JAX Big Genomic Data Skills Training - 2018
RNA-seq Data Analysis Module

Y. Ada Zhan

Tutorial Overview

In this tutorial, we learn to perform basic RNA-Seq workflow and do differential expression analysis on Galaxy. To achieve these goals, we will go through the following steps:

- Data preparation (Steps 0 – 4)
- Sequence alignment (Step 5)
- Data visualization in IGV (Step 6)
- Count features (Step 7)
- Identify differentially expressed genes with DEseq2 (Step 8, 9)
- Gene enrichment analysis in GOrilla (Step 10)

Background

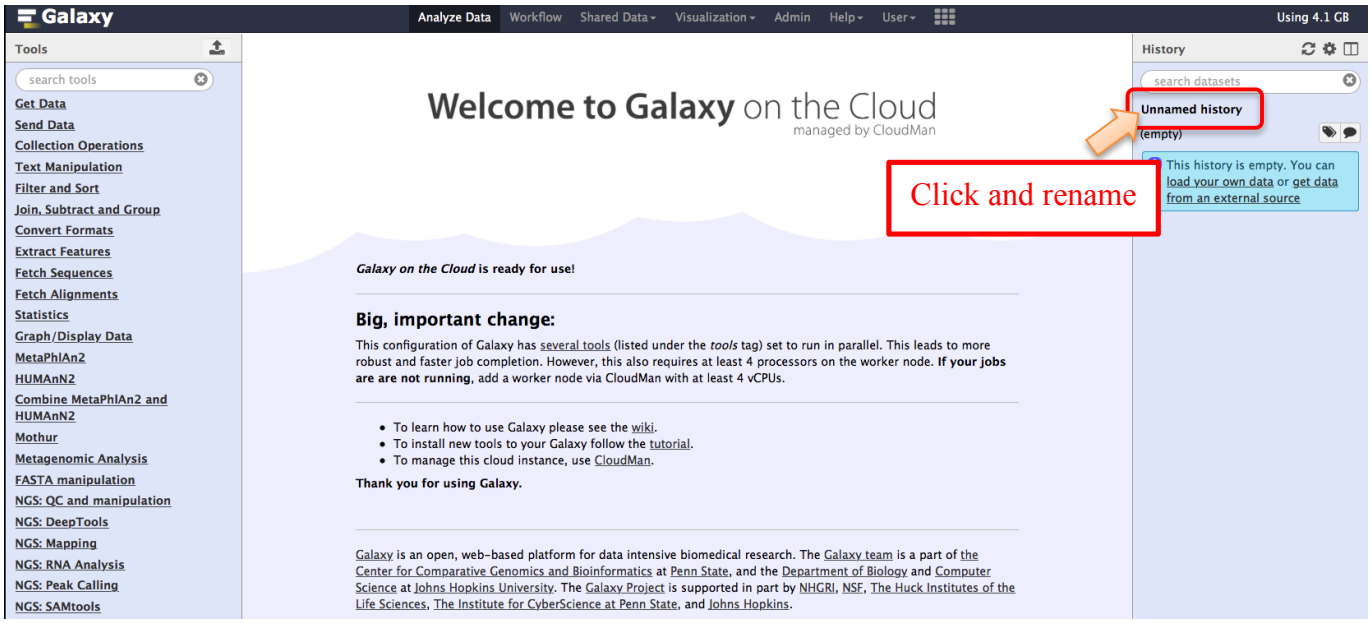
The dataset we are using is from the paper, The transcription factor Pax6 is required for pancreatic β cell identity, glucose-regulated ATP synthesis and Ca2+ dynamics in adult mice (http://dx.doi.org/10.1074/jbc.M117.784629). by Mitchell RK. The authors investigated significant transcriptional differences underlying the defective glucose-stimulated insulin secretion of Pax6 knockout mice in comparison to floxed littermate controls. In human, heterozygous mutations in the gene PAX6 lead to impaired glucose tolerance. Embryonic deletion of the Pax6 gene in mice (http://www.informatics.jax.org/marker/MGI:97490) causes loss of most pancreatic islet cell types. In this study, the authors revealed that in adult mice the inactivating Pax6 genes leads to reduced expression in many key beta cell genes but increase in some other genes that contribute to the reduction total islet insulin content.

From this study, we are using the RNA-Seq data which is publicly available in ArrayExpress https://www.ebi.ac.uk/arrayexpress/experiments/E-MTAB-5708/samples/. Sequencing was carried out on the Illumina HiSeq-4000 for pair end reads, and the libraries are reverse stranded. More experimental details can be found at https://www.ebi.ac.uk/arrayexpress/experiments/E-MTAB-5708/. In the tutorial, we will use the data for male mice that includes two replicates each for wildtype and beta cell Pax6 knockout (KO) conditions. We will align the data to mouse reference genome, identify the differentially expressed genes due to Pax6 knockout, and perform a simple gene set enrichment analysis. To fit the whole analysis into manageable time frame, we have prepared the data to only contain chromosome 2 (chr2) that hosts the Pax6 gene.

The data is associated with the following database records on ENA https://www.ebi.ac.uk/ena.

Study accession: PRJNA327115
Run accession: ERR1950095, ERR1950098 (WT), and ERR1950099, ERR1950101 (Pax6 KO)

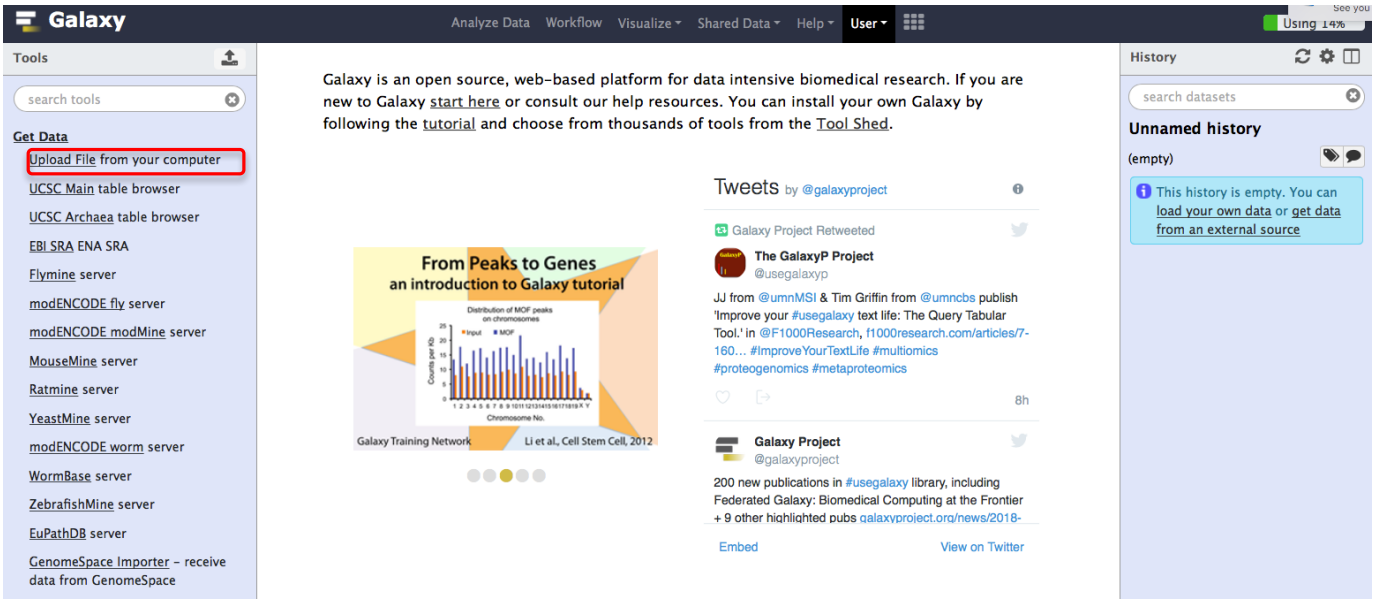
Step 0: Open up your Galaxy (Please register yourself first)

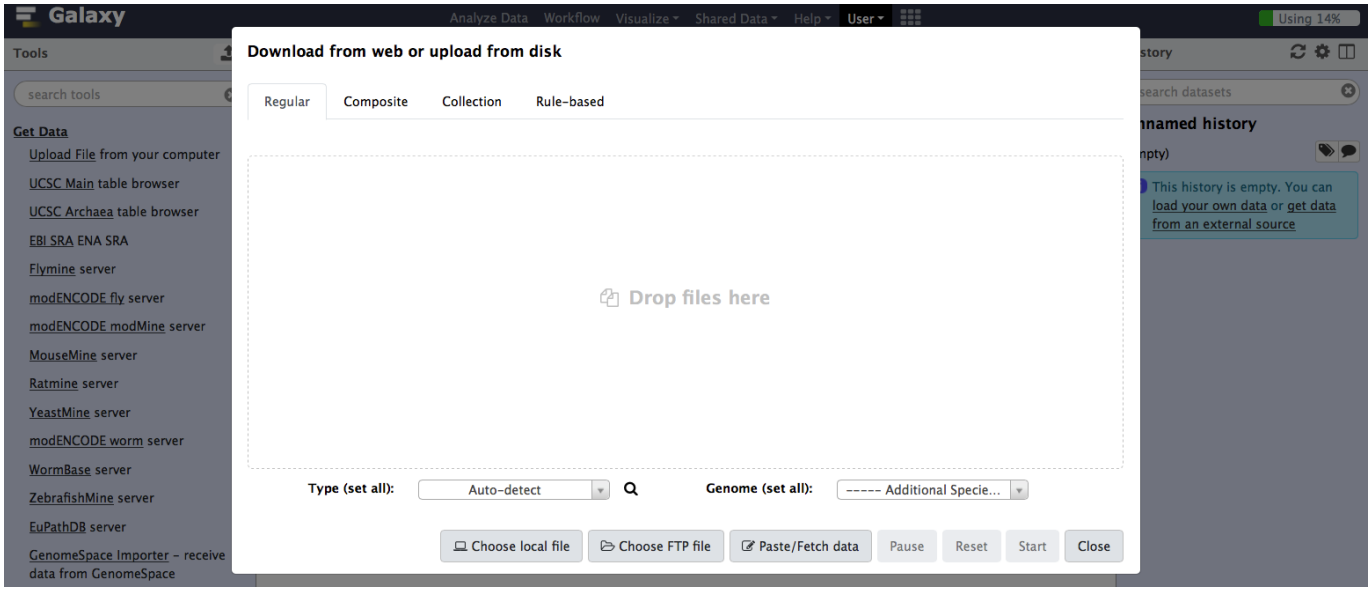


You may create a new history by click the little ⚙ on the right side of 'History'. Rename to 'Pax6_KO_mouse'.

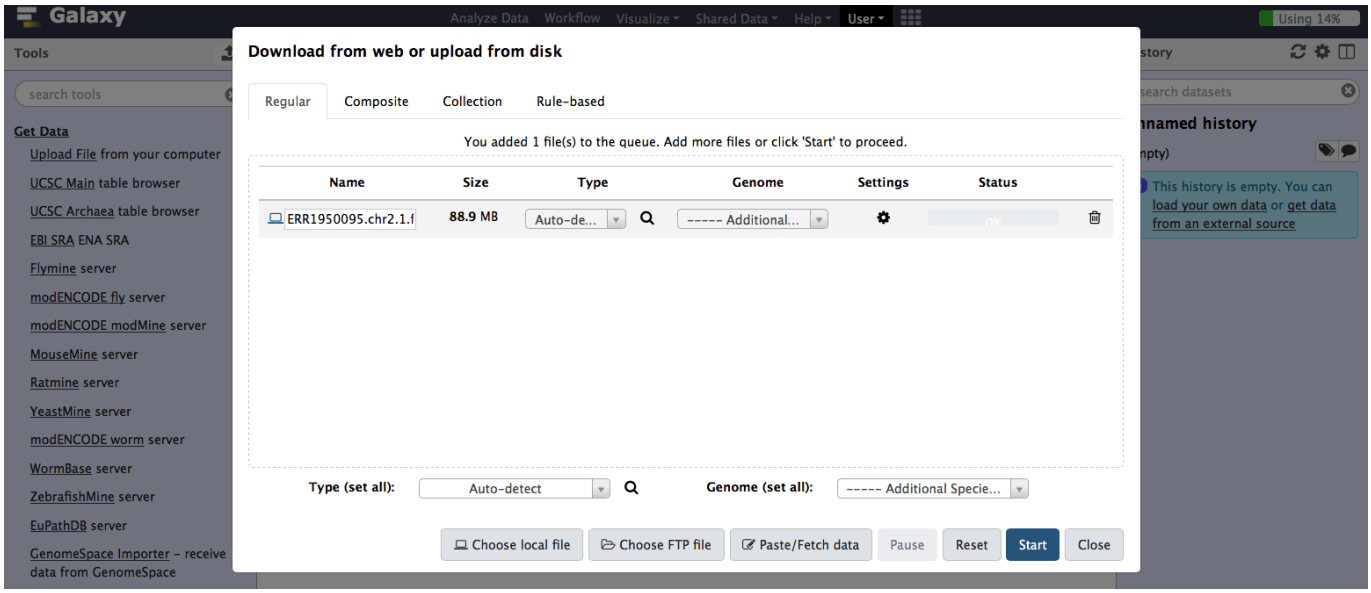
Step 1: Upload the sequences.

There are number of tools under 'Get Data' tab. You may upload the data from your computer using 'Upload File' or you may use any specialized database tools if you know the accession number. In this tutorial, we have uploaded the data using 'Upload File'.

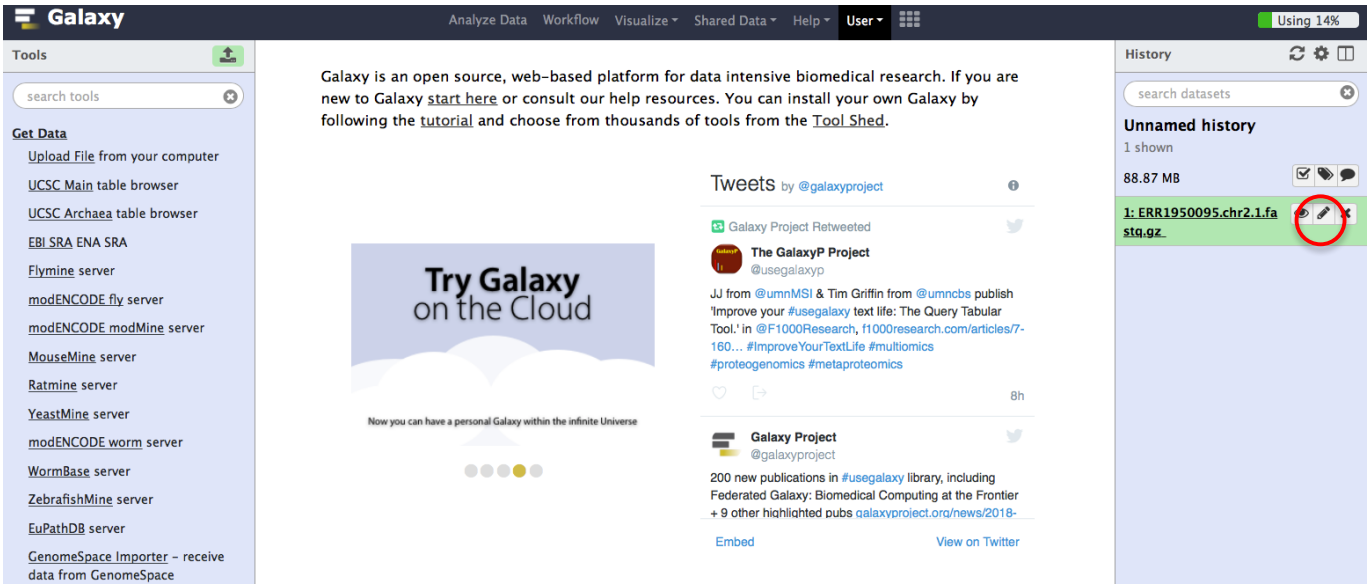




Now you may drag your files from your computer as directed.



Click ‘Start’. After a while the following interface will show up. Then you may change the name for easier track.



Modify the name following the rules below:

‘Genome’:

‘mm10.chr2.fa’: The reference sequence of mouse genome on chr2. mm10 or GRCm38 is the primary assembly released by Genome Reference Consortium in 2012. It can be obtained from UCSC http://hgdownload.cse.ucsc.edu/goldenPath/mm10/chromosomes/ site.

‘gencode.vM16.annotation.chr2.gtf’: The gene annotation for mm10 on chr2. It is the M16 GENCODE https://www.gencodegenes.org/mouse_releases/ version released in December of 2017.

‘RawData’:

‘betaPax6.KO.Rep1.chr2.1.fastq’: Pair-end RNA sequencing read 1 of Pax6 knockout mouse on chr2. Accession number ERR1950099.

‘betaPax6.KO.Rep1.chr2.2.fastq’: Pair-end RNA sequencing read 2 of Pax6 knockout mouse on chr2. Accession number ERR1950099.

‘betaPax6.KO.Rep2.chr2.1.fastq’: Pair-end RNA sequencing read 1 of Pax6 knockout mouse on chr2. Accession number ERR1950101.

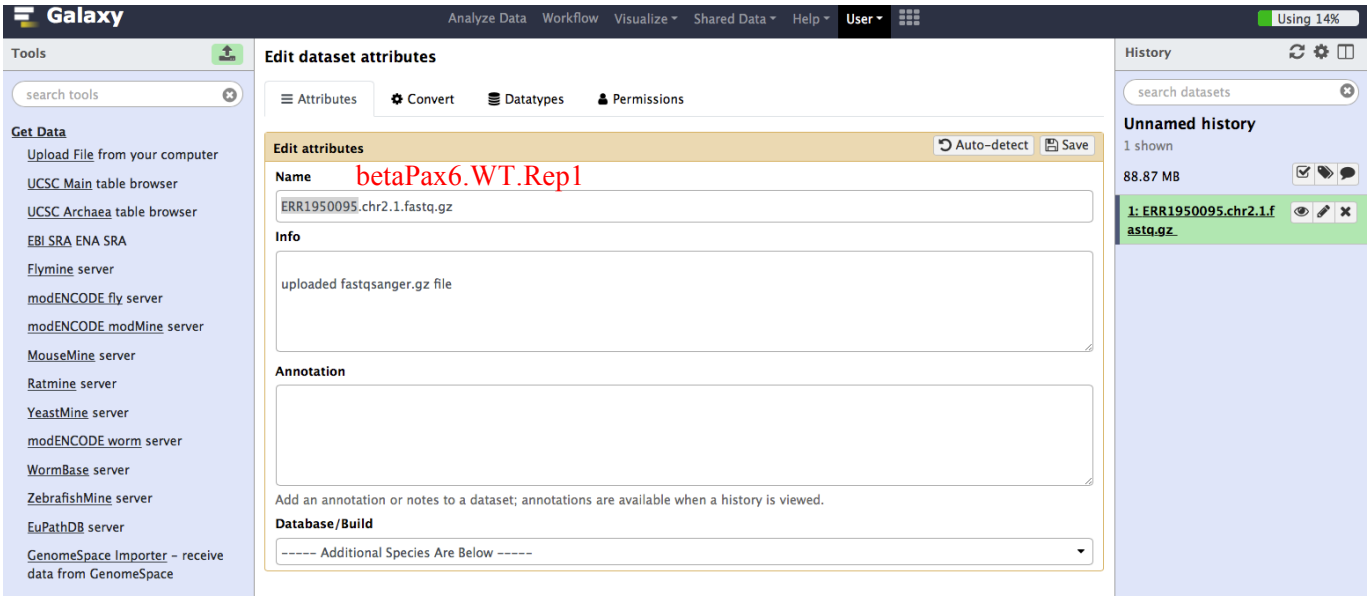
‘betaPax6.KO.Rep2.chr2.2.fastq’: Pair-end RNA sequencing read 2 of Pax6 knockout mouse on chr2. Accession number ERR1950101.

‘betaPax6.WT.Rep1.chr2.1.fastq’: Pair-end RNA sequencing read 1 of wildtype mouse on chr2. Accession number ERR1950095.

‘betaPax6.WT.Rep1.chr2.2.fastq’: Pair-end RNA sequencing read 2 of wildtype mouse on chr2. Accession number ERR1950095.

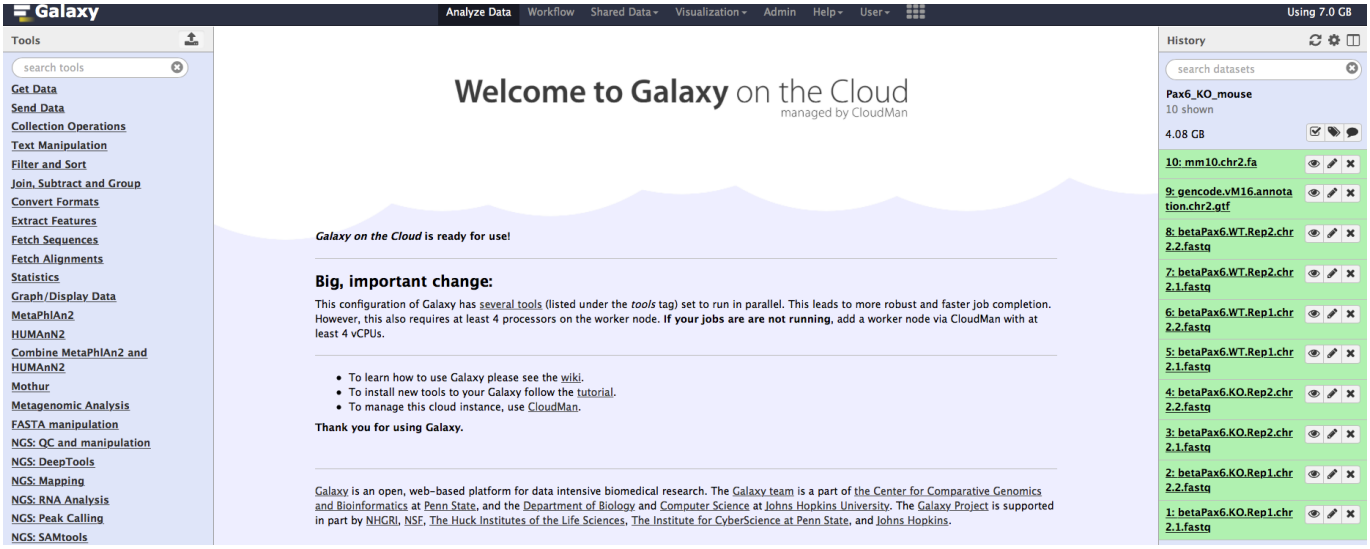
‘betaPax6.WT.Rep2.chr2.1.fastq’: Pair-end RNA sequencing read 1 of wildtype mouse on chr2. Accession number ERR1950098.

‘betaPax6.WT.Rep2.chr2.2.fastq’: Pair-end RNA sequencing read 2 of wildtype mouse on chr2. Accession number ERR1950098.



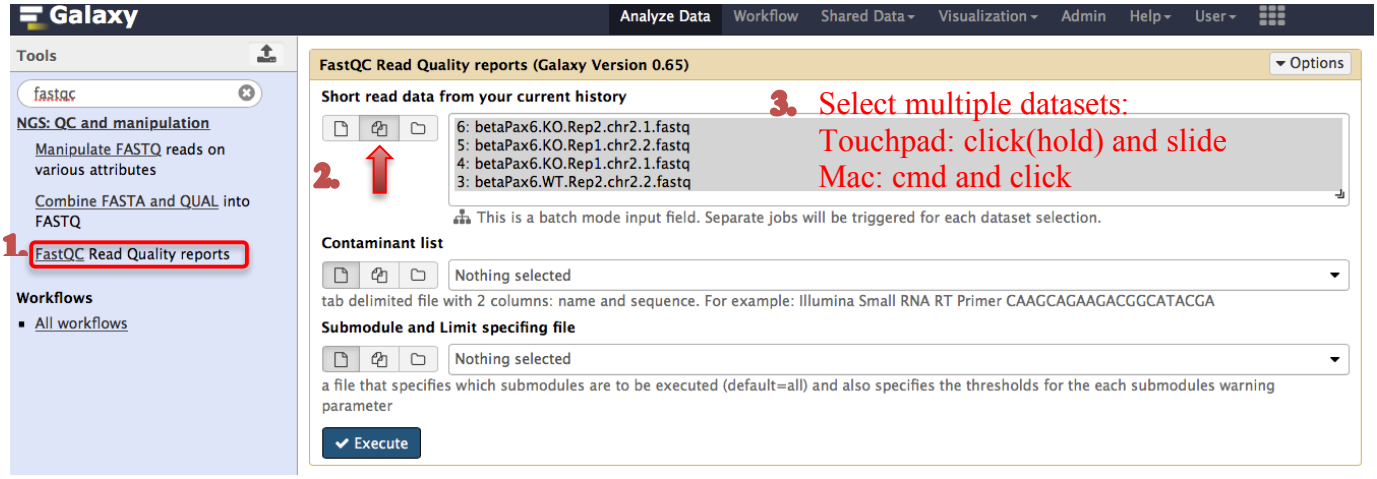
Then upload other files and rename them if necessary.

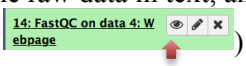
Now click 'Analyze data' to go back to your history. You will find the data are in your history and ready to run.



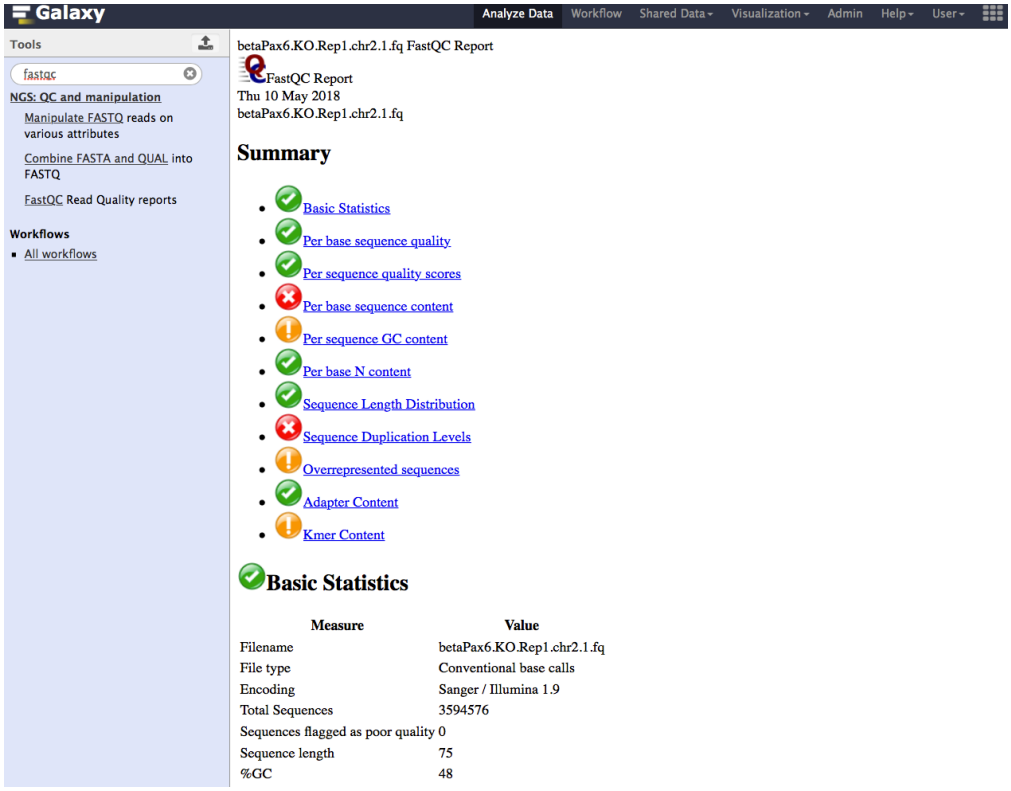
Step 2: Perform Data QC on each file

Use FastQC to check the quality of the data. Note that if your fastq files have not been set as type "fastqsanger" they might not be visible to many data processing tools on Galaxy. After assessment by FastQC, you will be able to tell whether the quality scores are Sanger Phred+33 or not. If it is, you may modify the datatype to "fastqsanger" directly if not yet. If it is not, you may want to run FASTQ Groomer to convert. The detailed steps can be found at Galaxy's help page https://galaxyproject.org/support/fastqsanger/. Different sequencing platforms normally have distinct quality score systems. Illumina pipeline usually produces "fastqsanger" format. In this tutorial, our data have been set to "fastqsanger" type.



FastQC analyzes multiple aspects of the input file, and allows for identification of systematic issues. Test details and interpretations can be found at https://www.bioinformatics.babraham.ac.uk/projects/fastqc/ . After running, each input file will have two associated output files, one with the raw data in text, and a second with an html-formatted visual presentation, as seen below (click eyeball to view 

):

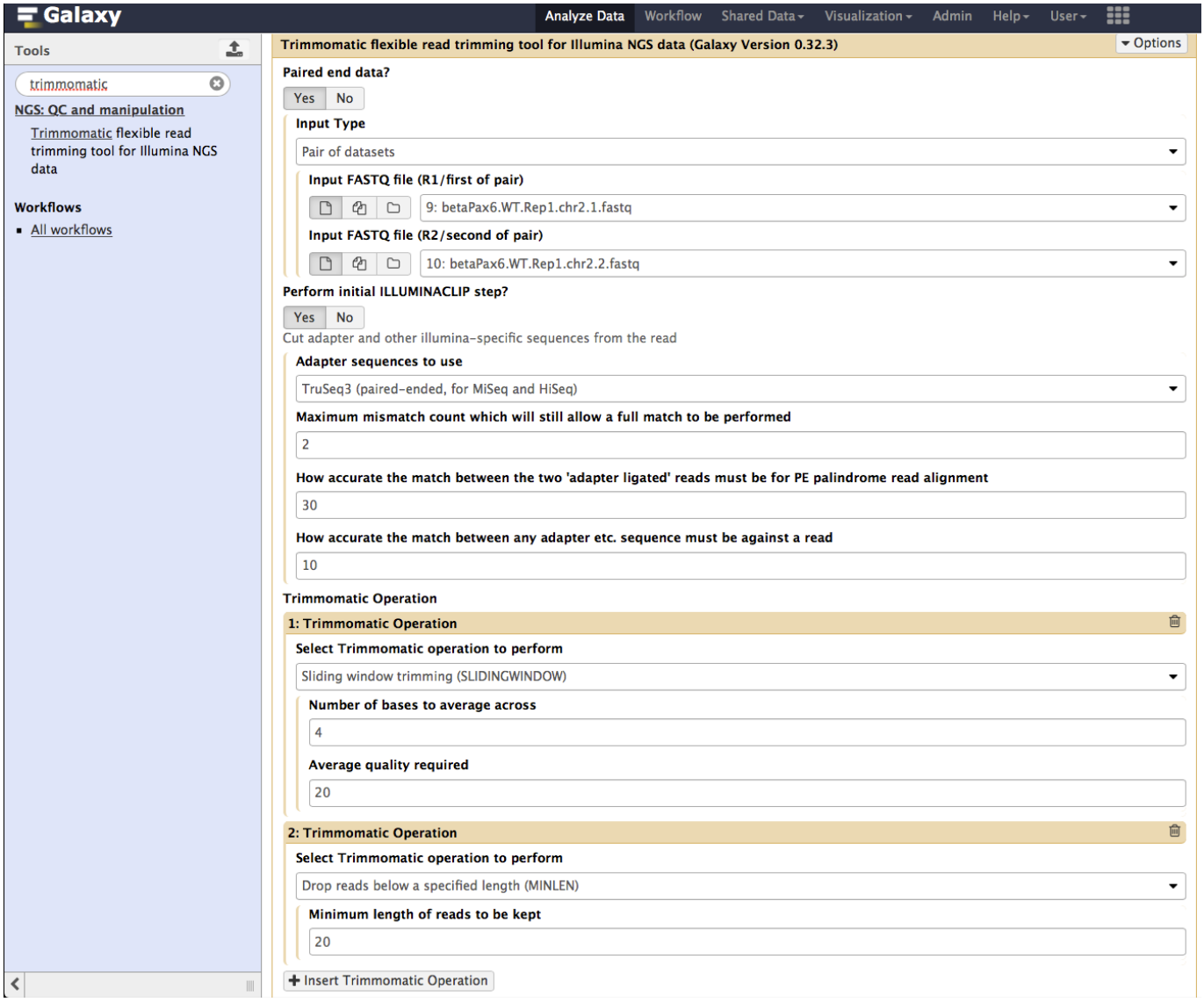


The top of the output page shows both statistics of the input file and a summary of the findings based on the analysis, with each of the individual test outcomes represented by either a green check (ok), a yellow exclamation point (warning), or red x (danger) icon. NOTE HOWEVER, that these settings are rather generic, and tuned towards what the authors of this program expect to see with a standard file (typically a mammalian RNAseq data set). The results for many of these tests can vary significantly, depending upon the nature of the data. You may still want to align your sequence regardless what you see in the FastQC report. Examination on the aligned results will help you decide whether you want to keep or toss the data.

Step 3: QC processing with trimmomatic.

Trimmomatic is one popular tool for removing low quality reads, adapters, or/and short reads and cut bases off the ends from NGS data.

A suggested set of settings for our data is shown below:



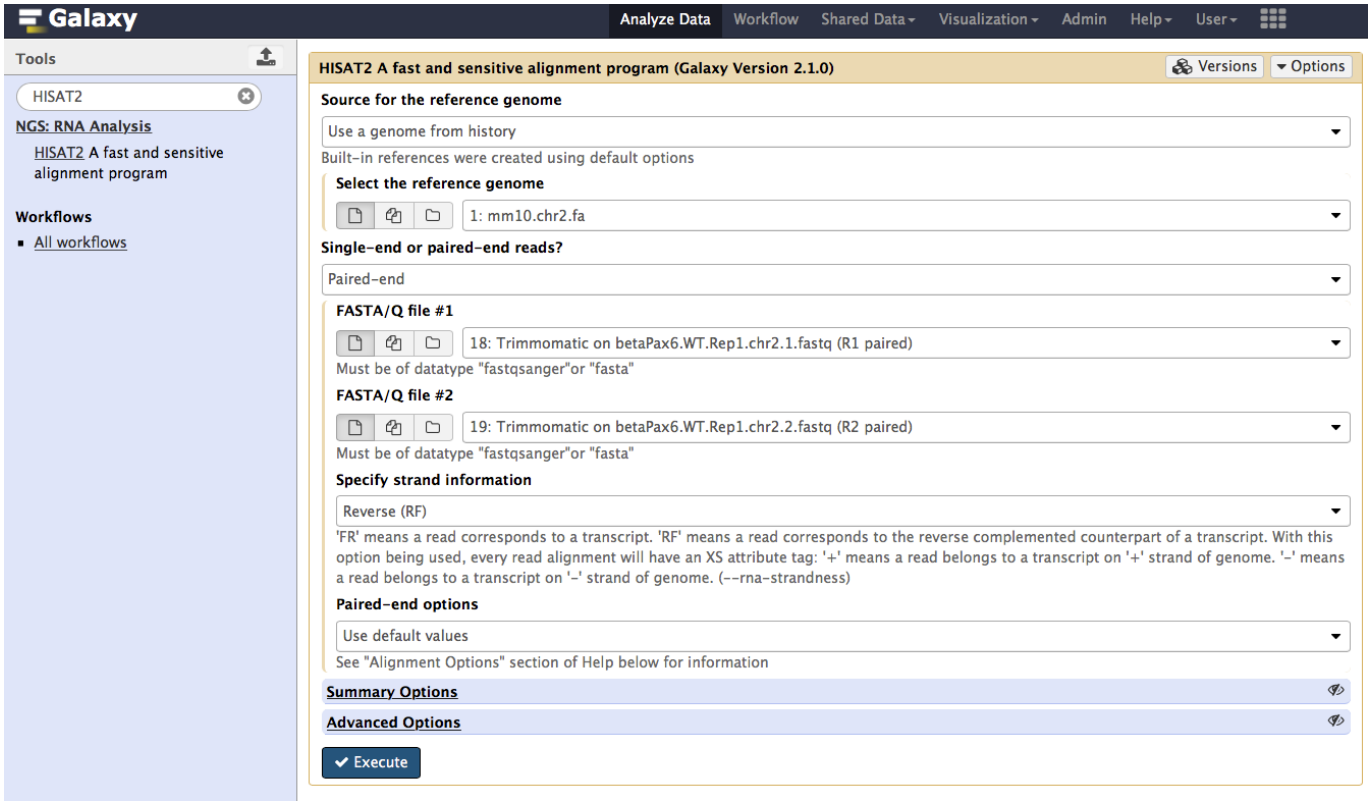
Trim or not? Currently there is a debate [whether trimming is necessary](http://www.ecseq.com/support/ngs/trimming-adapter-sequences-is-it-necessary). Many think it is unnecessary since our aligners are smart enough to remove those problematic sequences. Some others suggest we should keep this step at least for small RNA sequencing. It would an interesting question for undergraduate students to explore.

Step 4: Examine the updated fastq file with FastQC, using the same logic as for step 2.

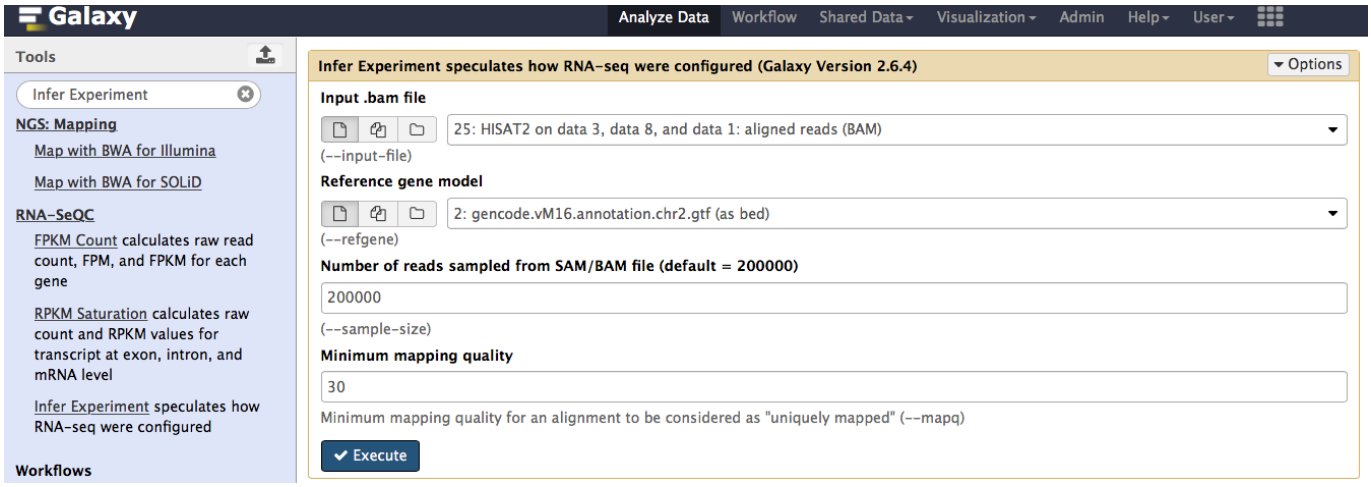
Have the tests changed and/or improved following the QC processing?

Step 5: Align the filtered reads from each of the data sets to the reference mouse genome using HISAT2

Alignment software develops very fast. Two popular aligners for RNA-Seq are HISAT2 and STAR. In this tutorial, we will use HISAT2 to save some memory. You are welcome to explore STAR afterwards.



Note: If you do not know how the experiment was run with no knowledge about the strand information, you may want to infer the strandness via RNA-SeQC. Based on the report, you may need to run the alignment again if you did firstly wrong. Result interpretation is shown in the page after you click the tool.

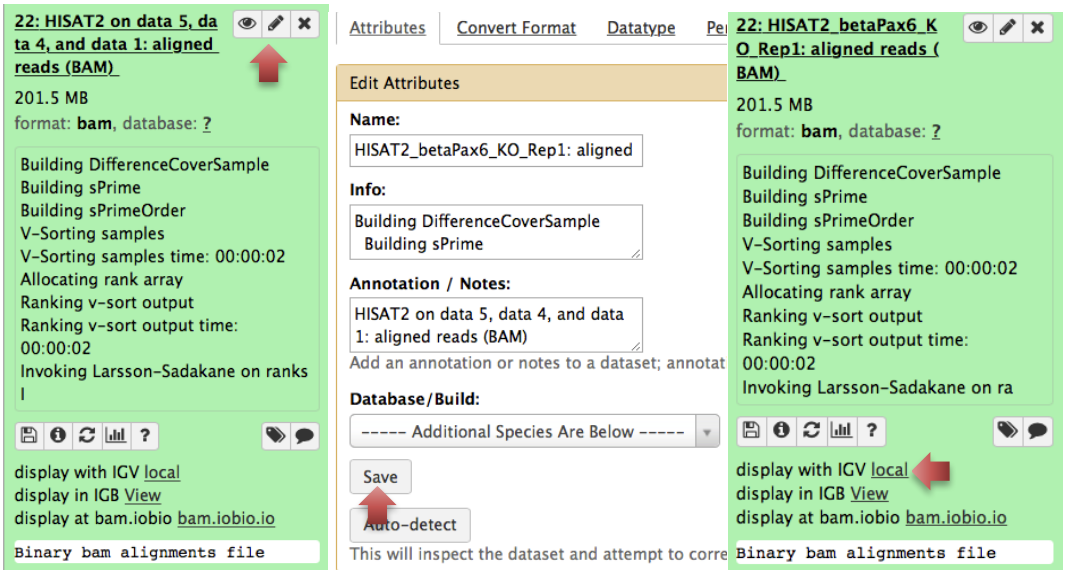


For our data, we know they were prepared using the Illumina TruSeq Stranded kit and they are reversed.

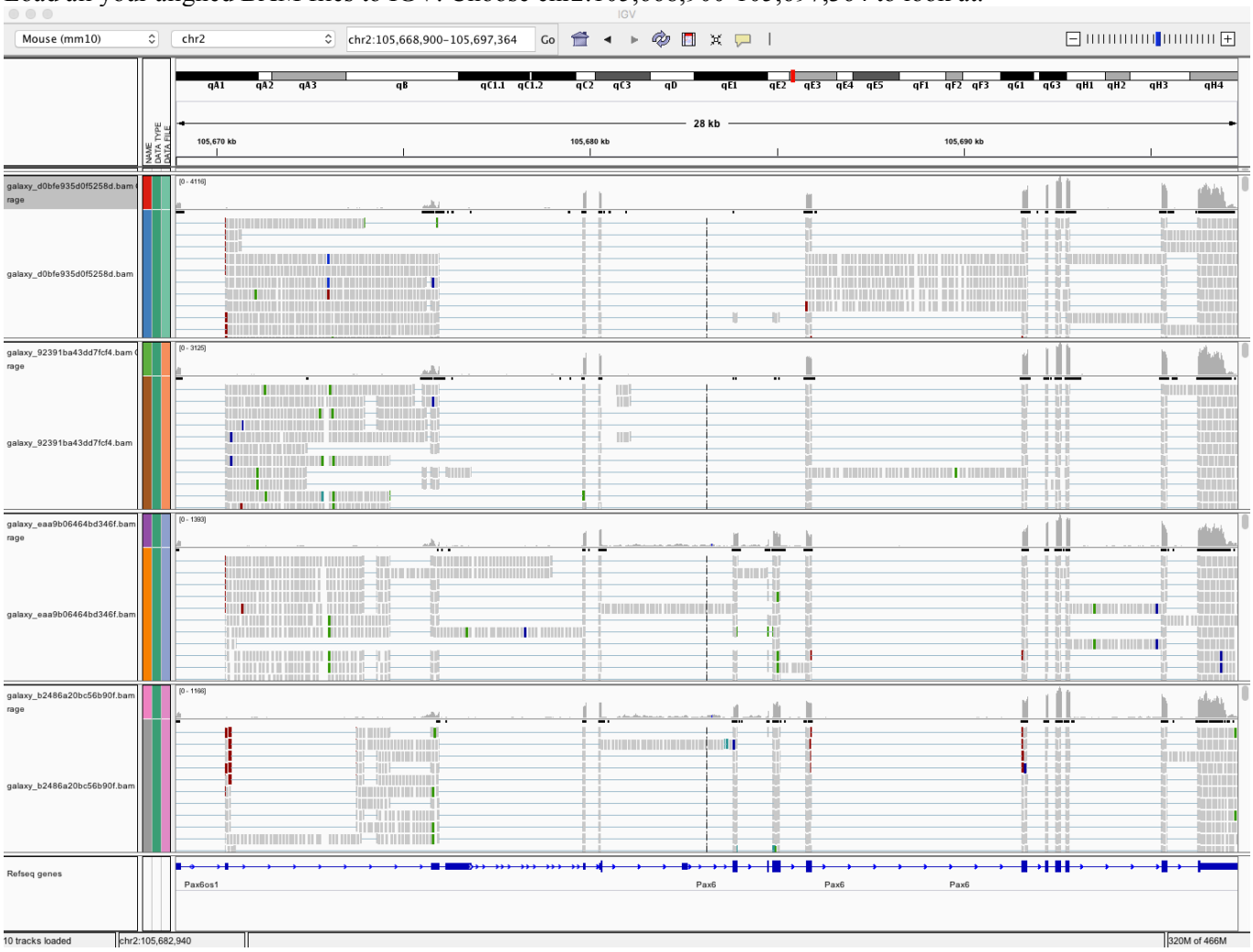
Step 6: Inspect the output files in IGV

Open your IGV and select Mouse (mm10) as your reference genome. 

To Better keep track of your data, we suggest you rename the steps in your history.



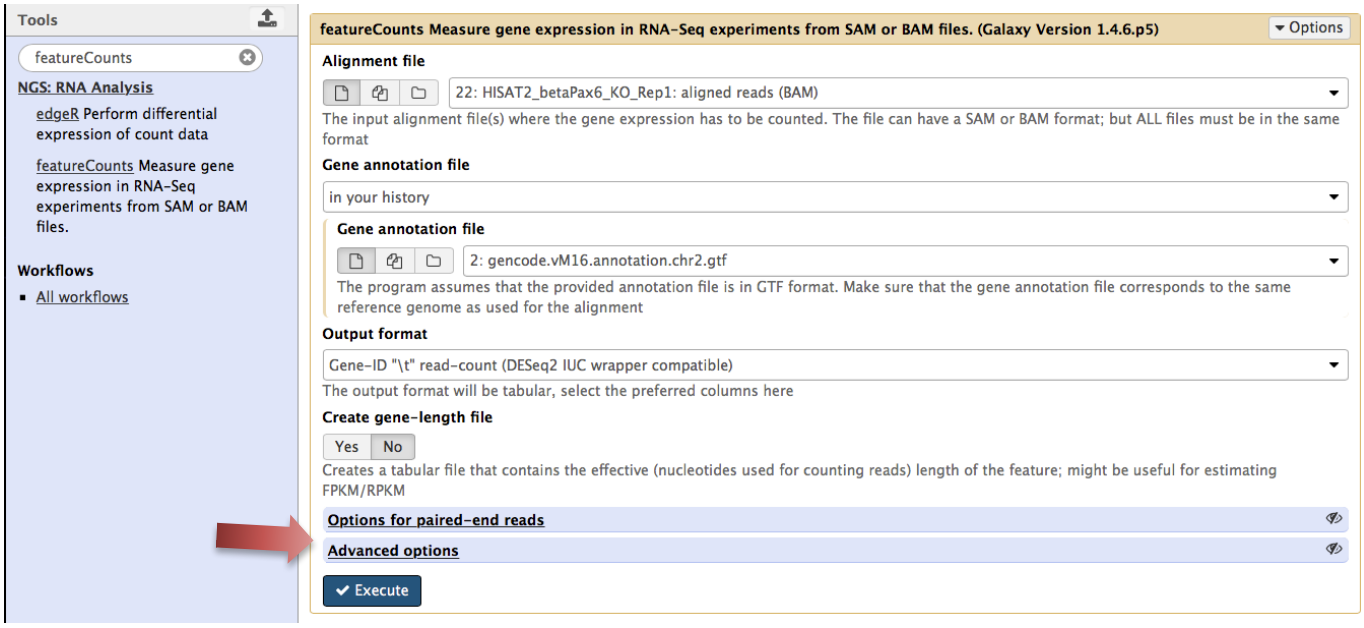
Load all your aligned BAM files to IGV. Choose chr2:105,668,900-105,697,364 to look at.

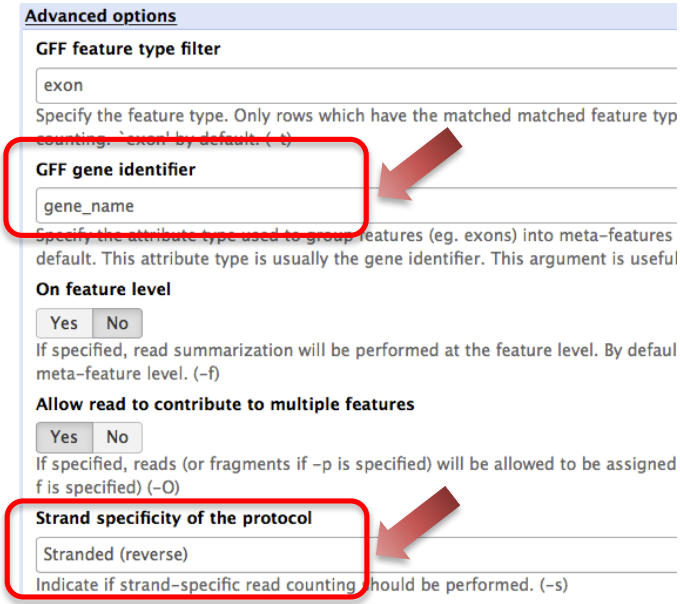


What do you observe?

Step 7: Count the features

For this step, we would like to know how many reads in each gene. FeatureCounts and htseq-count are two popular tools to achieve this goal. We will use 'featureCounts'.



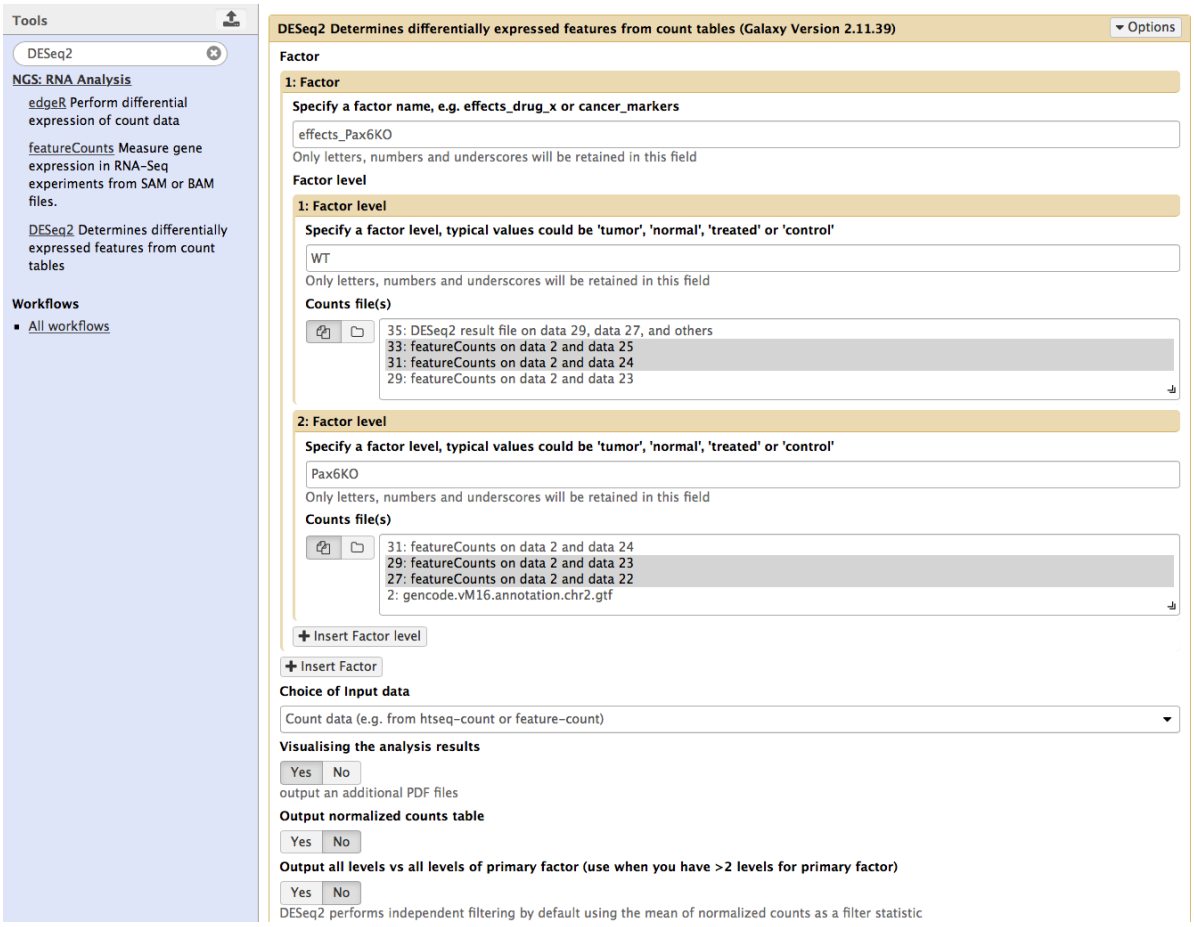


After run, the output looks like below:

Geneid	HISAT2 on data 5
Gm37392	0
Gm27306	0
Fam171a1	3667
Nmt2	5400
Gm22005	5
Rpp38	640
Acbd7	30
Olah	0
Gm37525	0
Meig1	326
Dclre1c	1329
Suv39h2	206
Gm13184	10
Hspa14	1840
Gm45902	211

Step 8: Differential expression analysis using DESeq2

DESeq2 uses a statistical approach based upon a "negative binomial" distribution the compare the counts of each transcript/gene between different samples (including replicates) to assign a probability to the observed counts being generated if the gene is NOT differentially expressed between conditions. Suggested setup for this analysis is shown below.



Pay attention to the data sets you are selecting. You may want to rename the previous steps to avoid confusion.

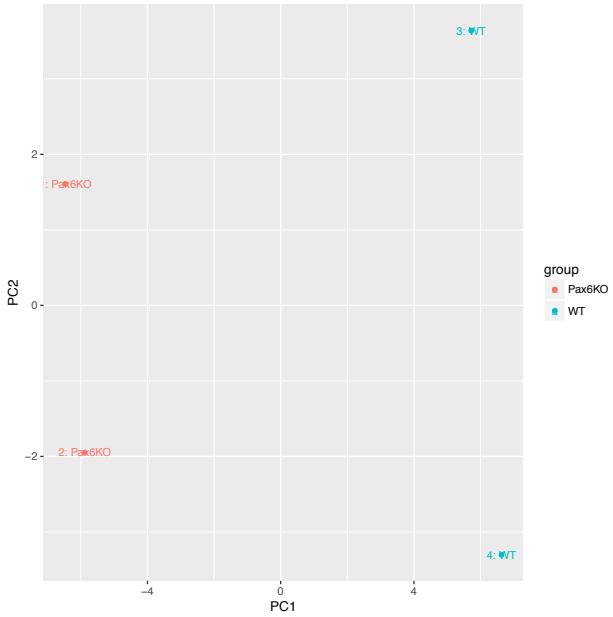
Make sure to select "Yes" for **Visualizing the analysis results**, as this produces very useful plots.

The option to "Output normalized counts table" is not necessary here, but it is very useful if the end user wishes to use the matrix of expression by sample to carry out further analysis such as hierarchical clustering or principal components analysis. This matrix differs from the input data in that it is normalized across samples to common input levels, and also transforms low-count transcripts/genes in a manner that reduces their influence on the overall results. See the DESeq2 documentation for further details.

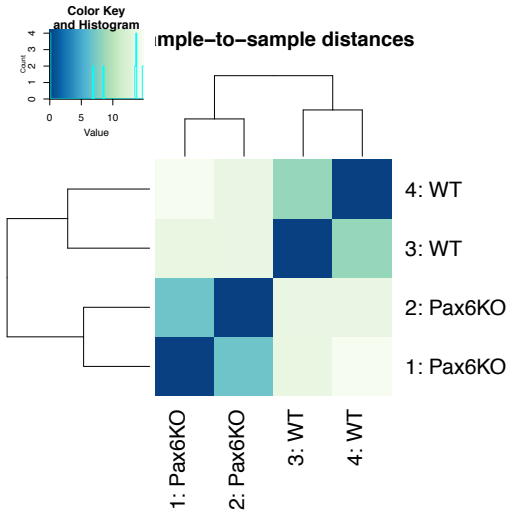
NOTE: A negative binomial distribution is similar to a Poisson distribution, which is commonly used in counting events, however, the negative binomial distribution has a wider variance, and more accurately takes into account the variation commonly observed between biological replicates. (In contrast, technical replication, such as found in either different sequencing lanes of the same sample, or even different library preparations from a common sample can be adequately modeled with a Poisson distribution.)

Step 9: Examine the output files and look for significantly differentially expressed genes.

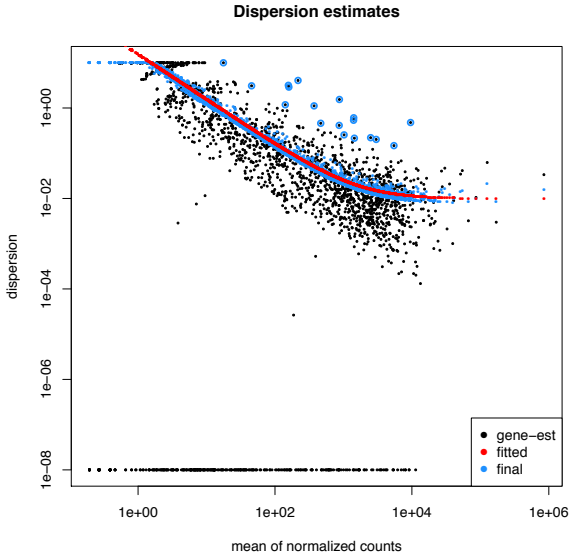
View the output file labeled "DESeq2 plots on data…" by clicking the eyeball next to the history record.
A principal components analysis of your samples is useful for exploratory data analysis. Samples which are more similar to each other are expected to cluster together. In our case, knockout (Pax6KO) and wild-type (WT) samples are well separated on principal components 1 (PC1, the x axis).



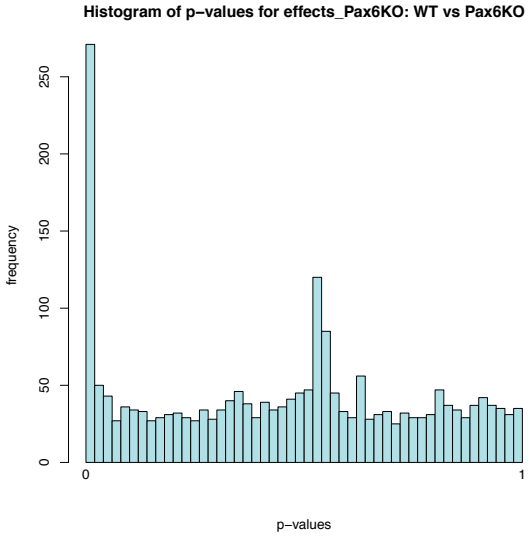
A hierarchical clustering dendogram plot showing distances between samples. Again, knockout and wild-type samples display highest similarity to each other.



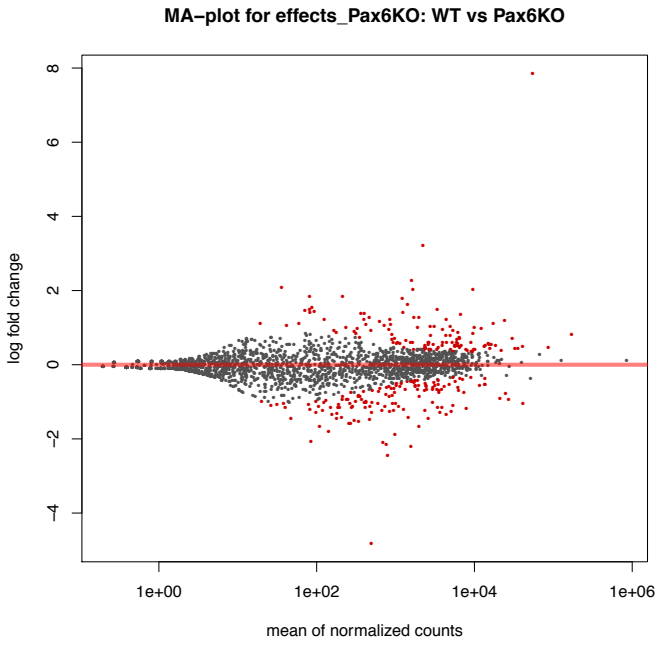
The dispersion estimates are trying to model the variability of expression between biological replicates. It provides the information on gene-wise estimates (black), the fitted values (red), and the final maximum a posteriori estimates used in testing (blue).



p-value histogram shows how many genes/transcripts received a p-value within ranges of 0.02 between 0 and 1. Low p-values suggest differentially expressed genes.



Finally, the MA plot is a scatter plot showing on the y-axis the base-2 logarithm of the estimated expression ratio (negative means decreased expression, positive increased) vs on the x-axis the logarithm of the average expression level across both conditions. The genes that passed the significance threshold (adjusted p-value < 0.25) are colored in red.



The gene-by-gene analysis can be obtained by viewing the file labeled "DESeq2 result file on data…"

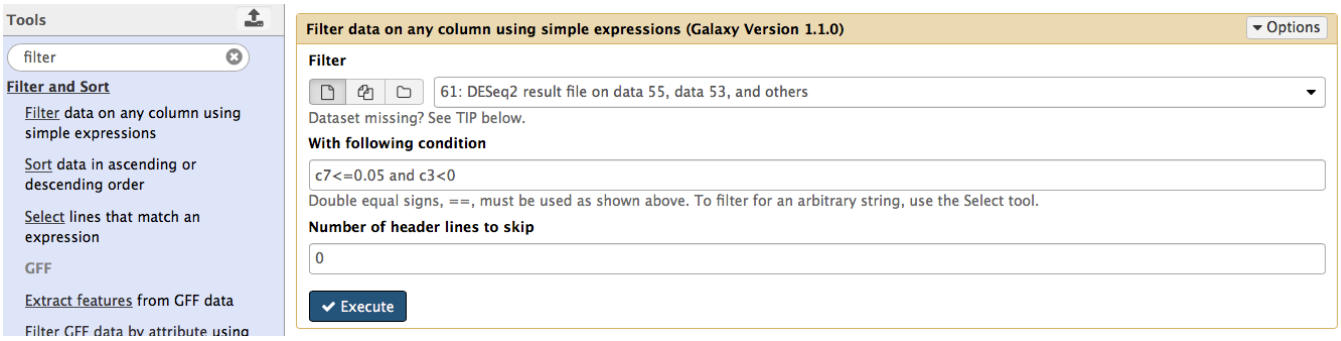
GeneID	Base mean	log2(FC)	StdErr	Wald-Stats	P-value	P-adj
G6pc2	54486.5288235294	7.84250391663964	0.176316557935404	44.4796790980506	0	0
Pde11a	492.033052077491	-4.8083406415085	0.321956217409658	-14.9347656032073	1.95761236019026e-50	1.4261206043986e-47
Spc25	2222.32838332969	3.21840943482975	0.228089358243442	14.1103007155411	3.28175579585727e-45	1.59383939818802e-42
Trib3	1597.83511250692	2.28372576843681	0.201906260151954	11.3108219959009	1.15997672443867e-29	4.22521521876784e-27
Slc4a10	1670.80112128982	2.03234430088654	0.183268049709545	11.0894632430886	1.41132040452399e-28	4.11258765878291e-26
Scn7a	793.664566648361	-2.43542024809419	0.229208456150478	-10.6253507789229	2.27156557520702e-26	5.51611840512772e-24
Chrm4	1563.36717315857	-2.19136533199029	0.211646269763385	-10.353904816939	4.0175804282422e-25	8.36230669135555e-23
Pygb	1990.52984422203	-1.66123108785707	0.172881201012222	-9.60909039346407	7.31926949974636e-22	1.33302195764131e-19
Gpr158	6666.70592033375	1.36995170152107	0.14807224655978	9.25191407133812	2.2050794861681e-20	3.56977867927436e-18
Surf4	24094.9044392546	1.19038621008798	0.130026882478416	9.15492386957426	5.4396150752038e-20	7.92551916457193e-18
Nebl	2598.33967854031	-1.43700271411235	0.169203113333942	-8.4927675726407	2.01773571447637e-17	2.6725826690837e-15
Neb	979.346777873549	-1.87645076572851	0.226890300135562	-8.27029963205732	1.33622764983989e-16	1.62240307151394e-14
Slc17a9	1219.39313031105	1.7994617590645	0.218294264129734	8.24328466090646	1.67546324706942e-16	1.87780765460011e-14
Itga6	5113.64070172806	-1.26478614033116	0.154252148091574	-8.19947181273809	2.41445524649881e-16	2.51275806724911e-14
Trp53inp2	17410.9510973002	1.12692241791494	0.138257232277905	8.15091116282265	3.61192391261861e-16	3.50838209379021e-14
Chac1	3374.35760400138	1.48406482559427	0.185689254630401	7.99219550182469	1.32556608160372e-15	1.20709361306038e-13
Pamr1	4421.23201888375	-1.33113213592402	0.16855967103689	-7.89709737646971	2.85473343725995e-15	2.44667448122809e-13
Asxl1	3646.78423833789	-1.29928904515626	0.167136338825983	-7.77382736921762	7.61495012033681e-15	6.16387906962818e-13
Upf2	5330.80130360126	-1.09216575182574	0.144499292551	-7.55827750118755	4.08441941942504e-14	3.13210478636962e-12
Dzank1	2361.81915538938	-1.24190195010703	0.165263387401902	-7.51468289275025	5.70490440115794e-14	4.15602285624356e-12
Pax6	41376.5678867265	-1.03940562149207	0.142156471380743	-7.31170105304731	2.63781461609417e-13	1.82778736251786e-11
Wfdc16	691.174730650516	-2.10673556202659	0.288371828228954	-7.30562196371671	2.7598711033214e-13	1.82778736251786e-11
Dnajc24	1426.81973456357	1.63180327721002	0.22587592574206	7.22433465119904	5.0356114080245e-13	3.1899503571703e-11
Cers6	3406.84661310868	-1.16063211892233	0.162176898865403	-7.15658103615351	8.27139936955143e-13	5.02142870059852e-11
A530058N18Rik	1979.32849523523	1.26608354197612	0.177118049394058	7.14824686872709	8.78931427818581e-13	5.12241236132669e-11

The output file is sorted with most significant (lowest p-value) at the top. The columns of this file are:

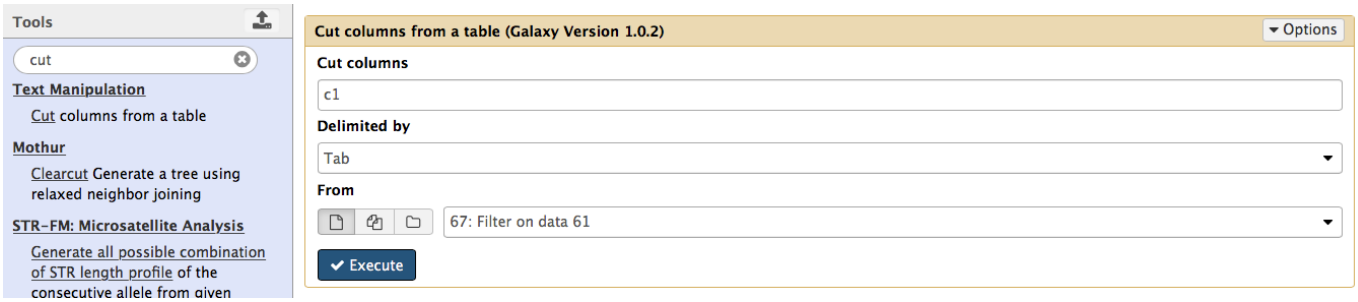
1. **GeneID**: the identifier of the gene as drawn from your GTF file
2. **Base mean**: the average expression level for this gene across all samples
3. **Log2(FC)**: the base two logarithm of the estimated ratio of expression between conditions
4. **StdErr**: the estimated standard error of the Log2(FC) value
5. **Wald-Stats**: A statistical value used in assessing the likelihood of observing this level of difference under the assumption of no differential expression.
6. **P-Value**: The probability of obtaining the Wald-Stat value in a single trial
7. **P-adj**: The adjusted probability, based on a Benjamini-Hochberg estimate of "False Discovery Rate" (FDR). In essence, this value assigns a probability of obtaining this value simply because many genes were tested rather than true divergence from equal expression in the two samples.

Step 10: Extract the differentially expressed genes

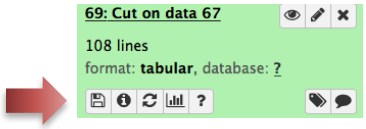
To output the differentially expressed genes, we need to do more one thing. We need to filter the table we obtained from the previous step. Note that the column 7 (c7) is the FDR rate and we are setting cut-off on it to get highly differentially expressed genes. Column 3 (c3) is the log2 of fold change. In our WT vs. KO comparison, negative values mean down-regulation in KO samples.



Then we 'Cut' to have the first column that contains gene names.



Now we have a list for down-regulated genes in KO samples. Click save to download.



Please repeat the above steps with filtering condition c7<=0.05 and c3>0 to get the list of up-regulated genes in KO samples.

Also cut and save the full list of genes in chr2 by 'Cut' from any of the 'featureCounts on data …' results.

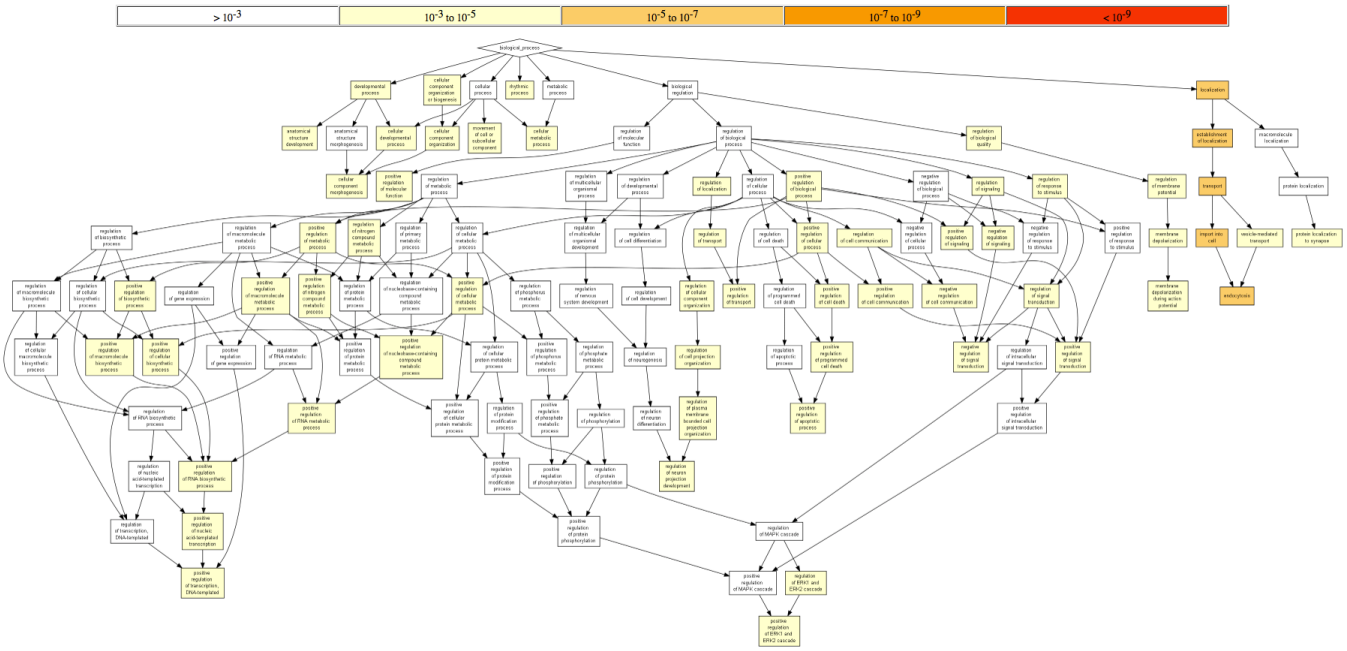
Steps 11: Gene set enrichment analysis on differentially expressed genes

Now we have three gene lists, up- and down-regulated genes in KO samples and all genes. We would like to know the functional enrichment among the differentially expressed genes.

We can input our list of differentially expressed genes to a Gene Ontology (GO) enrichment analysis tool such as GOrilla to find out the GO enriched terms.

1. Go to http://cbl-gorilla.cs.technion.ac.il
2. Choose the Mus musculus in organism.
3. Choose Two unranked lists of genes.
4. Upload up- or down-regulated gene list in the target set.
5. Upload all genes in the background set.
6. Choose 'all' for ontology quest.

You will be taken to the result page, like:



At this stage you may discover what has been changed due to the Pax6 knockout.