Institut für Informatik

Praktische Informatik und Bioinformatik

Prof. Dr. Ralf Zimmer

Armin Hadziahmetovic

Wintersemester 2021/2022

# GoBI Excercise 4

## Differential Analysis (200P)

**Deadline**: Monday, 24.01.2022, 12:00

In Exercise 4 we practise some basic methods and visualizations for differential analysis. You will perform most of the tasks in the high-level statistical language R and learn/refresh programming in R. Therefore, we practise basic analysis tools in R and apply them to a current COVID-19 data set (eg. DESeq). Next, we compute exon splicing information (PSI values) from NGS data (BAM-files). From such data we derive basic splicing models by parameter estimation via maximum likelihood (ML) and compare different models by a likelihood ratio statistic test (LRS). The results will be compared with results from a standard analysis tool for differential alternative splicing (DEXSeq).

### Task 1: Differential gene analysis (40 pts):

The task here is to learn about basic differential analysis method using R. As a tutorial introduction work we use chapter 8 of the book *Holmes&Huber, Modern statistics for modern Biology, 2019* (https://www.huber.embl.de/msmb/Chap-CountData.html, Chapter 8). Susan Holmes and Wolfgang Huber introduce basic analysis methods and visualization tools with R and Bioconductor. Thus, the book can help you to do a standard differential analysis of your own data and in Chapter 8 learn about state-of-the-art tools (e.g. DESeq) for differential gene expression based on count data. In order to learn to analyse your own data, apply some of the methods to a recent COVID-19 data set provided in the directory

`/mnt/biosoft/praktikum/genprakt/DiffAnalysis/Corona`

Here you find gene count data of samples with and without SARS-CoV-2 infection. Your task is, of course, to analyse the differential gene expression induced by and relevant for the virus infection.

Apply differential analysis tools, e.g. from chapter 8 of the book, to your data. Summarize and visualize your findings in a report (in the form of a scientific paper (data, methods, results, refs)). Focus on "new" findings/hypotheses on SARS-CoV2 infections.

**Task 2: Compute Percent-Spliced-In (PSI) values (60 pts):**

Extend your feature counting program from GoBI Assignment 3 and create an executable jar-file which computes Inclusion (IRC) and Exclusion Read Counts (ERC) and the associated PSI values from given BAM-files.

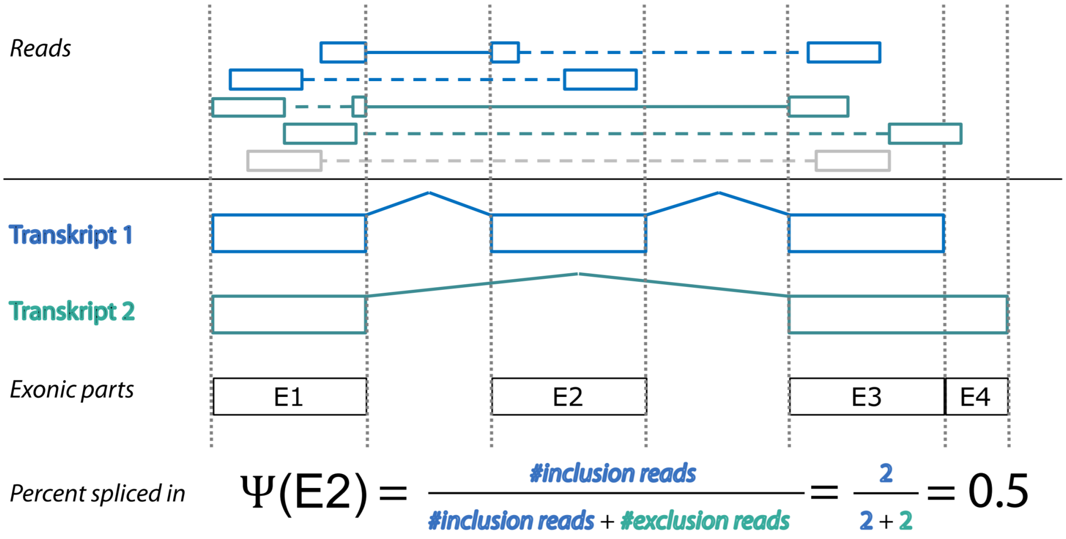
We consider only exon skipping events as defined in the GTF-file. For each "skipped exon" *se*, count reads that provide evidence for transcripts that include *se*. These are called "inclusion reads" (shown in blue in the figure below). For any skipped exon *se*, split alignments that are mapped to positions upstream and downstream of *se* and cannot be mapped to a transcript including *se* are called "exclusion reads" (shown in green in the figure below). Exclusion reads provide evidence for transcripts that do not contain *se*. "Ambiguous reads", which can be mapped to both transcripts contain no evidence about the inclusion/exclusion of *se* and are not counted. Compute the Percent-Spliced-In (PSI) values from Inclusion (IRC) and Exclusion Read Counts (ERC).

Implement a program with the following parameters:

- `-gtf <GTF file>` : genomic annotation
- `-bam <BAM file>` : BAM-file containing the reads to be counted
- `-o <output file path>`: path to the output where the table (defined below) containing all attributes will be written

Each row should correspond to an exon from the annotation with total read count greater than zero. The output should be written into a tab separated text (TSV) file with the following headers/columns:

- gene (gene id)
- exon (coordinates of the exon, e.g. 41048550-41048637)
- num_incl_reads (inclusion read count)
- num_excl_reads (exclusion read count)
- num_total_reads (total read count)
- psi (the calculated PSI value)


Reads
Transkript 1
Transkript 2
Exonic parts
E1
E2
E3
E4
Percent spliced in
Ψ(E2) = #inclusion reads / (#inclusion reads + #exclusion reads) = 2 / (2 + 2) = 0.5

**Guidelines and Deliverables:**

The provided GTF-file contains only genes for which exactly 2 isoforms/transcripts are annotated, which identify the skipped exons for any gene.

The output files should be written to your output directory `[...]/$user/psi/` and have the same basename as the BAM-files and the extension `.psi` (e.g.: `sample1.bam -> psi/sample1.psi`).

Save your solution to `/mnt/biocluster/praktikum/genprakt/${user}/Solution4`. Provide also an executable jar-file (including sources!) in this directory that allows reproducing your results. The jar should print a usage info if invoked without parameters.
Submit your jar-file also to the `<abgabeserver>` (template `psi`) at:

`https://services.bio.ifi.lmu.de:1047/abgabeserver/`

You find all referred input and example files in the directory:

`/mnt/biosoft/praktikum/genprakt/AlternativeSplicing`

The bam files are available here: `[...]/AlternativeSplicing/bam/`

For all of the following exercises use this transcript annotation:

`[...]/AlternativeSplicing/annotation_b37.gtf`

The ground truth for the exons in the annotation file can be found in:

`[...]/AlternativeSplicing/differential_exons.RData`

Apply your tool to all provided BAM-files in the directory:

`[...]/AlternativeSplicing/bam/`.

**Task 3: Simple differential test of PSI-values (60 pts):**

We assume that exon inclusion is a simple random process, where for each transcript there is a probability $p$ for the inclusion of exons. We further assume that for each transcript there is at least one read informative for the exon (either covering the exon or skipping the exon). If we consider now a total of $N$ reads informative for an exon, then $p$ corresponds to the percent spliced in (PSI) ratio. The number of inclusion reads ($i$ = IRC) out of $N$ = IRC+ERC total reads follow a Binomial distribution with parameters $p$ (inclusion probability) and $N$ (total reads):

$$P(i, N \mid p) = \binom{N}{i} p^i \, (1-p)^{N-i}$$

We now consider two groups with 5 samples each and we assume that the samples are independent realizations of the random process described above. Two probability models (the reduced model and the full model) will be compared to evaluate whether or not the PSI values of an exon differ between two groups of samples. The reduced model assumes one shared parameter $p_0$ for all samples, while the full model has two separate parameters $p_1$ and $p_2$ for each group of samples. You can think of the data and the models as being organized in the following table:

| Sample $i$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Group $g_i$ | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 |
| IRC | $i_1$ | $i_2$ | $i_3$ | $i_4$ | $i_5$ | $i_6$ | $i_7$ | $i_8$ | $i_9$ | $i_{10}$ |
| Total | $N_1$ | $N_2$ | $N_3$ | $N_4$ | $N_5$ | $N_6$ | $N_7$ | $N_8$ | $N_9$ | $N_{10}$ |
| Reduced model $\Psi \sim p_0$ | $p_0$ | $p_0$ | $p_0$ | $p_0$ | $p_0$ | $p_0$ | $p_0$ | $p_0$ | $p_0$ | $p_0$ |
| Full model $\Psi \sim (p_1, p_2)$ | $p_1$ | $p_1$ | $p_1$ | $p_1$ | $p_1$ | $p_2$ | $p_2$ | $p_2$ | $p_2$ | $p_2$ |

To find the optimal model parameters the Maximum Likelihood Estimation (MLE) will be used. With the assumption of independence for the samples we can write the likelihood function for the two models as the probability of all the sample data in the respective (reduced or full) model as:

$$L_{reduced}(p_0) = \prod_{j=1}^{10} P(i_j, N_j | p_0)$$

$$L_{full}(p_1, p_2) = \prod_{j=1}^{10} P(i_j, N_j | p_{g_j})$$

To make computations easier you can use the log-likelihood $log(L)$ instead of $L$ to solve for values $p_0$ or $(p_1, p_2)$ maximising the likelihood, respectively, as $log$ is monotonic function, To solve for the maximising $p$, compute the gradient (partial derivative) of the (log-)likelihood with respect to the model parameters ($p_0$ or $(p_1, p_2)$). To find the maxima, set the gradients equal to zero and solve for the parameters. Document and explain your solution (expressions for the Maximum Likelihood Estimates (MLE) for $p_0$ or $(p_1, p_2)$) in your report.

Compute the likelihoods $L_{reduced}$ and $L_{full}$ of the data according to the two models using the maximum likelihood estimates as parameter values and compute the likelihood ratio statistic (LRS):

$$LRS = -2\ log(\frac{L_{reduced}}{L_{full}})$$

The LRS values follow a Chi-square distribution with *df* degrees of freedom

$$df = \#\text{params.full.model} - \#\text{params.reduced.model}$$

distribution to compute the p-value of the observed LRS x for each exon (`P(LRS > x)`). Implement the model as a function in R. Create an R file called `LRS.R`. This file should only contain the following function definitions and not execute any commands. The function should have the following interface:

```
LRS <- function(incl, total, group)
```

*incl, total* and *group* are vectors of the same length (= number of samples, here 10). *incl* contains inclusion counts (IRC) of an exon for each sample and *total* contains the total counts (IRC + ERC) of one exon for each sample. *group* can take values 1 or 2 indicating which group (i.e. condition) a sample belongs to.

You can use the data provided in
`[...]/AlternativeSplicing/diff_psi_test.RData`
to test your method.

It contains two matrices: *inclusion* and *total* with read counts for 20 exons in 10 samples. In addition, it contains the grouping of samples in a vector called *group*.

Extend your script to be able to use the output of Task 2 above for differential alternative splicing analysis based on the likelihood ratio statistic test (LRS). Add another function to `LRS.R`:

```
diff.splicing <- function(psi.files, group)
```

The function should use the `LRS()` method you implemented and accept the following input:

- psi.files - character vector containing the filenames that specify the read count data for each sample obtained in task 2).
- group - integer vector as in the LRS function corresponding to the psi.files

The output should be a matrix or data frame with the following columns (including the header):

- gene (gene id)
- exon (coordinates of the exon, e.g. 41048550-41048637)
- p0 ($p_0$, estimated parameter for the reduced model)
- p1 ($p_1$, estimated parameter for the full model
- p2 ($p_2$, estimated parameter for the full model)
- llreduced ($L_{reduced}$, log-likelihood of the reduced model)
- llfull ($L_{full}$, log-likelihood of the full model)

- lrs (the calculated likelihood ratio test (LRS) statistic)
- pvalue (the p-value gotten from the ChiSquared for "lrs")
- padj (p-value adjusted with the Benjamini-Hochberg method, e.g. using the R function p.adjust)

Apply your program to the psi files generated in task 2). Samples are grouped in group 1: sample[1:5] and group 2: sample[6:10].

You will need those results for the next exercise. Upload your final `LRS.R` script to the `<abgabeserver>` (template `psi-test`).

**Task 4: Comparison with other alternative splicing tools (DEXSeq) (40 pts):**

Now we want to compare the results you computed on the simulated BAM-files with the ones we can calculate using DEXSeq.
The same BAM-files that you have used in the previous tasks were preprocessed for DEXSeq:
`[...]/AlternativeSplicing/dexseq/`
Now use DEXSeq to perform differential analysis of these read counts. To install DEXSeq for your R environment, you can use the following commands:

```
source("https://bioconductor.org/biocLite.R")
biocLite("DEXSeq")
```

To apply DEXSeq on your data, you can adapt and execute (in your working directory) the R commands given in:
`[...]/AlternativeSplicing/dexseq_script.R`

After you generated the results, you have to match the "binned exons" created by DEXSeq with the exons you have available in your annotation file (GTF-file). You can do this using the *GenomicRanges* package introduced in the hands-on part.

Compare the predicted differentially alternatively spliced (DAS) exons using your model as well as the DEXSeq model to the simulated ground truth (given above). Compute the sensitivity and the false discovery rate(FDR) for predictions with adjusted p-value $< 5\%$. You can use for example the ROCR R library to create a ROC plot using the p-value for the prediction or the method you developed yourself in the previous steps. Note that small p-values predict positive labels. Plot both ROC plots into the same plot using different colors and add a legend indicating which color corresponds to which model. Incorporate this plot and a description of the evaluation into your report.

### Submission and Deliverables

Please also send Report + Supplement **before the deadline** to `gobi@bio.ifi.lmu.de` with the subject `Topic1 Report - Group X`, where `X` is your group id.

To exercise each member can/should make individual submissions to the `<abgabeserver>`, but submit only one Report, Supplement and program per group. In this case, you should divide the DE (SARS-CoV-2 data eval) and DAS (LRS/DEXSeq eval) into two separate reports.

As a supplement to your report you should briefly describe your approach for solving the exon skipping and ML/LRS problem. This includes some analysis of the correctness, time complexity, and actual time usage of your method.

The weighting of the various deliverables is:

- up to 40 pts - Differential Gene Expression on SARS-CoV-2 data
- up to 60 pts - `<abgabeserver>` submission for `psi`
- up to 60 pts - `<abgabeserver>` submission for `psi-test`
- up to 40 pts - LRS evaluation and comparison with DEXSeq

* **Guideline:** a bunch of plots without a clear explanation of what is shown, without description of what can be observed, or without a statement what can be concluded is not a sufficient report and will be graded accordingly.