_2 | g2_1 | g2_2 | g3_1 | g3_2 | logFC_g1g2 | logFC_g1g3 | padj_g1g2 | padj_g1g3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | ENSMUSG00000C | Nodal | 7.357 | 7.414 | 7.737 | 7.629 | 7.487 | 7.624 | -0.367 | -0.083 | 0.469 | 0.628 |
| 3 | ENSMUSG00000C | Isl2 | -7.782 | -7.782 | -7.782 | -7.782 | -2.400 | -2.208 | 0.242 | -4.240 | 0.922 | 0.013 |
| 4 | ENSMUSG00000C | U1 | 2.251 | 2.746 | 4.508 | 3.397 | 2.513 | 4.006 | -1.409 | -0.900 | 0.411 | 0.122 |
| 5 | ENSMUSG00000C | Trpv2 | -0.024 | 10.237 | 10.443 | 10.719 | 0.961 | 1.525 | -6.365 | 2.731 | 0.444 | 0.317 |

Figure 2:

Analyzed data must contain some kind of expression measure for each sample (i.e. counts, normalized intensities, CPMs), and a set of p-values with corresponding fold changes for those p-values. For instance, if you have a p-value for the comparison of group1 vs group2, you can upload the observed fold change or log2(fold change) between group1 vs group2. If you have a more complex design and do not have fold changes readily available, you may upload the test statistics or other similar measures of effect size as placeholders. The fold changes are mainly used in the volcano plots. We recommend uploading p-values that are adjusted for multiple comparisons (such as q-values from the qvalue package, or adjusted p-values from p.adjust() function in R).

Example file: https://github.com/jminnier/STARTapp/blob/master/data/exampleanalysisres_short.csv

### *TIP*: Save Data for Future Upload

After submitting a raw data or analyzed file, you may download the .csv file with the analysis results for your own use (or to upload as an "analyzed data") or more conveniently click the button "Save Results as RData File for Future Upload" so that you may easily and quickly upload your data to the START app in the future under the "RData from previous START upload" option with one click.

After uploading your data to START, click red button

to download an .RData file to upload your data to START with one click.

Next time use the "Input Data" tab –> "START RData file" option.

# Visualizations

## Group Plots

### PCA Plot

This plot uses Principal Component Analysis (PCA) to calculate the principal components of the expression data using data from all genes. Euclidean distances between expression values are used. Samples are projected on the first two principal components (PCs) and the percent variance explained by those PCs are displayed along the x and y axes. Ideally your samples will cluster by group identifier.

### Sample Distance Heatmap

This plot displays unsupervised clustering of the Euclidean distances between samples using data from all genes. Again your data should cluster by group.

## Analysis Plots

These plots use the p-values and fold changes to visualize your data.

### Volcano Plot

This is a scatter plot log fold changes vs –log10(p-values) so that genes with the largest fold changes and smallest p-values are shown on the extreme top left and top right of the plot. Hover over points to see which gene is represented by each point. (https://en.wikipedia.org/wiki/Volcano_plot_(statistics))

### Scatter Plot

This is a scatter plot of average gene expression in one group against another group. This allows the viewer to observe which genes have the largest differences between two groups. The smallest distances will be along the diagonal line, and points far away from the diagonal show the most differences. Hover over points to see which gene is represented by each point.

## Gene Expression Boxplot

Use the search bar to look up genes in your data set. For selected gene(s) the stripchart (dotplot) and boxplots of the expression values are presented for each group. You may plot one or multiple genes along side each other. Hover over points for more information about the data.

## Heatmap

A heatmap of expression values are shown, with genes and samples arranged by unsupervised clustering. You may filter on test results as well as P-value cutoffs. By default the top 100 genes (with lowest P-values) are shown.