Babraham Bioinformatics

Exercises:
Multi-Condition RNA-Seq

Version 2018-04

Licence

Introduction

In this follow-on exercise to the basic RNA-Seq exercises we will look at a more challenging dataset. We're still asking the same question, looking for genes which are differentially expressed. The differences in this dataset are:

- The size of the data is larger. We are working with 18 separate samples this time.
- The structure of the experimental design is more complex. We have 4 separate conditions and unequal sample sizes between them.

Many parts of the analysis of this data will be the same as before, but may be more challenging on a larger data set. Where the analysis will differ will be in the way the statistics are performed, and in the methods we can use to try to reduce the complexity of the initial results we get – to leave us with gene sets which are more amenable to biological interpretation.

Software

The software which will be used in this session is listed below. In this case we are starting with data which has already been mapped and loaded into a seqmonk project so we're only looking at the software we're using for visualisation and statistical analysis:

- SeqMonk (http://www.bioinformatics.babraham.ac.uk/projects/seqmonk/)
- R (http://www.r-project.org/)
- DESeq2 – part of bioconductor (http://www.bioconductor.org/)

Data

The data in this practical comes from ArrayExpress E-MTAB-822 Transcription profiling by high thoughput sequencing of human cell lines Ishikawa, MCF7 and T47D treated with estrogen, progesterone and their antagonists.

The data has already been mapped against the GRCh37 genome assembly using TopHat (this is quite an old dataset), and was imported into SeqMonk using default parameters.

The data should be provided to you in a SeqMonk project file called "large_rna_seq.smk" which you can open with `File > Open Project`.

Exercise 1: Data Preparation

Before we get stuck into the exploration and analysis of this data, it will make our lives easier if we can clean up the data a bit first. The main things we need to do are:

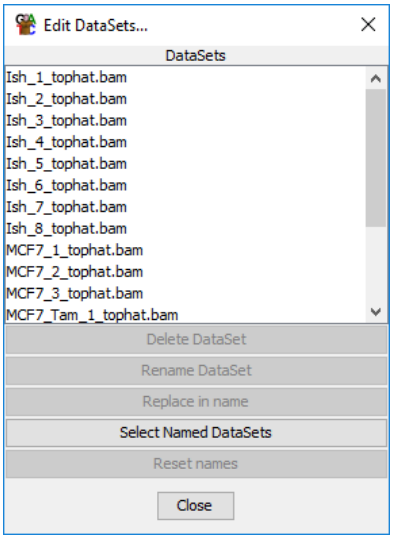
1. Tidy up the sample names
2. Create appropriate replicate sets

We did both of these in the previous exercise, but now that we have more samples it will be a pain to go through and rename and group them all individually so we're going to take some shortcuts.

Step 1.1 Renaming samples

At the moment all of the samples have their original BAM file names attached to them. This makes them unnecessarily long and will clutter up some of the figures we are going to generate. We can make things cleaner by removing the unnecessary parts of the sample names.

We're going to use a bulk rename to do this. You can access this from:



`Data > Edit Data Sets`

We are going to remove the "_tophat.bam" from all names. To do this first select all of the data sets in the top part of the dialog (you can use Control+a as a shortcut to do this). Then use "`Replace in name`" and then opt to replace "_tophat.bam" with "" (nothing). You should see all of the sample names in the chromosome view shorten.

Step 1.2 Creating Replicate Sets

We are also going to use a shortcut to generate our replicate sets. In this experiment we have planned ahead and made sure that the file names which were generated earlier in the processing were informative and contained the name of the condition which they came from. We can use this information to automatically generate our replicate sets.

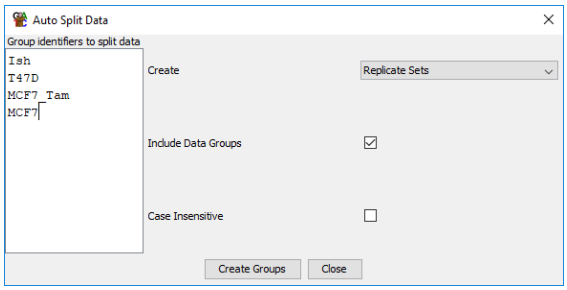
The 4 groups we are working with are:

1. Ish
2. T47D
3. MCF7_Tam
4. MCF7

To divide our samples we can use:

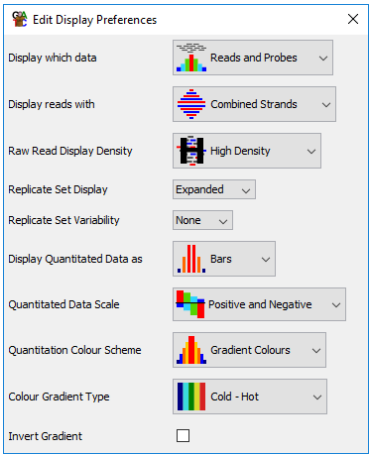
`Data > Auto Create Groups/Sets`

We can then fill in the parts of the DataSet names which define the replicate sets and SeqMonk will automatically create the sets for us. Each DataSet will only be assigned to one replicate set, and it will check from the top of the list down and stop when it finds the first match.



In our case what this means is that we have to put MCF7_Tam ahead of MCF7 otherwise samples will be mis-assigned to the wrong group.

Once you've created the replicate sets you will get a pop up to say how many samples were entered into each group. You should check that the numbers reported look correct, and then go to **Data > Edit Replicate Sets** to check the assignments look correct.



Finally, once we've created our replicate sets we can change the main chromosome view to show the sets instead of the individual, unlinked data sets. We still need to see the individual samples though, so after using **View > Set Data Tracks** to just show the replicate sets, we can use **View > Data Track Display** to set **Replicate Set Display** to **Expanded**.

Exercise 2: Initial Quality Control

The first step in any analysis should be to explore the data you have been given. Use the same approaches you used in the smaller analysis to explore this new dataset to work out how the data is behaving and to see if there are any technical issues we might need to address.

Spend some time looking around the data. Think about the following questions:

1. Does this look like RNA-Seq data?
2. Do the positions of the mapped reads look correct against the transcript annotations?
3. Does the data look directional? If so, is it same-strand or opposite-strand?
4. Is there any evidence of PCR duplication?
5. Is there any evidence of DNA contamination?
6. Are there obvious places where the expression differs between samples? If so, do these align with the annotated sample groups?
7. Are these samples male or female? Are they all the same?

As well as a general exploration you can use the **RNA-Seq QC Plot** and the **Duplication Plots** to help answer these questions. For the duplication plot, remember you will need to make probes over exons before constructing the plot (it doesn't matter for the plot how they are quantitated).

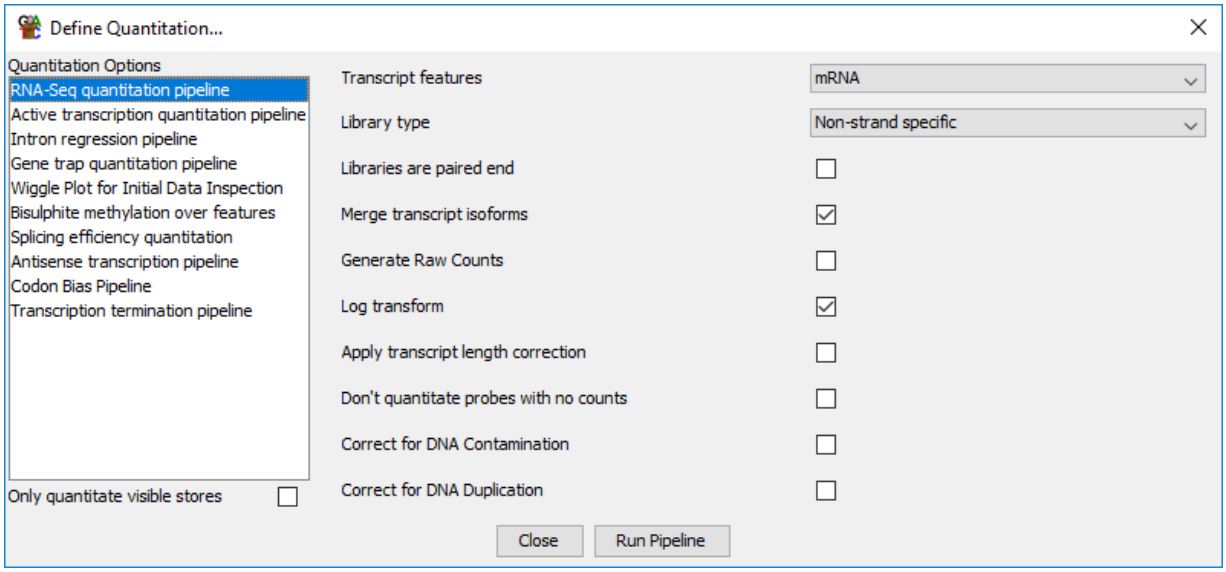
Exercise 3: Quantitation and Sample Clustering

Step 3.1 – Performing an initial quantitation

Once we're happy with the quality of the data and we have confirmed the directionality of the library we can proceed to do an initial quantitation. The point of this analysis is to look for differential expression, so we're always going to be comparing the same gene in different samples. This means that we won't need to apply any transcript length correction.

For our initial exploration we're going to want to work with log transformed, normalised counts. We will switch to raw counts later for some of the statistics, but now we want values which are easy to work with.

We again use the `RNA-Seq Quantitation Pipeline` to quantitate our data. We're going to use the standard mRNA features, and we're doing a gene-level quantitation (merging transcript isoforms). Our libraries are not strand specific, so we don't need to set that up. We didn't see any evidence for DNA contamination or PCR duplication so we don't need to apply any additional correction for those.



Step 3.2 – Checking the normalisation

After performing the quantitation have a quick look at the quantitated values in the chromosome view. Just check that the magnitude of the measures matches with the raw reads, and that the positions of the probes make sense with the locations of the genes (remember you should be seeing gene-level features as we merged transcript isoforms).

We also want to look at the distributions of our values to check that the automatic normalisation worked well for this data. The easiest way to do this is with:

`Plots > Cumulative Distribution Plot`

If the plots don't completely line up then are they far enough apart that you might want to apply additional normalisation? Could they be better aligned if lines were moved up or down?

Step 3.3 – Manual similarity assessment

Now that we have a set of quantitative values we can start to look at the similarity of our different samples. This will perform two functions:

1. We can check to see whether there is any evidence that biological replicates are showing significant variation from one another.
2. We can check our different sample groups and get an idea of whether we can see genes which are obviously differentially expressed.

As well as gaining a clearer impression of the nature of the changes within our data, we should also be looking for technical artefacts which might cause our samples to group inappropriately. We should also be open to the possibility that two or more samples could have been swapped and should look for reassurance that our annotations are correct.

To look at this open up:

`Plots > Scatterplot`

..and then try looking at different comparisons. Make sure you look at some pairs of data sets which come from the same sample group and from different sample groups. You can also compare some replicate sets to each other to look for overall differences.

When looking at a plot remember that you can double click on any point to look at the underlying data. Pick some larger changes and make sure the data behind them is convincing.

Step 3.4 – Automated sample clustering

To get a more holistic view of the relationship between the samples we will use some automated clustering to look at the overall similarity between them.

We are going to do this in a fairly crude way, using all of the quantitative data we have available. In a more nuanced dataset we might want to remove globally lowly expressed genes to clean up the similarity measures used to compare the samples.

The sample clustering tools are all under:

`Plots > Data Store Similarity`

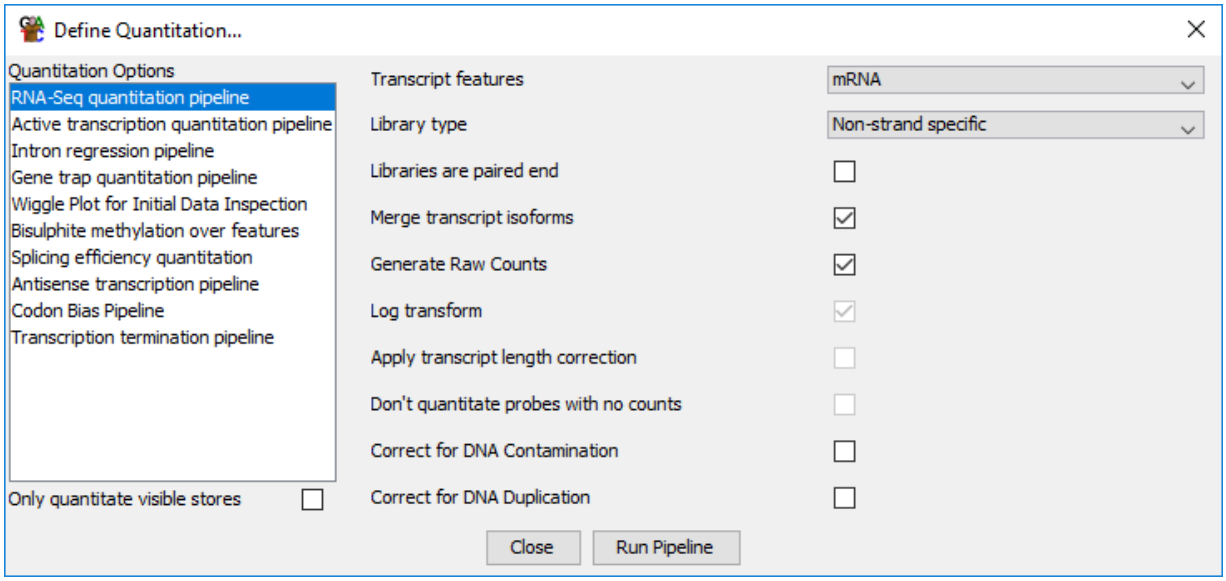
There are several different plots available and you can try a few to see if they give the same impression. Where it is an option, try colouring the plot by the replicate sets to make it easier to see if samples from the same set group together.

Exercise 4: Identifying Differentially Expressed Genes

We would like to find genes which are differentially expressed between our different conditions, and our exploration certainly suggests that these should be plentiful. In this design we are not performing a pairwise analysis, but instead are going to make a single comparison of all 4 conditions to find genes which change anywhere.

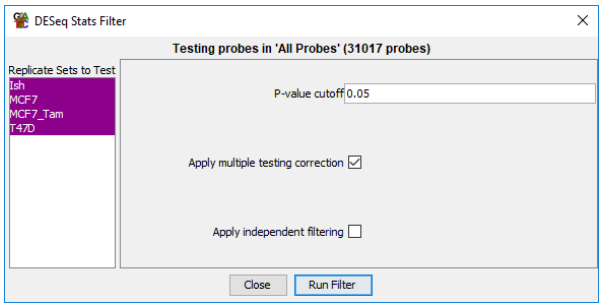
Step 4.1 DESeq analysis

As before we can use DESeq to make this comparison, but using a slightly different test to the pairwise comparison we used previously. Since DESeq uses a binomial based statistic we need to re-quantitate our data as raw counts before we can run the test. We therefore need to re-run the **`RNA-Seq Quantitation Pipeline`**, but opting to generate raw counts:



Once this is done we can run the DESeq analysis. This can be accessed under:

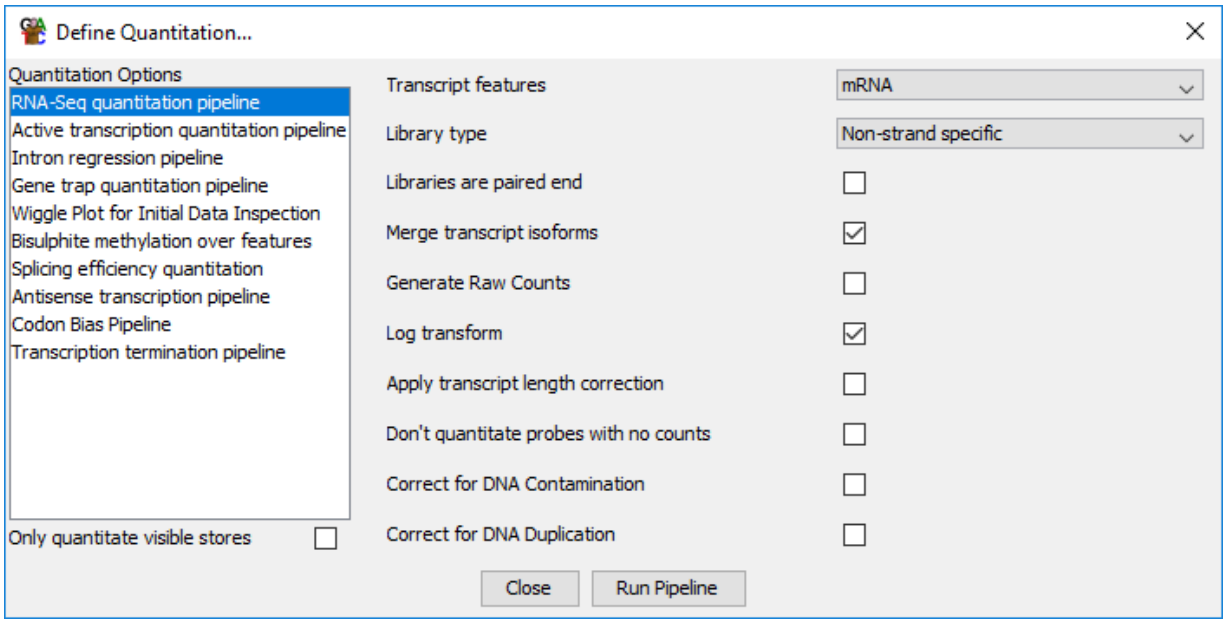
`Filter > Filter by Statistical Test > Count based statistics > Replicated Data > DESeq2`



The only practical difference in the way we run this is that instead of selecting 2 replicate sets, we can select all 4. Behind the scenes what this will mean is that DESeq will using the "Likelihood Ratio Test", which can work across multiple groups (kind of like an ANOVA for count data), instead of the original Wald test, which works for pairwise comparisons.

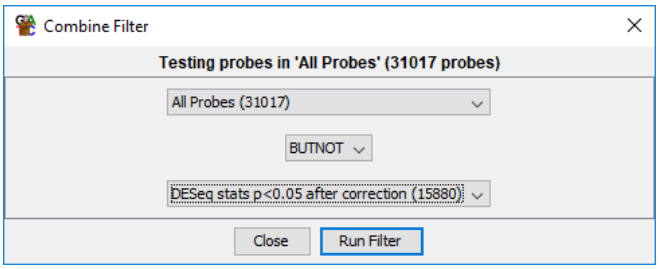
Once you've run the test and saved the hits, have a look at the number of hits which were generated. Think about whether this number will be practical to take forward for further analysis.

Once you have generated a list of hits you will need to re-run the **`RNA-Seq Quantitation Pipeline`** again (for the last time you'll be happy to hear!) to get back to log transformed normalised counts which we can use for visualisation.



Once you have your normalised quantiation back you can re-run some of the scatterplots between replicate sets, highlighting the DESeq hits to see whether they make sense. It might also be helpful to generate a list of genes which were not selected as hits and should therefore have (roughly) equal expression across all conditons. You can make this using:

`Filtering > Logically Combine Existing Lists`



Do the hits seem to make sense?

Step 4.2 Intensity Difference Hits

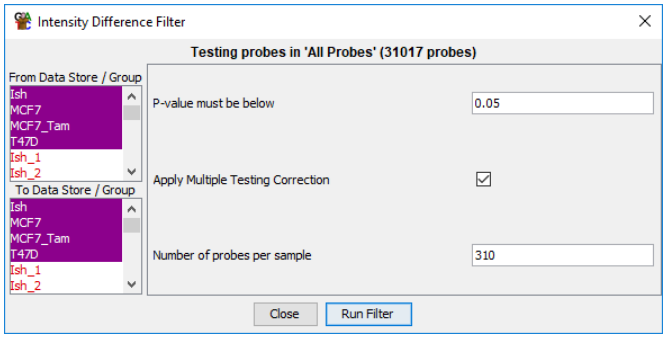
Since we got way too many hits from DESeq we are going to use the Intensity Difference Test to reduce the number of genes we need to deal with. Again we can perform a single test which will compare all of our groups and should give us a much smaller hit list to work with.

The intensity difference test works from the **Normalised** quantiations, so we can keep our existing quantitations.

You need to check that you have the **All Probes** probe list selected before running the intensity difference test. This is the data from which the background model is built, so you definitely don't want to run this just on the DESeq hits.

To run the test select:

`Filtering > Filter by Statistical Test > Continuous value statistics > Unreplicated data > Intensity difference`



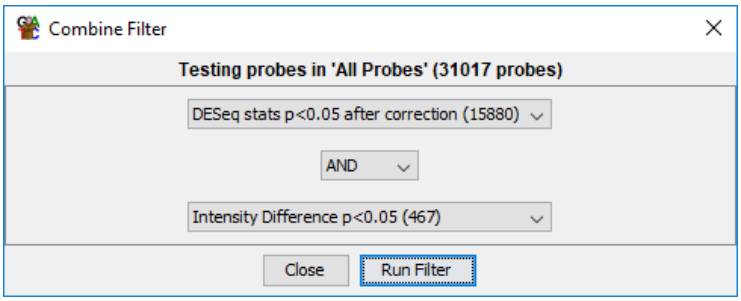
This might seem a bit odd since this is (originally) count data which has replicates, but we're working here with the properties of the data used for the test. Our normalised data is continuous (no longer just counts) and although we have replicates, the test is just using the mean value.

As before, select all of the replicate sets and run the test.

Do we get a much smaller list of hits now?

The hits we would really like are those which were hits from both the intensity difference and DESeq statistics, so create a probe list which contains only hits in both tests. This will again use:

`Filtering > Logically combine existing lists`



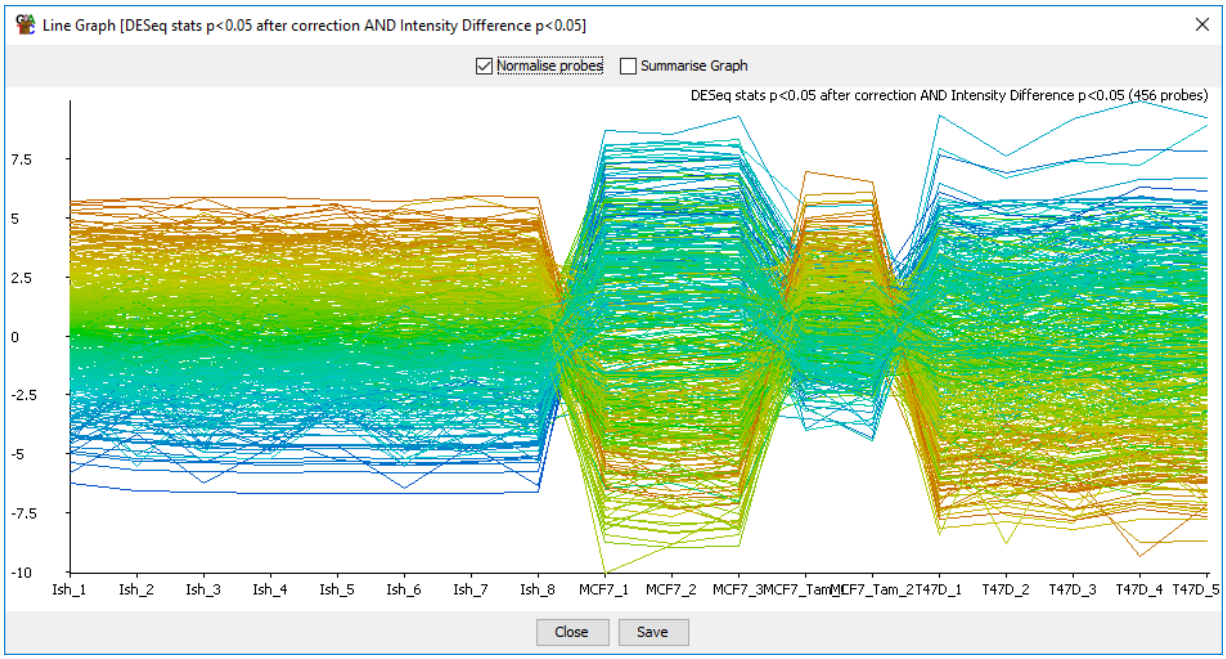
What is the overlap between the two lists like?

As with the DESeq hits, have a quick look at the DESeq + Intensity difference hits in the scatterplot view.

Exercise 5: Clustering and separating gene groups

You should now have a list of a few hundred genes which change markedly between your 4 groups. You could take these forward for biological interpretation as a set, but the problem is that those genes will not all be behaving similarly to each other. You can see this if you draw a line graph of the DESeq + ID hits across all of your samples.

`Plots > Line Graph`



We'd like to clean this up a bit before completing the analysis. To do this we're going to cluster the genes and separate out sub-groups which behave similarly across all samples. To do this, firstly we will do a visual clustering of the data.

Step 5.1 Hierarchical Clustering

Select your DESeq + ID hit list (should be a few hundred genes) and then run:

`Plots > Hierarchical Clusters > Per-probe normalised`

This will generate a heatmap of the expression patterns of your genes, with the genes being clustered down the y-axis. You should be able to see that there are obvious sub-groups of genes within your hit set.

As well as visually inspecting the clustering of genes you can also see the categorical separation which is able to be performed by the program. If you drag the slider on the left up and down you set an R value cutoff which can be used to group the data. You should see the dividers running across the data moving their positions as you change the cutoff. Adjust the slider to a point where you think it cleanly separates the groups you can see in the plot, and then use

`Save clusters`

To create separate lists for each of the groups. There is a filtration step which sets a minimal number of genes per cluster so you don't get tons of clusters with only 1 or 2 genes in each.

How many clusters do you get?

How many of your starting list of hits were classified into a cluster?

Step 5.2 – Visualising clustered genes

Now that you have divided your original hits into sub-groups you can re-run the line graph view of the data, but this time plotting separately for each sub-cluster. This should show you have you have cleaner lists of genes to take forward for biological interpretation. To do this select:

`Plots > Line Graph > Multiple Probe Lists`

..and then select all of the clusters you created in the previous step.

Do you see a cleaner pattern within each cluster, and are the different clusters clearly different from each other?

Example Plots

So you know what you should be seeing here are copies of the plots you should generate in this practical:

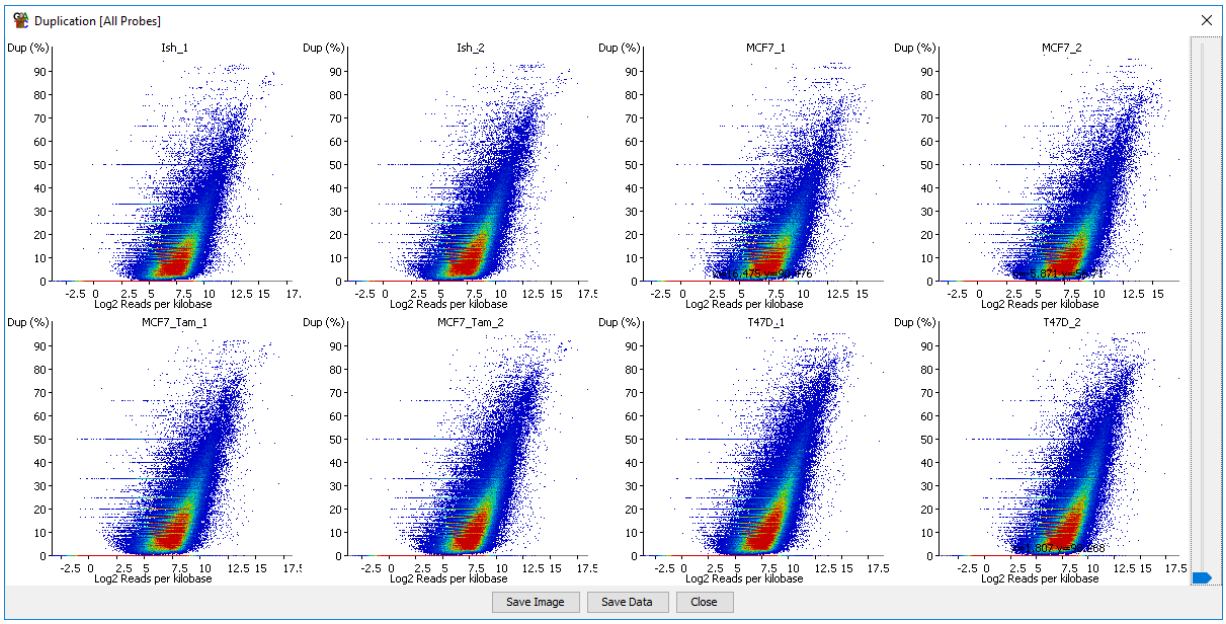
Exercise 2

RNA-Seq QC Plot

The quality here looks very good with no significant variation between the samples. The data size varies somewhat, but not consistently, and not by more than 2X from the lowest to the highest.

The final metric shows that this data is not directional. Everything is equally present on both strands.

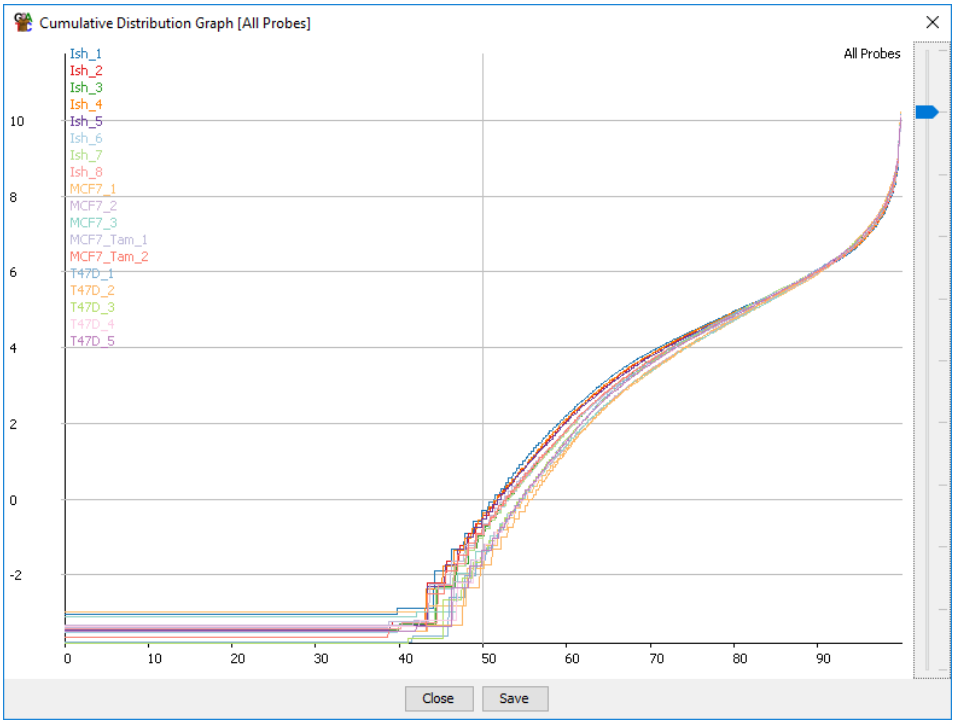
Duplication Plot (not all samples shown)



The duplication plot doesn't show any evidence of technical duplication.

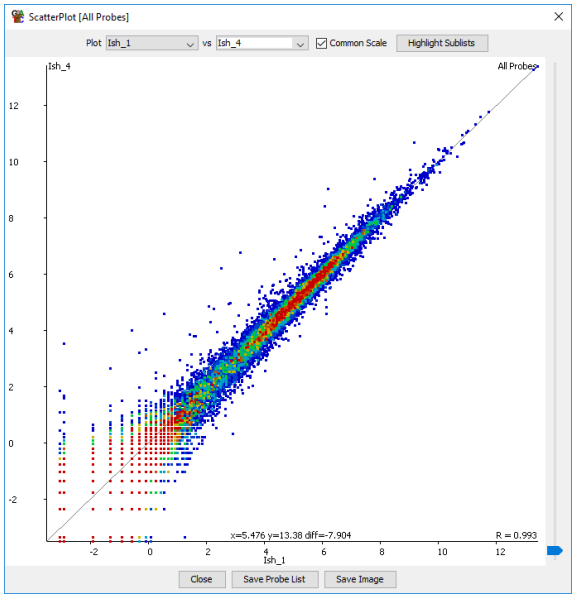
Exercise 3

Step 3.2 – Cumulative Distribution Plot



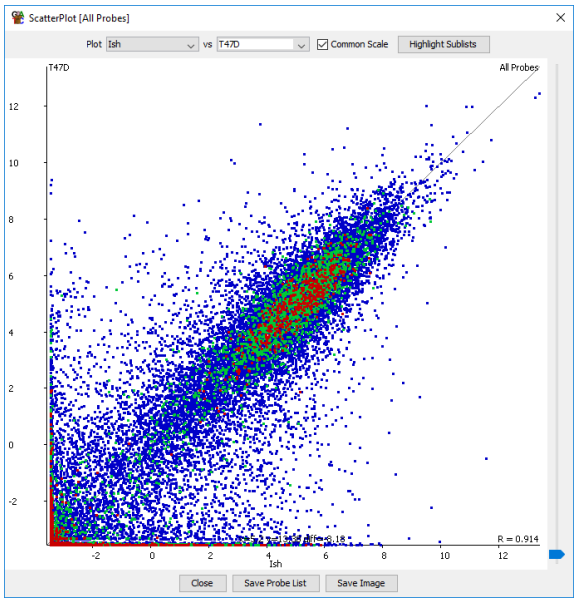
The data aren't perfectly normalised, but they're pretty close, and wouldn't be improved by moving lines up or down – suggesting that there won't be a simple fix which would improve the normalisation.

Step 3.3 – Scatterplot comparisons



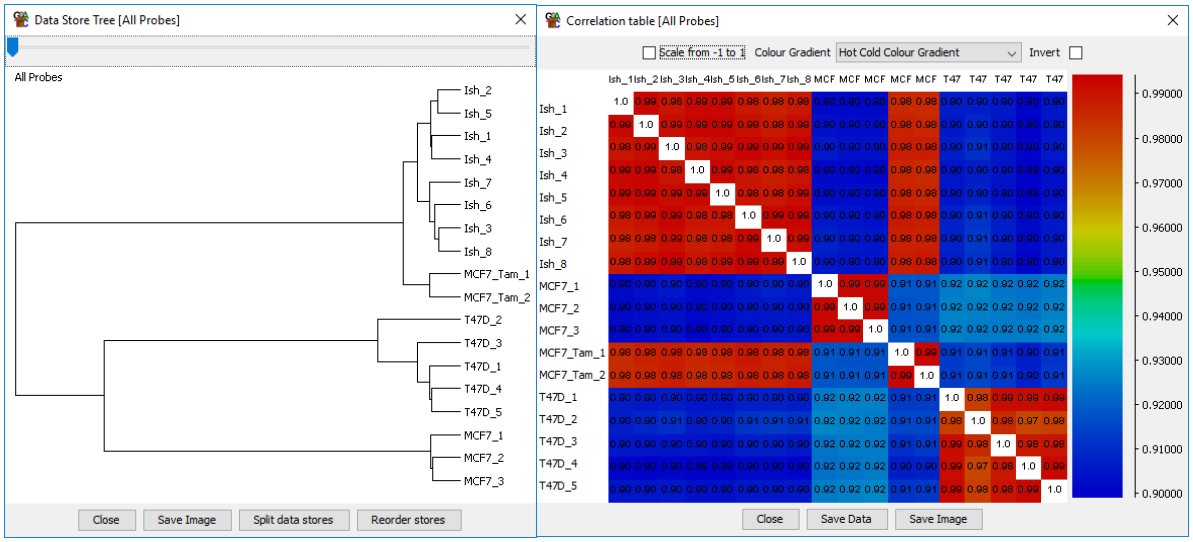
This is a comparison of two Ish samples, so we expect that they are going to be very similar to each other.

We can see that overall they do look very similar, but we can also see that there are a number of genes whose expression changes markedly between the two samples, suggesting that some biological or technical variation exists between our replicates.

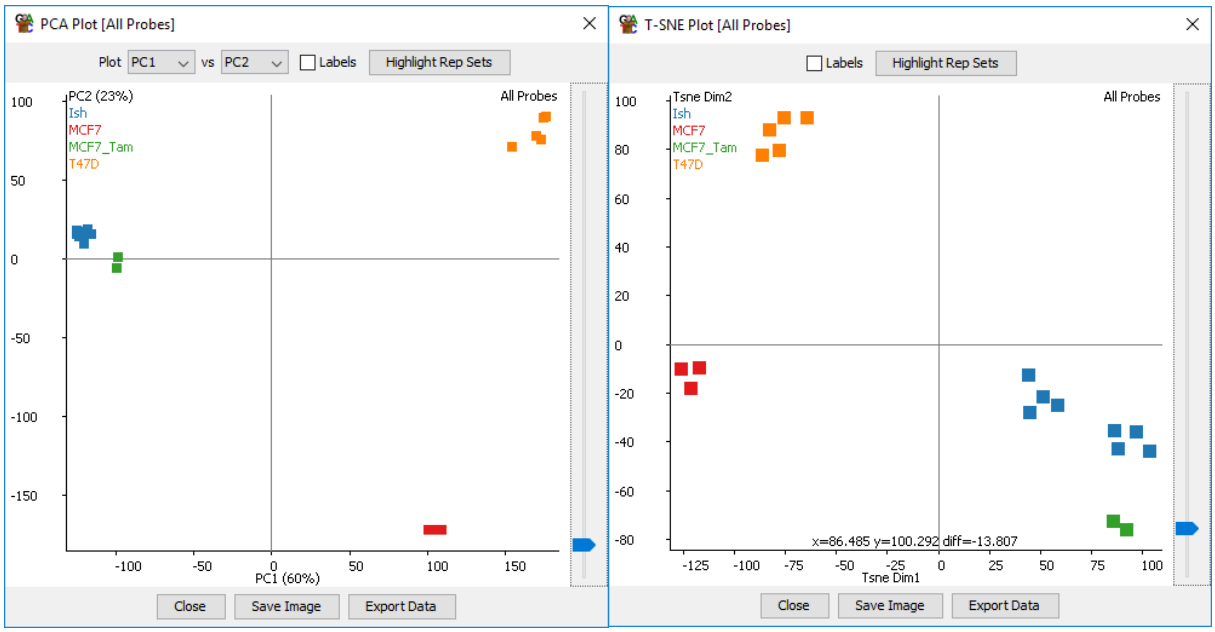


This is a comparison between two different replicate sets. We don't know what level of difference to expect, but we can clearly see that the two samples are hugely different to each other. This would suggest that we should be able to find lots of differentially expressed genes, as long as the differences are consistent. We are assuming that the experiment was properly designed and that the cell type differences aren't conflated with any other batch effects, otherwise we could have technical effects which we are unable to control for.

Step 3.4 – Automated Clustering



The correlation plot and the data store tree (built off the correlation matrix) both show two very different groups, with two separate sub-groups within each. The samples in each replicate set clearly group differently to each other.

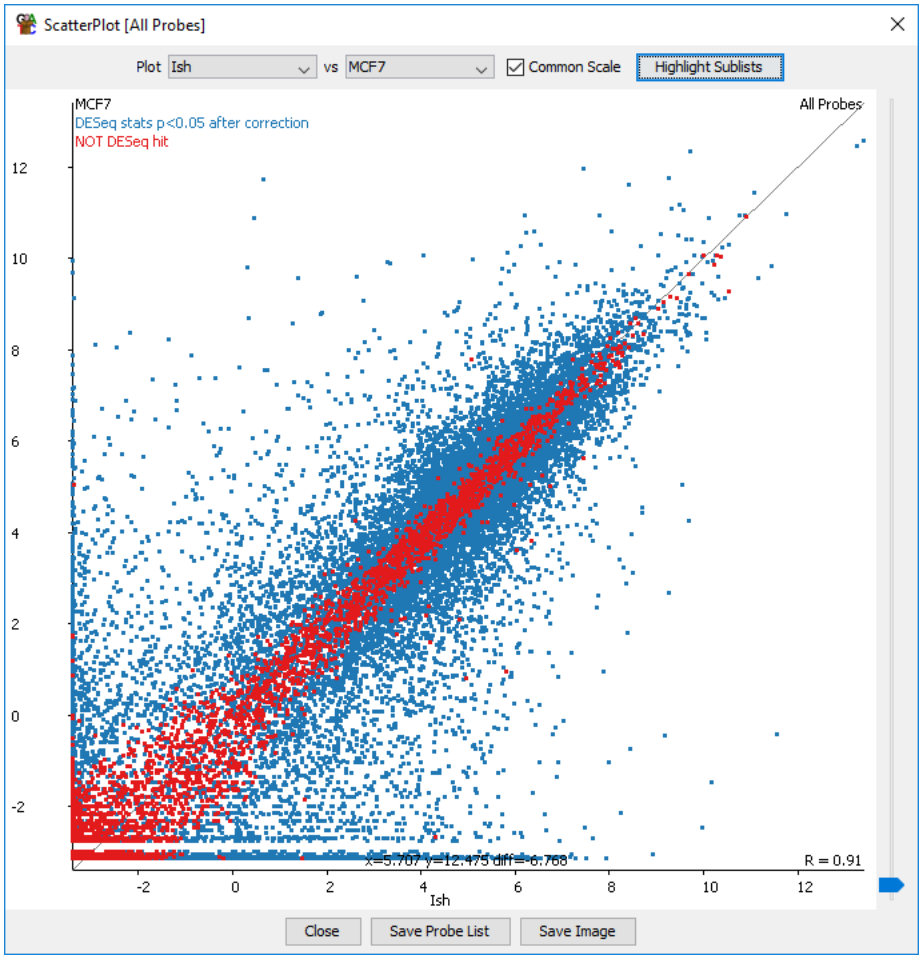


The PCA and tSNE plots both show the same clear separation as the data store tree. Both suggest that the MCF7_Tam group looks quite similar to the Ish group and, not so close to the other MCF7s. This is either an interesting biological observation, or a clue that there might be confounded technical artefacts in the data.

The tSNE plot shows some suggestion that the Ish group might split into two sub-groups, but this isn't supported by the PCA.

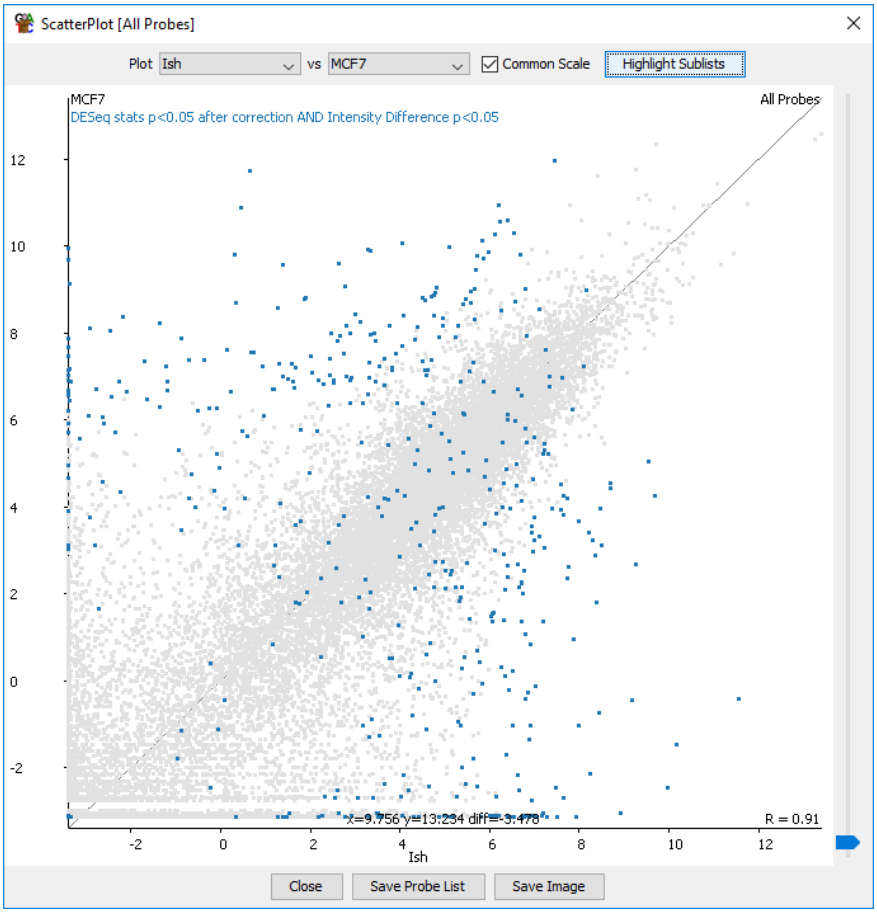
Exericse 4: Differential Expression

Step 4.1 – DESeq LRT Hits



You should be able to see that the genes which are not selected to indeed have very similar expression levels across all samples. The hits will be more variable though because of the number of comparisons made making these much more difficult to assess.

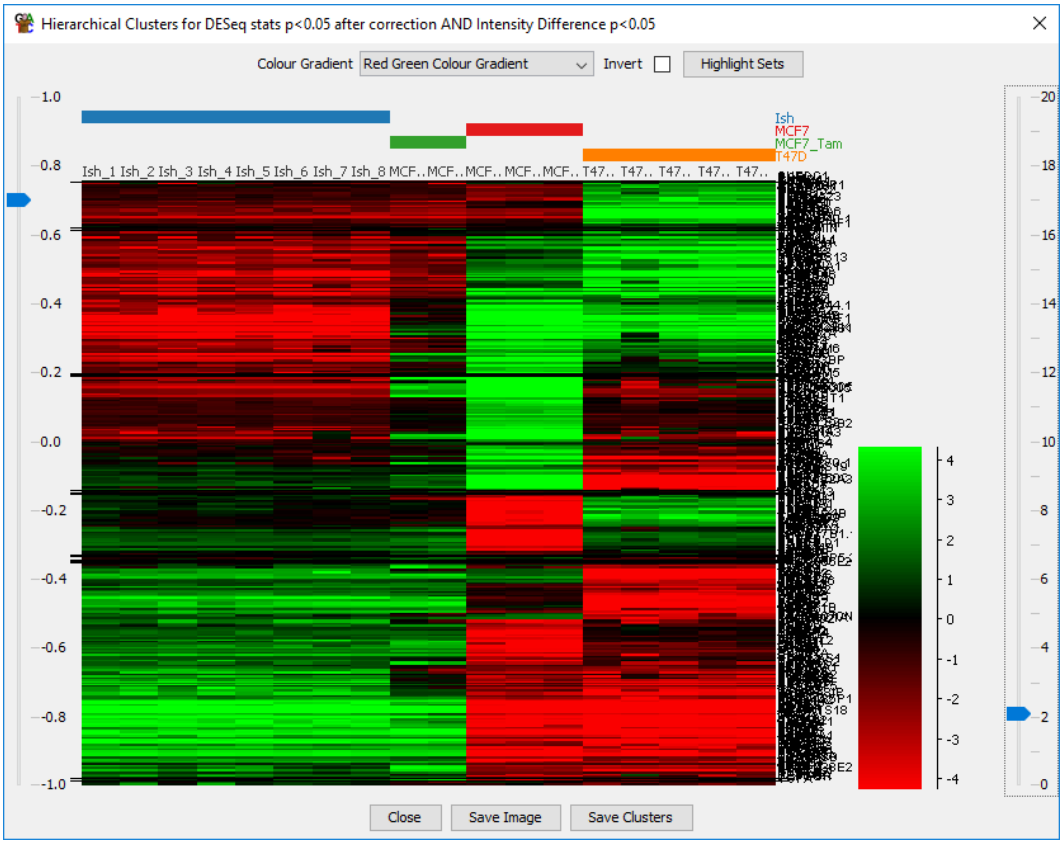
Step 4.2 Intensity Difference (plus DESeq) Hits



You should see a much more restricted set of hits, but one which contains all of the most extremely changing genes. If you see genes which are changing by a large amount but weren't selected then it is likely that they vary a lot within one of the replicate sets.

Exercise 5: Clustering and grouping genes

Step 5.1 – Hierarchical clustering



Clear sub-groups of genes should be visible. In this version of the plot I changed the order of the replicate sets in the chromosome view to put the MCF7 next to the T47D since these were the most similar overall.

Step 5.2 Line graphs of separated clusters

After separating out the different clusters you should be able to more clearly discern consistent patterns within each cluster when plotting across all of your samples.