Babraham Bioinformatics

Final Exercise: R Notebooks

Version 2020-11-02

Licence

Setting up a git repository

Create a new git repository for this work on github. Initialise the repository with a `README.md` file and add some text in there to explain what the repository is.

Check out the repository to your machine. Copy the `rnaseq_counts.txt` file from the Notebook data folder and check it in to the repository.

Create an R Notebook called `rna_seq_analysis.Rmd` and check that into git.

For each step below check your changes into git with a suitable commit message after each change.

Data Preparation

This data is raw counts for all mouse genes (called Probes in the dataset) for 10 samples belonging to two groups (`Ish` and `T47D`)

Read in the tab delimited data into a variable. Fix any import problems.

Split the gene annotations and counts (plus `Probe` name) into separate tibbles.

Restructure the count data into tidy format using `pivot_longer` and `separate` the original sample name into `condition` and `replicate`, but retain the original sample name column too.

Remove any Probes from the data where their count in all samples is zero

Data Normalisation and Summarisation

Normalise the raw counts within each sample to be `log2` counts per million counts (in that sample). Since you'll have some zero values still you need to add one to all counts before doing this.

Plot out a violin plot of this new log2rpm value for all samples to check that they appear to be correctly normalised. Colour this by condition.

From the log2rpm values calculate the mean value of the replicates per condition for each Probe to get a single value per Probe for `Ish` and `T47D`. Restructure the data so that `Ish` and `T47D` appear in separate columns.

Plotting and analysis

Plot a scatterplot of all of the genes with `Ish` on the x-axis and `T47D` on the y-axis. Make sure the plot is (roughly) square.

Calculate a list of the 50 genes with the highest absolute difference between the two conditions and print this table.

One gene family which changes a lot is the GAGE family. Pull the expression values for these genes out from the dataset and annotate them with the chromosome they came from (you'll need to join back to the annotation data you separated earlier).

Redraw the scatterplot but highlight on it any gene whose name starts with GAGE. You can do this either by adding a new column to the dataset to indicate GAGE or Non-GAGE and colouring by that, or by adding additional geometries and specifying the GAGE subset as the ggplot data for that layer. The first is somewhat easier to do, but the second gives you more flexibility.

Generate an HTML report of this analysis. Make sure that the steps are properly described and discussed. Have headings to help navigate. Make sure the document has a table of contents.