Isoform Switch Analysis Exercise

Part 1 : An introduction

Make sure you have installed IsoformSwitchAnalyzeR

- library(IsoformSwitchAnalyzeR)
- If not you need to follow the installation instructions on Absalon

Overal Exercise Idea

- Aim:
  Get familiar with the overall workflow of IsoformSwitchAnalyzeR
- Approach:
  Use the vignette of IsoformSwitchAnalyzeR and answer questions along the way.

Assignment

Individually read/work through the following sections of the vignette:

- “Abstract”
- “Workflow Overview”
- “Short Example Workflow” - in this section you should also run the R code and check in/ output

And answer the questions in the rests of the slides. Remember to look up the documentation of the functions (run ?functionName) as all the details are there.

Open vignette

Load library into R:
library(IsoformSwitchAnalyzeR)

Open vignette

vignette("IsoformSwitchAnalyzeR")

Online alternative here

Questions

- Q1: What is the core functionalities of IsoformSwitchAnalyzeR?
- A1: Identification and visualisation of alternative splicing and isoform switches with predicted consequences both for individual genes and genome wide
- Q2: What external sequence analysis tools are currently supported and what do they do?
- A2: Pfam (predict protein domains), SignalP (predict signal peptides), CPAT (predict coding potential)

Questions

- Q3: Why is it smart to identify isoform switches as the first step in a workflow? (hint look at the documentation of isoformSwitchAnalysisPart1())
- A3: Then we do not need to analyse genes/ isoforms without switches resulting in faster runtimes.
- Q4: How many high-level functions (functions that automatically performs multiple step of the pipeline) are there in IsoformSwitchAnalyzeR?
- A4: There are 3: isoformSwitchAnalysisPart1, isoformSwitchAnalysisPart2, isoformSwitchAnalysisCombined

Questions

- Q5: What is a BSgenome object and why is it needed
- A5: It is a container for genetic sequences (here human) and is needed to predict ORFs and amino acids

- Q6: What is the result if you change the dIFcutoff parameter to 0.5 in the isoformSwitchAnalysisPart1()

- A6:
  0.4: 12 Isoforms, 7 genes
  0.5: 6 Isoforms, 4 genes

Questions

- Q7: What is the main functionality of isoformSwitchAnalysisPart1() vs isoformSwitchAnalysisPart2()
- A7
  part1: identify switches, annotate ORF and output sequences.
  Part 2: integrate external analysis, predict consequences and plot top X isoform switches.
- Q8: How many switches with consequences did you identify (when using a dIF cutoff as 0.5):
- A8: 2 genes have a switch with consequences

Questions

- Q9: What is the consequence of the switch in the “LDLRAD2” gene? (identify the predicted functional consequence)
- A9: hESC -> Fibroblast: Coding -> non-coding
- Q10: Why is a cutoff on both the q-value (alpha) and the dIF necessary?
- A10: Because q-values measures the certainty of the difference while the dIF value measure the effect size. A cutoff on both is nessesary to avoid significant effects with very small effect sizes.

Questions

- Q11: How many genes have an isoform switch in both conditions of the “exampleSwitchListAnalyzed” data?
- A: From extractSwitchOverlap():
  57
- Q12: Use the extractConsequenceEnrichment to figure out which enrichment/depletion is the most certain (smallest q-value)
- By setting returnResult = TRUE in extractConsequenceEnrichment():
  COAD: ORF is shorter. Q-val 9.65e-38

Questions

- Q13*: Using a alpha of 0.05 answer: For many genes with isoform switch are the gene also differentially expressed and what does that indicate?
- A13:
  465 of 587 (89%)
  (if you counted the result stored multiple times due to multiple isoforms per gene: 1309 of 1608 (81%))
  It highlights that genes that are differentially expressed can also have isoform switches!

Isoform Switch Analysis Exercise

Part 2: Improved understanding

Overal Exercise Idea

- Aim:
  To dig into the details of an IsoformSwitchAnalyzeR workflow
- Approach:
  Use the IsoformSwitchAnalyzeR vignette and answer questions along the way.

Assignment

Thoroughly read/work (meaning run the R code and check output) through the following sections of the vignette:

- “Detailed Workflow”

And answer the questions in the rests of the slides.

Remember to look up the function documentation for all functions you use as all the details are there.

Questions

- Q1: How many functions are isoformSwitchAnalysisPart1() and isoformSwitchAnalysisPart2() internally using?
- A1: Par1: 6. Part2: Up to 7 (both also outputs the summary).
- Q2: What class is the switchAnalyzeRlist object?
- A2: It is a named list meaning each entry can be accessed via its name (using obj$name or obj[['NAME']]).
- Q3: What is the name of the main entry in the switchAnalyzeRlist object, how is it accessed and what does each row correspond to?
- A3: ‘isoformFeatures’, “switchAnalyzeRlist$isoformFeatures”, each row have information about a single isoform in a specific comparison of conditions.

Questions

- Q4: What does all the analyze* and extract* functions do?
- A4: respectively:
  - analyzing and annotating the switchAnalyzeRlist.
  - extracting (summarized) data from the switchAnalyzeRlist
- Q5: Which functions can be ude to importing RNA-seq quantification data and creating a switchAnalyzeRlist?
- A5: importCufflinksFiles(), importIsoformExpression(), importRdata(), importGTF(), createSwitchAnalyzeRlist()

Questions

- Q6: Why is the summary statistics of these two commands not different:
  e1 <- preFilter(exampleSwitchListAnalyzed, keepIsoformInAllConditions = TRUE)
  e2 <- preFilter(exampleSwitchListAnalyzed, keepIsoformInAllConditions = FALSE)

  When the resulting switchAnalyzeRlists are different:
  > nrow(e1) == nrow(e2)
  [1] FALSE

- A6: Because the parameter does not change the total number of isoforms kept - just whether the isoform is kept only in the comparison where it passes the filter or in all comparisons.

- Q7: How many different ways of testing for isoform switches are supported by IsoformSwitchAnalyzeR and which is the (current) recommended?

- A7: Three. Cufflinks (automatically imported), isoformSwitchTest() and isoformSwitchTestDRIMSeq() which is the recommended.

- Q8*: Which algorithms for identifying ORFs are supported? Which is the default and what do you think of that?

- A8: ‘longest’ (default), ‘mostUpstream’, ‘longestAnnotated’, ‘mostUpstreamAnnoated’. Currently the ‘longest’ method is the better choice as it gives very high accuracies when compared to annotation.

Questions

- Q9: Which consequences are affected by the ‘ntCutoff’ and the ‘AaJCsimCutoff’ partner in the analysis of switch consequences, and what does the cutoffs do.

- A9:
  ntCutoff: nucleotide cutoff for the difference in ’ORF_length’, '5_utr_length', '3_utr_length', 'isoform_seq_similarity', '5_utr_seq_similarity' and '3_utr_seq_similarity'
  AaJCsimCutoff: The cutoff for the Jacard distance in AA calculated when analysing ‘ORF_seq_similarity’

- Q10: How many different types of consequences can be predicted from the domain annotation and why is there a need for all of them?

- A10: 3. ‘domains_identified’, ‘domain_length’, ‘genomic_domain_position’. ‘domains_identified’ compares which domains are identified. ‘genomic_domain_position’ compares the position of the domain (domain movement). ‘domain_length’ identified differences in lengths (domain truncation).

- Q11: How do you access the details of the consequence analysis and which is the first two isoforms compared in the example (use exampleSwitchListAnalyzed)?

- A11: ‘exampleSwitchListAnalyzed$switchConsequence’. uc001kqe.3 and uc001kqd.3.

Questions

- Q12*: Using exampleSwitchListAnalyzed: What is the overlap between the top 10 switches (with consequences) and what does that indicate.
- A12: No overlap which indicate effect size and statistical certainty do not measure the same thing.
- Q13: What is the consequence of the switch in the gene which is nr 3 on the top list when sorting by Values values?
- A13: Intron retention gain.
- Q14: What is the relation between isoform switches and gene log2 FC?
- A14: Nothing - as seen from the “short-example-workflow” isoform switches are found in both genes where the expression is changing and non-changing.

Questions

- Q15*: Visually inspect the relation between isoform switches and gene expression
- A15: Modifying the code from “short-example-workflow” where log2FC is plottet vs dIF you se there is no relation. Switches are found in both highly and lowly expressed genes.