**­Individual homework BOHTA 2017**

Name: **William Bullock**

KU ID (what you use for Absalon): **ZHN775**

Exam number: **49**

Deadline is Thursday 2017-06-22, 10.00. **Submission must be made through digital exam.** Late submission counts as no submission. Earlier submissions are recommended!

The homework should be as a single PDF file including R code, images etc., and whatever else that is requested in the respective question. As in the earlier home works, you need to describe what you do so that it is reproducible – for analysis in R you have to supply both the answer (whatever it is – numbers, a table, plots or combinations thereof), as well as the R code you used to make the plots. Note that if the R code gives different results from your results, you will get severe point reductions or even 0 points for the exercise.

Please start your answer with the actual question, and do not “hide” your code at the end of the document – it should be together with plots and results.

For **plots**, remember to use labels on the axes, a title and explain the colors you use (a legend or a text next to the plot).

**For analysis:** Your methodology descriptions have to be detailed enough to reproduce. This applies to R code, Galaxy or Linux. So, you need to supply the code (both linux and R) that will reproduce your results as well as an explanation of why you do what you do, for example as code comments. Also remember that word sometimes reformats things when you copy/paste the text in – it is your responsibility to make sure the code can be copy/pasted and still work as is.

For **statistical tests**, you must

1) Motivate the choice of test,

2) State exactly what the null hypothesis is (depends on test)

3) Comment on the outcome – do you reject the null hypothesis (and using what threshold) or not, and what does this mean for the actual question we wanted to answer (interpretation)?

Please remember the answers that get the most points are the short and concise ones – we may deduct points for long texts with little information in it.

Copying other sources, including other students, is not allowed – clear cases of this will be reported and the best thing that can happen is severe reduction in points. Reading articles, Wikipedia etc. is very fine, as is citing of findings from there. Screen dumps from the UCSC browser are of course allowed.

Questions marked with a star (\*) are extra difficult.

Datasets are at two places: this is due to that some of the files are really large. Small datasets (used in part 3,4,5) are at <http://people.binf.ku.dk/albin/teaching/bohta2017/data.zip> , while large files (part 1, 2) are at /home/bohta/HW4/ at the ricco.popgen.dk server that you used previously . You typically want to process those on the server anyhow. Remember you log into the terminal with “ssh [username@ricco.popgen.dk](mailto:username@ricco.popgen.dk)” (no need to use the –X or –Y for the assignments). Your username is as always your KU ID (ex: abc123) and your password is bohta17.

Interpretation questions are welcome on the Absalon forum: Discussions->Exam questions (homework 4) but remember to check whether the questions have already been asked.

Good luck!

**Part 1: 10 super-transcription factor clusters to rule them all? (total 11p)**

**Question 1:** In the ENCODE project, a large number of labs have pooled resources to make many ChIP experiments for a large number of transcription factors (TFs) in different cell lines. Each such experiment will give a large number of ChIP peaks, corresponding to transcription factors binding DNA. ENCODE made a “master” bed track which shows ChIP peaks for all the transcription factors that they made ChIP experiments for. This can be found in the UCSC browser at Regulation->ENCODE regulation->Txn Factor ChIP. The below image shows an example of the track around the RXRA TSS. Each block is the ChIP peak of one TF.



As the picture show, there seem to be hotspots in the genome where many different TFs bind. At /home/bohta/HW4/part1/txnChIP.bed on the ricco.popgen.dk server there is a link to a bed file corresponding to the master track above, from the hg19 assembly.

**A:** Using BEDtools in Linux, merge the ChIP peaks that overlap over 1bp or more, and produce a BED file with these regions. How many merged regions are produced compared to how many peaks you started with? **2p**

**B:** Using R, plot the distribution of ChIP peaks in each merged region using ggplot. Use log2 scaling for the x-axis, not logged values (in the example below, the first plot is a raw distribution, the other is with log2’d values and the third is with log2 scale). Briefly comment your plot (30 words max). **2p**

 

**C:** Using R **only** (no pasting or editing allowed in external programs), produce a new BED file that contains the 10 merged regions having the highest number of ChIP peaks. **1p**

**D:** Upload this into the UCSC browser. Look at each merged region and try to interpret it, also taking the peaks it contains into account. What do the merged regions typically overlap? Is there a particular factor that is responsible for the clusters? **4p**

**E:** In hindsight, given the result in D, what could we have done to improve the analysis? **2p**

**When you are done with this question, please delete the .bed files you made or copied from your directory on the server – they take a lot of space.**

**Part 2: Selection of tools – server part (total 18p)**

**Intro:** The professor you are doing your master thesis for wants you to analyze an RNA-seq dataset. Furthermore, she is very keen on using some of these so-called pseudo-aligner tools that she has heard so much about – more specifically Salmon and Kallisto (<https://www.ncbi.nlm.nih.gov/pubmed/28263959> and <https://www.ncbi.nlm.nih.gov/pubmed/27043002> respectively).

For the final analysis you can naturally only use one tool, which means you have to select either Kallisto or Salmon. Since you are a budding data scientist, you will select the best tool based on performance on real data instead of flipping a coin. More specifically, you want to analyze 3 paired end RNA-seq libraries (biological replicates) using both tools and see which one performs better. There is no need to quality trim the sequences.

Part 2 is about doing the actual quantification with the two tools and the next question (part 3) is about doing the comparison (using the data we provide!).

Part 2 is going to be done exclusively on the Ricco server. Note that we provide you with real-sized dataset so some waiting time might occur.

To run Kallisto and Salmon you naturally need the data. In your own directory ( /home/<KU\_ID>/ ) use the “ln -s” function to make a symbolic link (shortcut) to all 6 fastq files located in the /home/bohta/HW4/part2/” folder. This is done instead of copying the file to your directory and saves a lot of space (please don't copy – we will run out of hard disk space if you do!).

Remember you can always get information about a command you don’t know using ‘man <commandName>’.

Note that the files with the “R1” and “R2” suffix correspond to each end of the paired end data (so the first sequence in each file is a pair, the second sequence in each file is a pair etc).

**Question 2.1**: Provide one line of code which will make symbolic links only to the 6 fastq files

(hint you can write “man ln” to get the manual page on the server or look here: <http://man7.org/linux/man-pages/man1/ln.1.html>) **1p**

Next, you want to check how well the sequencing run went.

**Question 2.2**: use the “wc” function to calculate the number of reads in all fastq files and comment on the results using max 50 words. **2p**

**Question 2.3**:

Now you are ready to quantify the data. To save time and computational resources, you will in this exam just quantify the WT1 library. You start with Kallisto which can be run on the server with the following command: “nice /home/bohta/bin/kallisto quant”. It is important that you include the “nice” part of the command as that will enable multiple users to use the server simultaneously! Note that you can open the Kallisto documentation by typing: “nice /home/bohta/bin/kallisto quant”

You want to:

1. Tell Kallisto what isoforms to quantify (as stored in a precompiled index) by specifying   
   “--index /home/bohta/HW4/part2/kallistoIndex”.
2. Tell Kallisto that it is a first stranded RNA-seq library by adding “--fr-stranded” to the command.
3. Tell Kallisto to use 6 cores for the quantification by adding “-t 6” to the command (Only use 6 so there are also cores for your classmates!).
4. Tell Kallisto to output a plain text document by adding “--plaintext” to the command.
5. Enable bias correction.
6. Output result to an appropriately named directory.

Assignment:  
A) Report the command for running Kallisto on the WT1 RNA-seq data. **1p**

B) Report the number of pseudo-aligned reads. **1p**

C\*) Report the estimated average fragment length. Based on this result, what is then the distance between the 3’end of an average read pair? Comment on the result using max 50 words. **4p**

**Question 2.4**:  
Next, use Salmon for the quantification which can be run on the server with the following command: “nice /home/bohta/bin/salmon quant”. Note that you are quantifying reads (not alignments) and only need to consider “basic options” and that you can open the Salmon documentation by typing: “nice /home/bohta/bin/salmon quant -h”

You want to:

1. Quantify the same isoforms as you did with Kallisto by adding “--index /home/bohta/HW4/part2/salmonIndex” to the command.
2. Tell Salmon to use 6 cores for the quantification by adding “-p 6” (Only use 6 so there are also cores for your classmates!).
3. Tell Salmon to automatically detect library type by adding “--libType A”.
4. Turn on the bias correction algorithms.
5. Output the result to an appropriate directory.

Assignment:  
A\*) Report the command for running Salmon on the WT1 RNA-seq data. **2p**

B) Report the most likely library type as identify by Salmon. **1p**

C) Report mapping rate. **1p**

**Question 2.5**:

Which tool aligned more reads? Comment on the result using max 50 words. **1p**

**Question 2.6\***: The effective length estimate (estimated by both tools) incorporates the bias corrections performed by each tool respectively into the length of the isoform. This means you can calculate RPKM/FPKM values less biased by sequence specific effects (such as length and GC content). The file with the quantification is called ‘abundance.tsv’ and ‘quant.sf’ for Kallisto and Salmon respectively. (hint: you might need to look at the actual files previously used in this assignment).

1. Compare the estimated “effective length” of the isoform ‘TCONS\_00000020’ from Kallisto and Salmon to each other and the reference length. This must be done either in R or with the command line tool “grep”. **1p**
2. \*What could explain the difference in the effective length? Which estimate do you trust more? Answer using max 75 words. **3p**

**Part 3: Selection of tools – analysis part (total 19p)**

In the attached data in the part 3 folder in [http://people.binf.ku.dk/albin/teaching/bohta2017/data.zip](http://people.binf.ku.dk/albin/teaching/bohta2017/dat.zip) you will find results from Kallisto and Salmon runs for all three biological replicas in the file “part3.Rdata”. This file (created with the save() function) can be loaded into R with the load() function and contains 3 R objects:

* A replicate count data.frame (“countDF”). This data.frame will be referred to as the “**count data**”.
* An annotation matrix (“annotationDF”). This data.frame will be referred to as “**annotation**”.
* A filtered RPKM replicate expression matrix that has been log10 transformed with a pseudo count of 1 (“logRpkmDF”). This data.frame with be referred to as **logRpkm** data/values.

Samples from Kallisto have the prefix “K\_” and ones from Salmon the prefix “S\_”.

Use these files for the rest of the part 3! Note that countDF and logRpkmDF data.frames contain different isoforms and should not be compared.

**Question 3.1**: Load the data into R. How many isoforms are quantified in the count data? **1p**

**Question 3.2**

You naturally want to normalize the data – more specifically you want to calculate RPKM values from the count data using the length from the annotation data. Use vectorised analysis to calculate RPKM values for all samples. A vectorised approach means using R code that calculates the operation for the whole vector (aka row or columns) simultaneously instead of one at the time (this means without using loops, the apply family of functions (apply, sapply etc.) or similar functions).

Assignment:

- Report the R code for how to calculate RPKM values. **3p**

**Question 3.3**: Make a one-liner (without the use of “;”) that outputs the mean RPKM value of each sample. The restrictions from the previous question no longer apply. **2p**

**For the rest of this question, use the logRpkm data (not the RPKM values you just calculated in 3.3; the logRpkm file cannot be used as a check for the correct answer for 3.3, it is different)**.

Now that you have the normalized data: time to make some summary plots

**Question 3.4**:

Make histograms of the distribution of the logRpkmexpression values. For this plot use colour to indicate the tool and use ggplot2 facets to make subplots that show each replicate (meaning you have 3 subplots in a larger plot). Provide all the R code necessary to produce the plots and comment on the distributions with max 50 words. **3p**

**Question 3.5**:

For each tool use logRpkm to calculate all pairwise replicate Pearson correlations of the replicate expression values and report the numbers in a table (one table per tool). **2p**

**Question 3.6**: One of the best ways of evaluating replicate agreement is to calculate the cross replicate variance (often abbreviated CV) so of course you want to do that.

A CV value can be calculated for isoform *i* as follows: cvi = variance(xi) / average(xi), where xi is a vector with the replicate expression values for isoform *i*.

1. Construct a function which calculates the CV and use the apply() function to calculate the CV based on logRpkm values for Kallisto and Salmon (separately). **1p**
2. Plot the distribution of CV values as density lines (in one single plot) using color to indicate the tool and log transform the x-axis (using log10). **1p**
3. Comment on the CV plots using max 75 words. **2p**

**Question 3.7\***:

Based on all the results you have collected here (all of part 2 and all of part 3), discuss in max 100 words which tool would you choose to continue with if you wanted to make a differential expression analysis. **4p**

# 

# Part 4: Exploratory Data Analysis (15p total)

Two postdocs in your lab, Bob and Alice, have been collaborating on an experiment. Using your labs favorite cell line, they have generated 25 knockdowns of two transcription factors along with 25 control samples. For each knock down sample, they have measured the knock down efficiency using qPCR and quantified expression of 10000 genes using RNA-Seq.

Last week the two postdocs got into an argument: Bob is accusing Alice of being sloppy in the lab and thereby ruining the experiment. Upon hearing these allegations, Alice responded by accusing Bob of sneezing into one of the samples.

Your professor, who knows you have been attending a bioinformatics course, has charged you with analyzing the data to solve the dispute between Bob and Alice and determine whether the dataset is ruined.

You have been supplied with an expression matrix containing normalized and log-transformed expression values, as well as the study design describing the content of each sample.

**Question 0:** Before you start, set your seed by set.seed to 2017 and load all packages you need for your analysis (0p – but you will get a minus point if you miss this).

**Question 1:** Read both the expression matrix ("ExpressionMatrix.tab") and study design ("StudyDesign.tab") into R. These are both found in <http://people.binf.ku.dk/albin/teaching/bohta2017/data.zip>, part 4. Report the number of samples and the number of genes. **1p**

**Question 2:** Perform PCA on the samples and report the amount of variance contained in the first 5 principle components. Note: You should scale the expression values before the PCA. **2p**

**Question 3:** Inspect the first 2 principle components**:** Make a plot of PC1 vs PC2, where knock down efficiency is indicated by color and knock down target is indicated by shape. Comment on the plot using a maximum of 75 words. **2p**

**Question 4:** Perform a K-means clustering of the data, using k=3 and 10 random starting points. Visualize the clustering by making a plot of PC1 vs PC2, where the clusters are indicated by color. Briefly comment on how the clustering corresponds to the known knockdown targets, using a maximum of 75 words. **3p**

**Question 5\*:** Using a maximum number of 3 plots, investigate whether there is any truth to the postdoc allegations: Can you see a difference in samples prepared by Alice and Bob? Is there any indication Bob has ruined a sample? Discuss you results using a maximum of 75 words. **4p**

**Question 6:** Based on all you observations in Question 1-5, discuss whether the experiment is still useful, or whether the postdocs have ruined it. Use a maximum of 100 words. **3p**

# Part 5: Differential Expression (DE) (total 18p)

Another PhD student in your lab, named Caroline, has already performed a DE analysis on a subset of the above dataset using the popular R-package limma (limma performs tests for DE using a modified version of gene-wise linear regressions). She has asked for you help in further analyzing the data. She is particularly interested in whether the two knockdowns affect the same genes.

The DE analysis file contains mean expression across all samples, and for each transcription factor knockdown (geneA and geneB) log fold change values (logFCs), p-values and p-values corrected for multiple testing using Benjamini-Hochberg.

**Question 1:** Read the DE analysis ("DifferentialExpression.tab") results into R, and show the first few lines of the file. This can be found in <http://people.binf.ku.dk/albin/teaching/bohta2017/data.zip>, part 5. **1p**

**Question 2:** As part of a single plot, produce two MA-plots (one for each knockdown). To mitigate overplotting, points on the plot should be transparent. **2p**

**Question 3:** Explain how an MA-plot can be used to investigate whether expression data has been properly normalized (max 75 words). **2p**

**Question 4:** Report the number of upregulated and downregulated genes that are significantly DE after correction for multiple testing (at alpha=0.05). Which knockdown target has the highest total number of DE genes? **2p**

**Question 5:** Plot the two sets of logFCs (log2 fold changes) against each other. Color points based on whether they are significantly DE after multiple testing correction (at alpha=0.05) in either one or both of the knockdowns. **3p**

**Question 6:** Calculate the Pearson correlation between the logFCs of the two knockdowns, and test whether this correlation is significant. **2p**

**Question 7:** Caroline asks you to construct a 2-by-2 contingency table showing whether genes are significantly DE (after correcting for multiple testing, alpha=0.05) in the two knockdowns. Perform a Fisher’s Exact test for any association between the DE genes in the two knockdowns. **3p**

**Question 8:** Based on Question 4-7, do you think the geneA and geneB transcription factors regulate the same genes? Use a maximum of 100 words. **3p**