Alternative-splicing detection by NGS

Wen-Dar Lin
Bioinformatics core, IPMB
wdlin@gate.sinica.edu.tw

Preface

- In addition to gene expressions, alternative splicing isoforms provide diversity of RNAs and protein products.
- In this presentation, we will go through theories of two programs for alternative splicing analyses,
- PowerPoints and links to walk-through logs
  - https://maccu.project.sinica.edu.tw/20250930/

Aims

- Know theories of described algorithms
- Know the way to *reproduce* the walkthroughs
  - Reproduce => mimic => create!

Disclaimer

- This presentation was made based on my work experiences
  - mainly for plants.
- This presentation is *not* intended to cover related biology knowledge.
- In this presentation, the words “transcript” and “isoform” have the same meaning.
  - In some context, isoforms mean protein variants

Topics

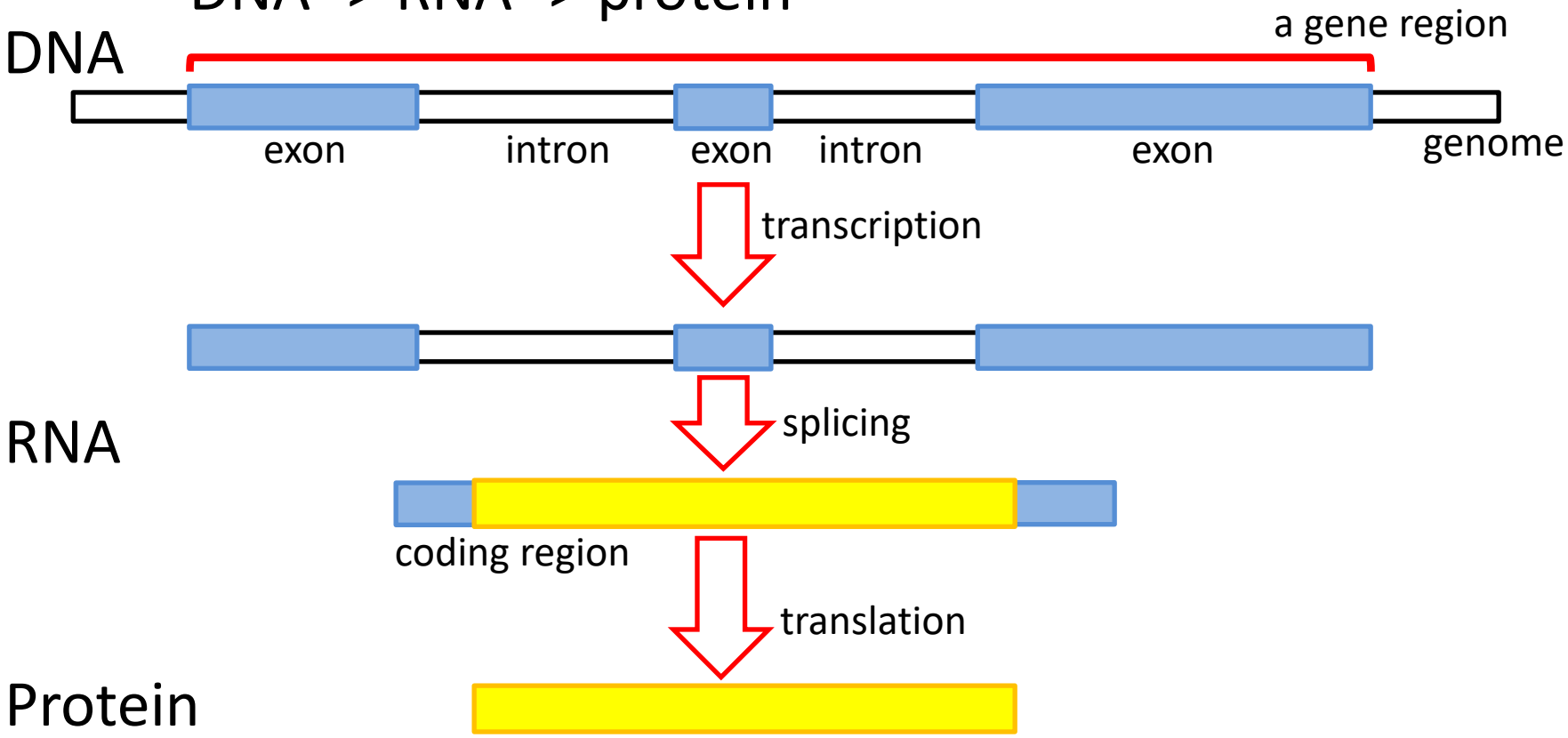
1. Detecting alternative splicing (AS)
2. Theories of isoform-based algorithms
3. Applications with isoform expression levels
4. Walk-through of the isoform-based algorithms
5. Theories of event-based algorithms
6. Walk-through of the event-based algorithms
7. Discussions

Appendixes

    1. dealing with long reads
    2. dealing with short reads and long reads at the same time

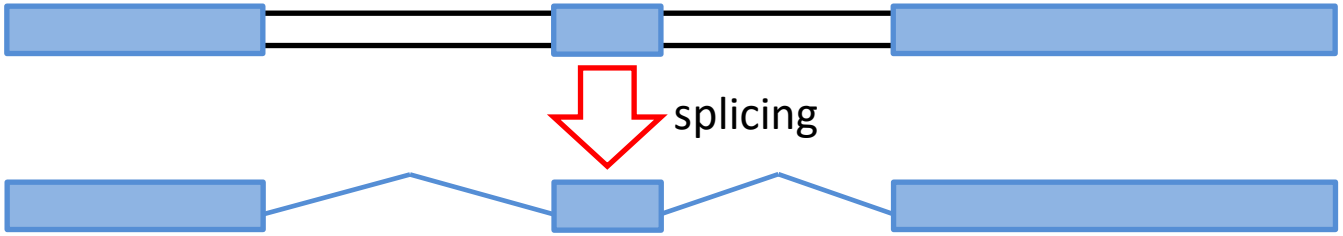
Detecting alternative splicing

- The central dogma
  - DNA -> RNA -> protein

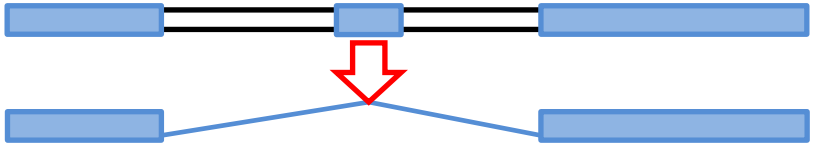


Detecting alternative splicing

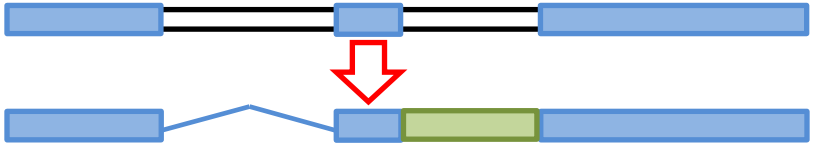
- Splicing events
  - Types of splicing junction variation



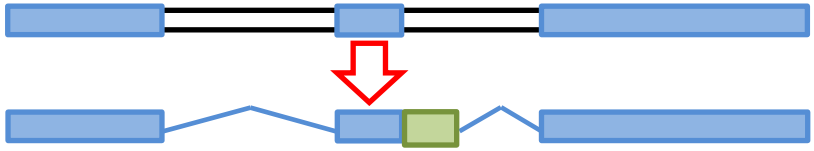
exon skipping



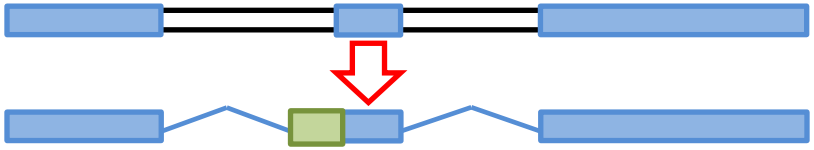
intron retention



alternative donor



alternative acceptor



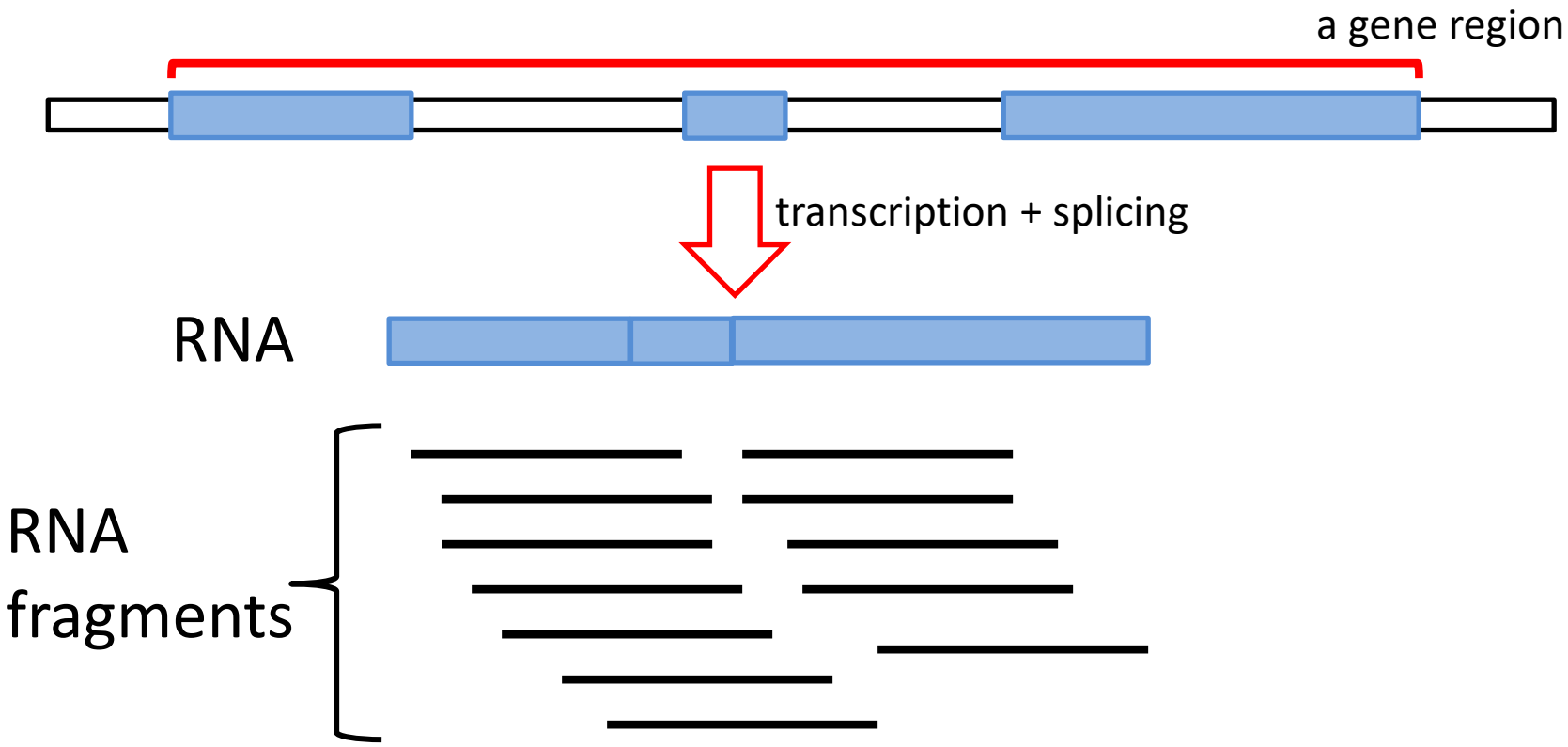
Various combinations of splicing events => various isoforms

Detecting alternative splicing

- Currently, algorithms said to be detecting alternative splicing can be *roughly* classified into two categories
  - Isoform-based
    - Predict expressed isoforms (*combinations of splicing events*)
    - Predict expression levels of isoforms => differentially expressed isoforms AND differentially preferred isoforms
  - Event-based
    - Collect read counts related to *splicing events* and do corresponding computation

Detecting alternative splicing

- RNAseq
  - Sequencing of RNA fragments

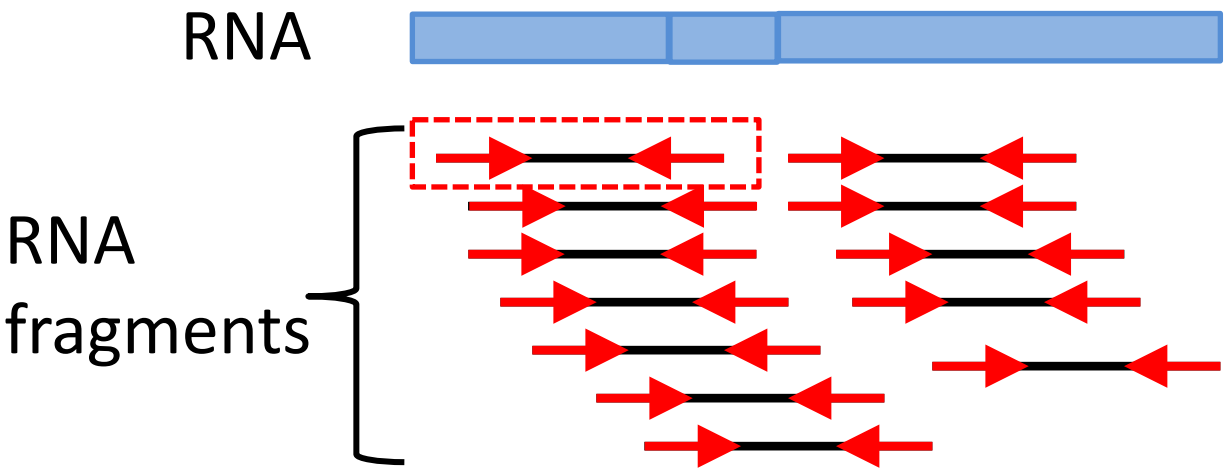


Detecting alternative splicing

- Illumina YouTube video
  - https://youtu.be/fCd6B5HRaZ8
  - Keywords
    - fragment
    - lane / tile
    - amplification / cluster
    - read 1 / read 2
    - fluorescently tagged nucleotides

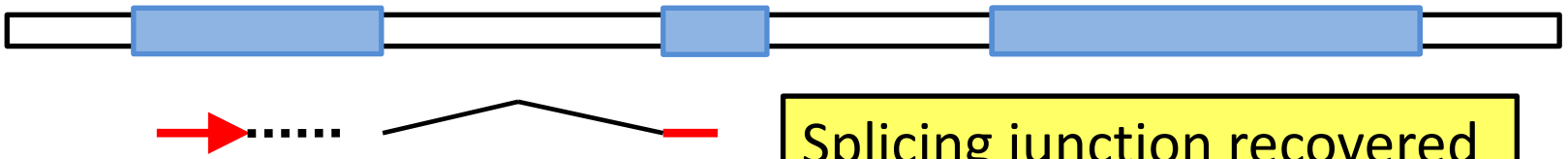
Detecting alternative splicing

- Read pairs in RNAseq data



Every two arrows of one fragment would be read1 and read2 in RNAseq data

When we mapping reads back to the genome



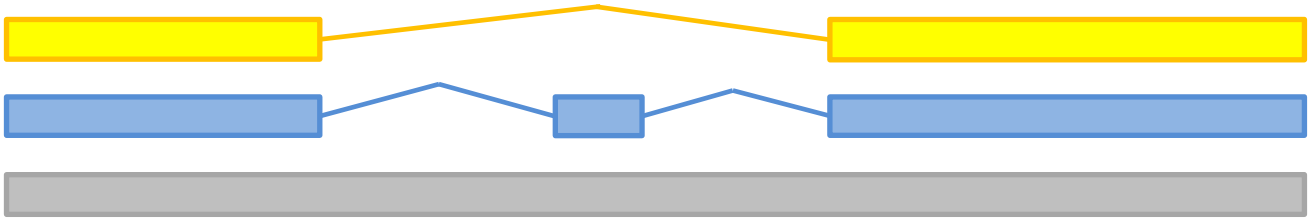
Splicing junction recovered if reads are *spanning* it

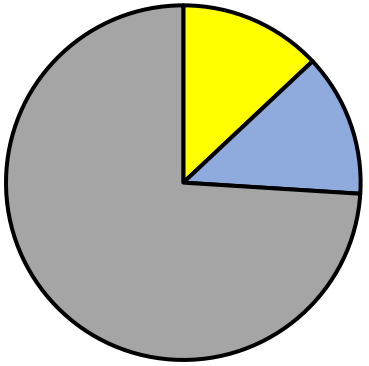
Detecting alternative splicing

- Short conclusions
  - Different isoforms were made by different combination of splicing junctions (events)
  - Splicing junctions could be recovered by RNAseq reads

Theories of isoform-based algorithms

- What isoform-based algorithms do?
  - Predict transcripts
  - Predict expression level of transcripts
    - So that we can use frameworks of detecting differentially expressed genes to detect differentially expressed isoforms.





Theories of isoform-based algorithms

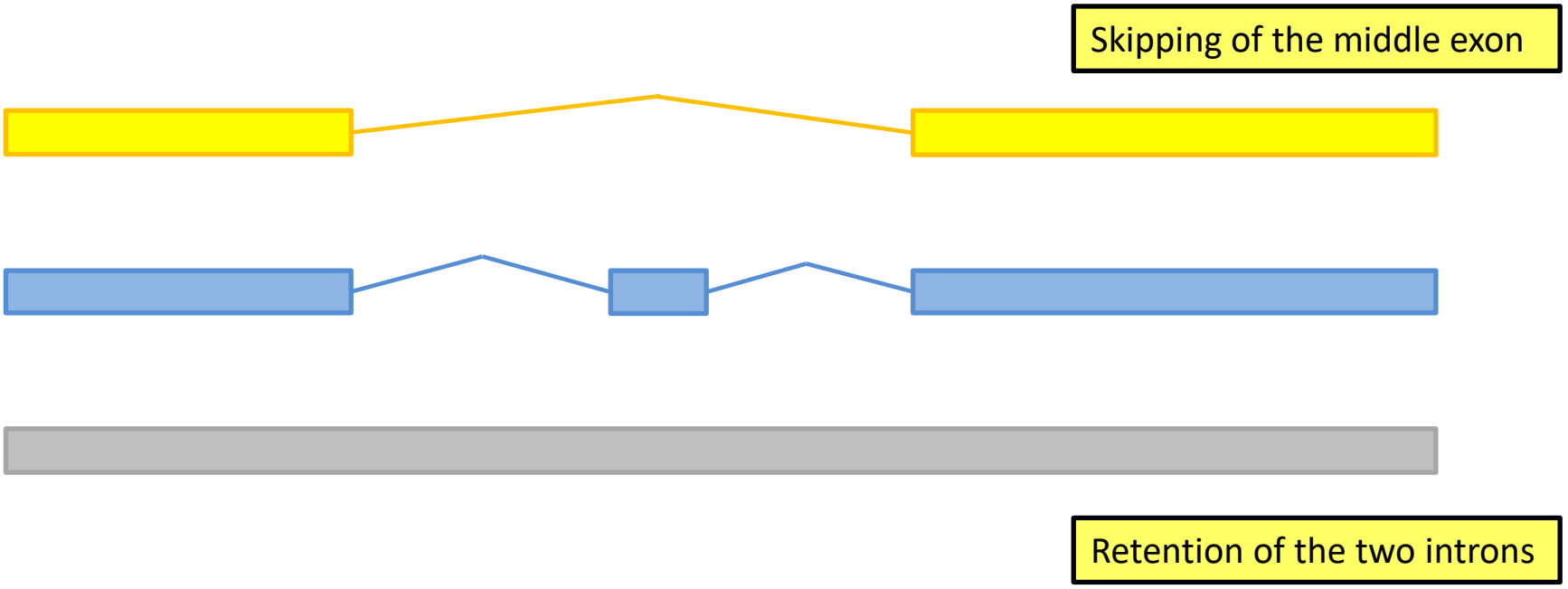
- Two of the best-known isoform-based algorithms
  - Cufflinks
    - Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation
      - Trapnell *et al.*, Nat Biotechnol. 2010
  - StringTie
    - StringTie enables improved reconstruction of a transcriptome from RNA-seq reads
      - Pertea *et al.*, Nat Biotechnol. 2015

Underlying theories of StringTie

- In this tutorial, we will go through underlying theories of StringTie
  - divides a gene region into segments (as nodes) based on splicing junctions expressed by reads
  - connect two nodes (genomic segments) if some reads are spanning them
  - treat the resulted graph as a graph of the maximum flow problem

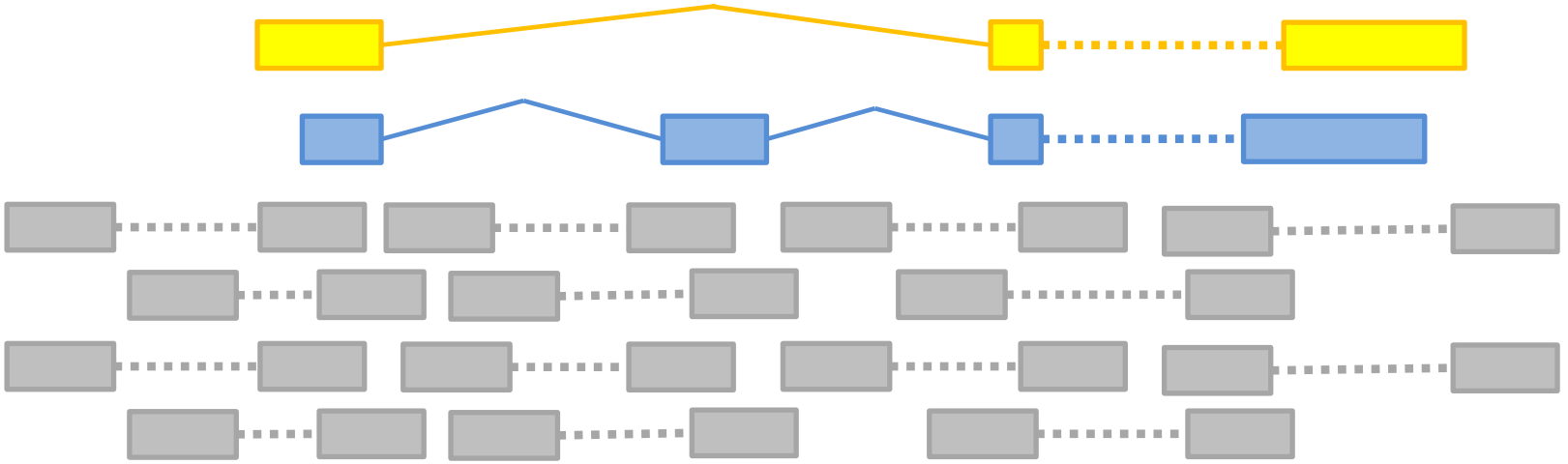
Underlying theories of StringTie

- Suppose that we have a gene locus, which can generate three isoforms



Underlying theories of StringTie

- Consider the following read pairs been mapped to the reference genome



Reads colored according to their source isoform

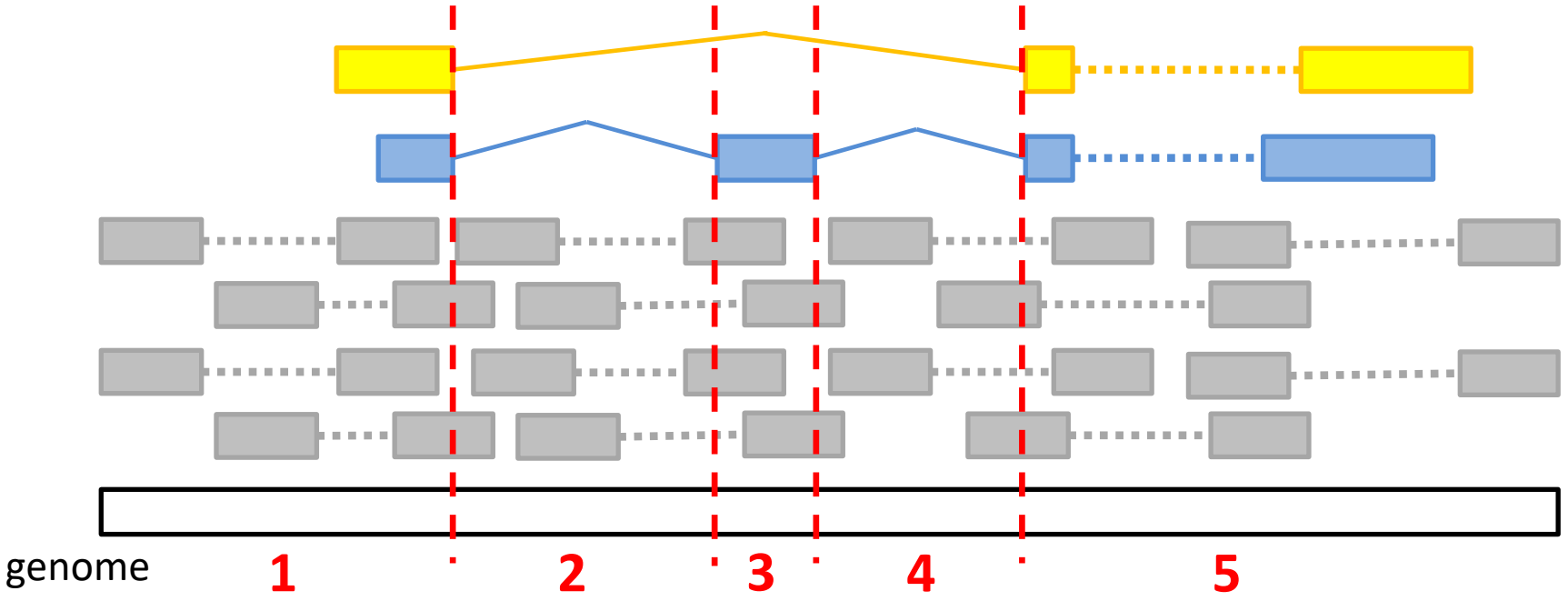
splicing junction

read pair

A remake of Fig1 of the Cufflinks 2010 paper

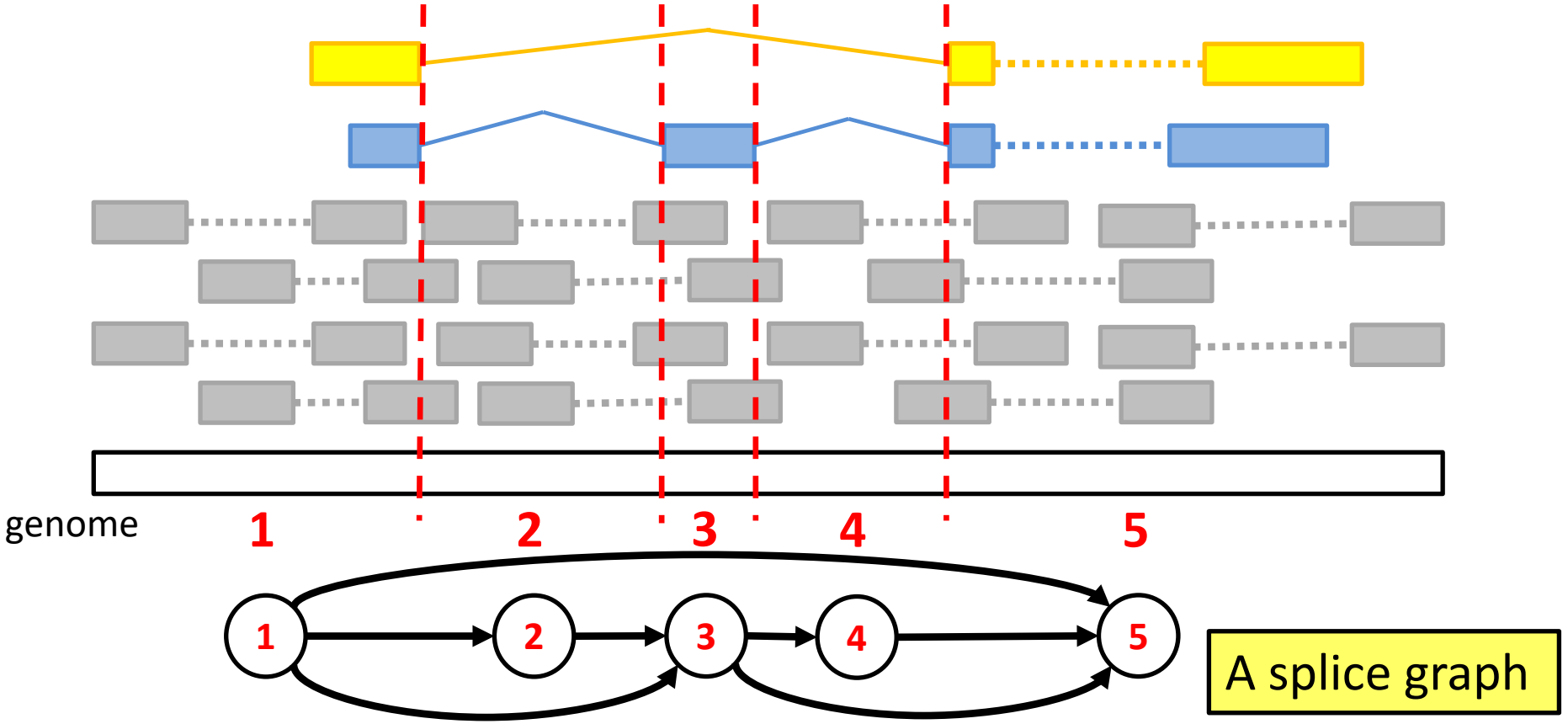
Underlying theories of StringTie

- The first step is to divide the gene region into segments



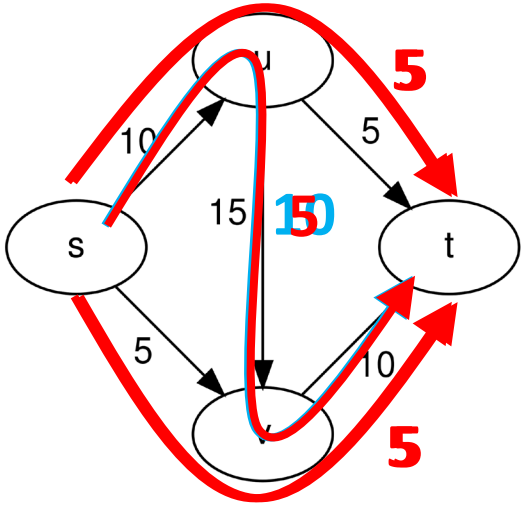
Underlying theories of StringTie

- By treating segments as nodes, connect two nodes if some reads are spanning them



Underlying theories of StringTie

- The next step is to transform the problem into a maximum flow problem
- What is a maximum flow problem?
  - "finding a *feasible* flow through a flow network that obtains the maximum possible flow rate"

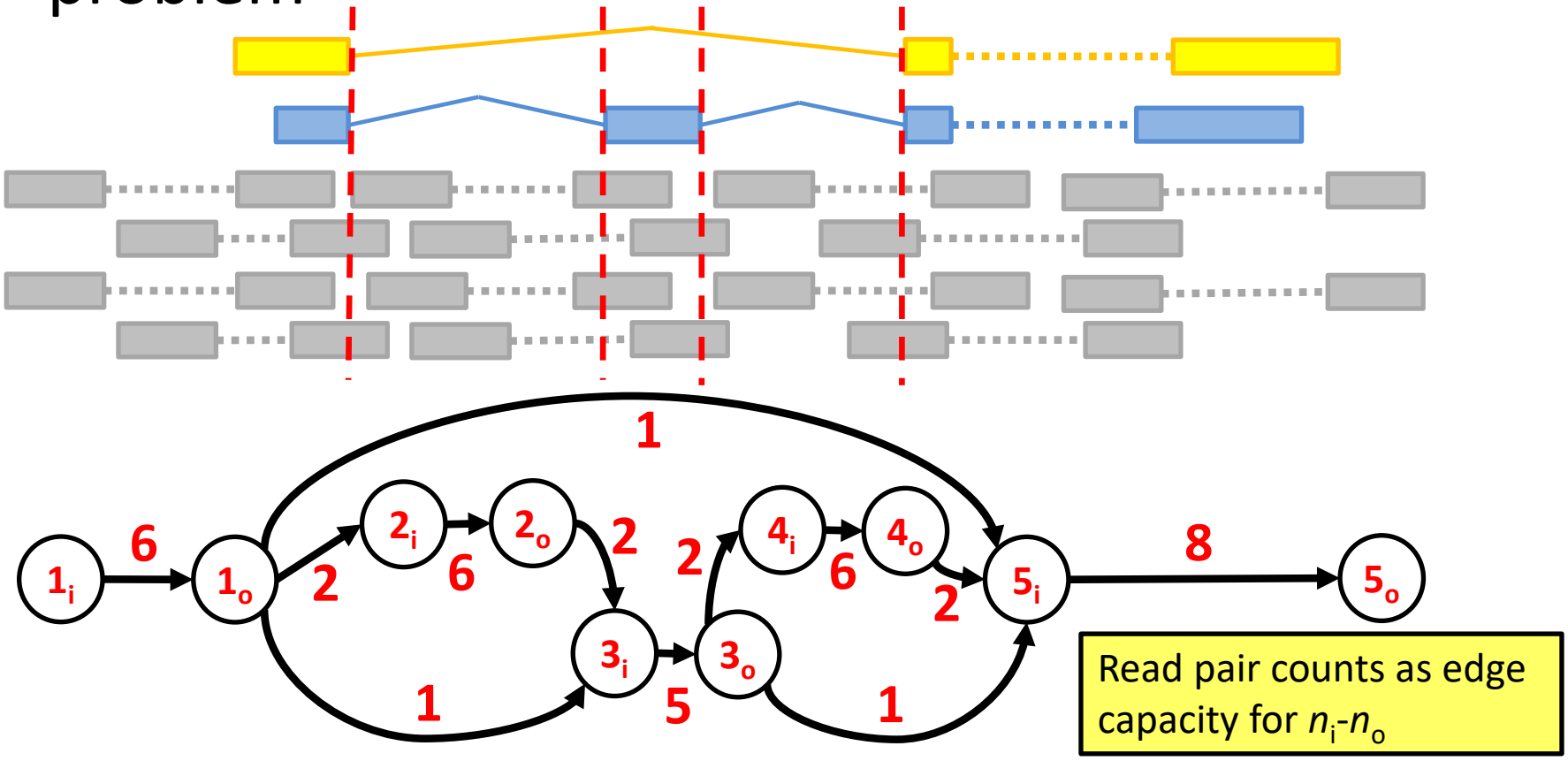


How much flow can be obtained from source to terminal?
(black numbers as *capacity*)

Source: wikipedia

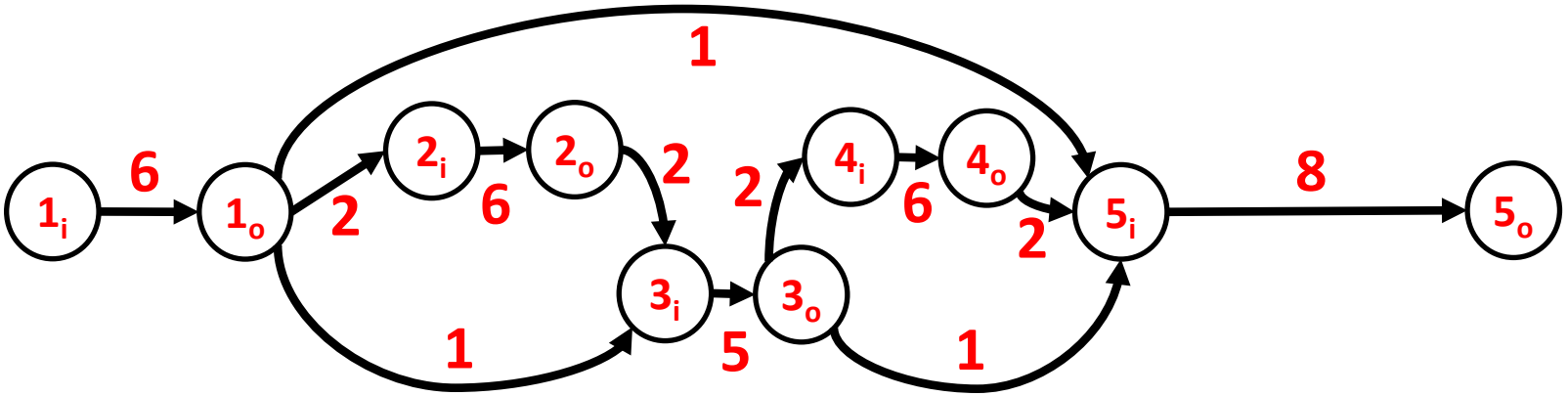
Underlying theories of StringTie

- Transform the graph into a maximum flow problem

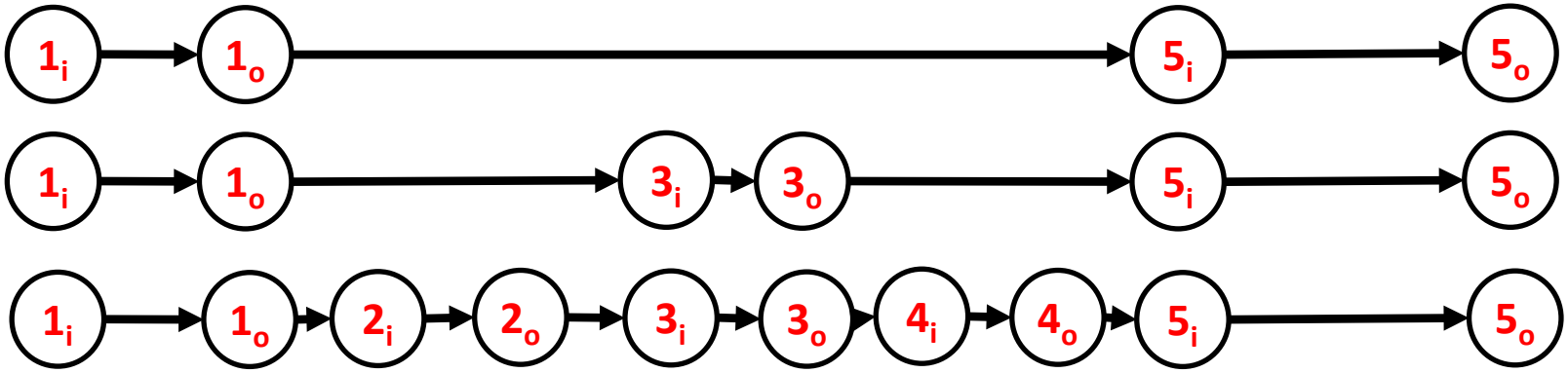


Underlying theories of StringTie

- The maximum flow?



- Three paths, with flows 1, 1, and 2, respectively



Underlying theories of StringTie

- By treating each path as an isoform, we would obtain the three isoforms
  - For each isoform, StringTie counts reads when computing the corresponding flow

  => expression levels of isoforms

Underlying theories of StringTie

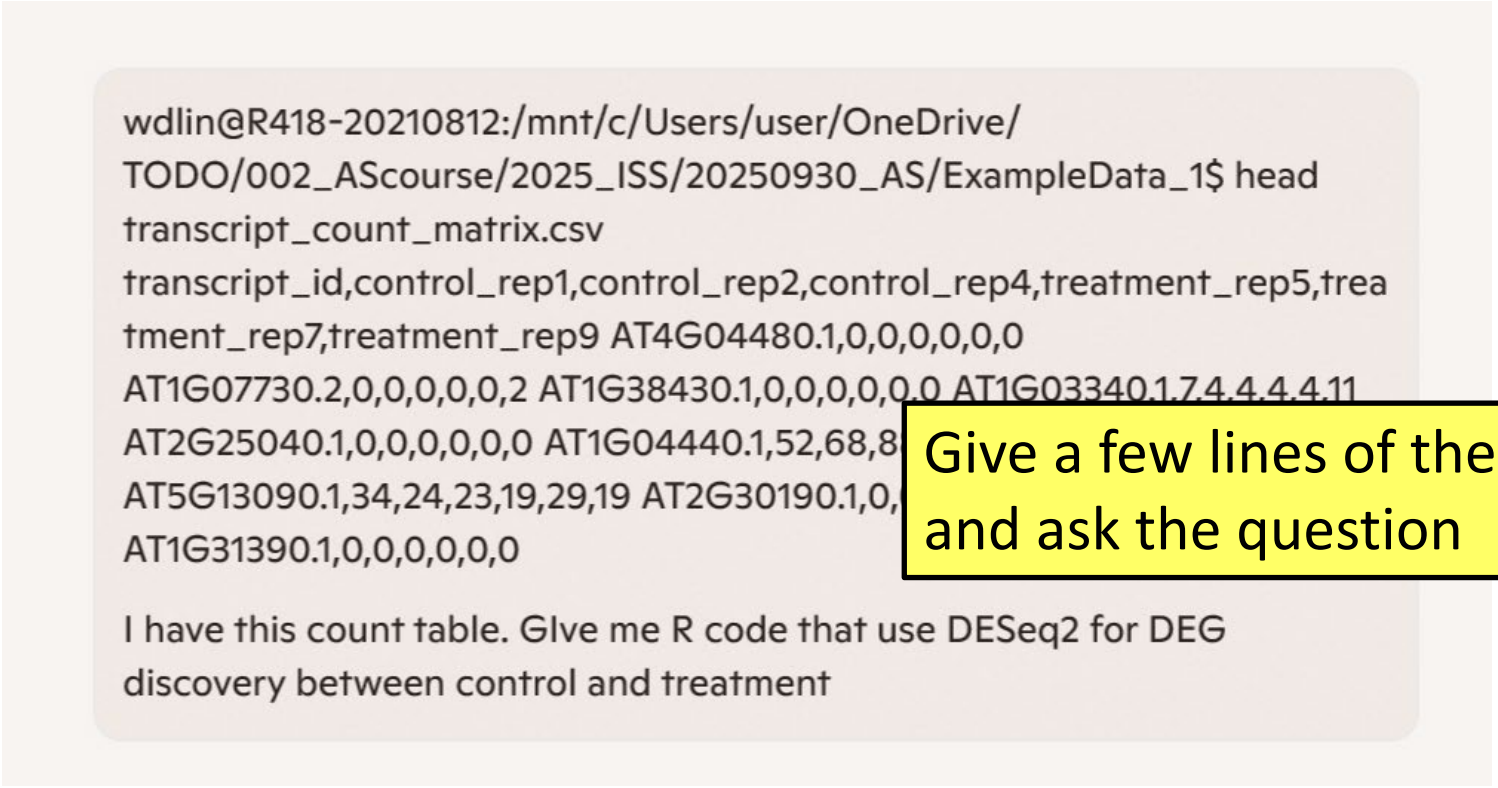
- A short conclusion
  - Reasonably transforming questions into some mathematical models could be helpful for solving problems.

Applications with isoform expression levels

- As StringTie provides two count tables
  - Read counts of genes
  - Read counts of transcripts
- What can we do with these two tables?
  - Detect differentially expressed genes/isoforms
  - Detect differentially *preferred* isoforms

Applications with isoform expression levels

- It might be an easy task to detect differentially expressed genes/isoforms in this AI era



Screen capture of using Copilot

Applications with isoform expression levels

- It might be an easy task to detect differentially expressed genes/isoforms in this AI era

DESeq2 Workflow for Transcript Count Matrix

R Copy

```r
# Load required libraries
library(DESeq2)

# Step 1: Read the count matrix
counts <- read.csv("transcript_count_matrix.csv", row.names = 1)

# Step 2: Define sample conditions
# Extract column names and assign conditions manually
col_data <- data.frame(
  row.names = colnames(counts),
  condition = c(rep("control", 3), rep("treatment", 3))
)

# Step 3: Create DESeq2 dataset
dds <- DESeqDataSetFromMatrix(countData = counts,
                              colData = col_data,
                              design = ~ condition)

# Step 4: Pre-filter low-count transcripts (optional but recommended)
dds <- dds[rowSums(counts(dds)) > 1, ]

# Step 5: Run DESeq2 pipeline
dds <- DESeq(dds)

# Step 6: Extract results
res <- results(dds)

# Step 7: Order by adjusted p-value
res_ordered <- res[order(res$padj), ]

# Step 8: Save results to CSV
write.csv(as.data.frame(res_ordered), file = "DESeq2_results.csv")

# Optional: View top DEGs
head(res_ordered)
```

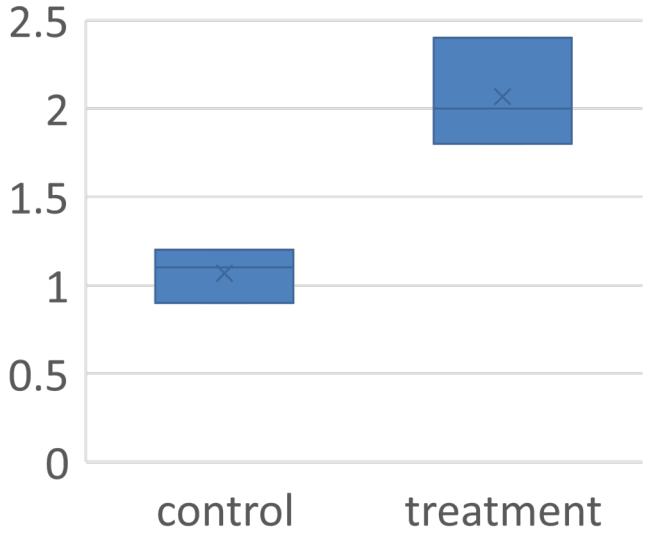
I checked the code. It is correct.

No worry. R code for differential gene/isoform discovery is included in our walkthrough.

Screen capture of using Copilot

Applications with isoform expression levels

- So, what is our understanding regarding to DEG discovery? And, how can we do better based on the understanding?

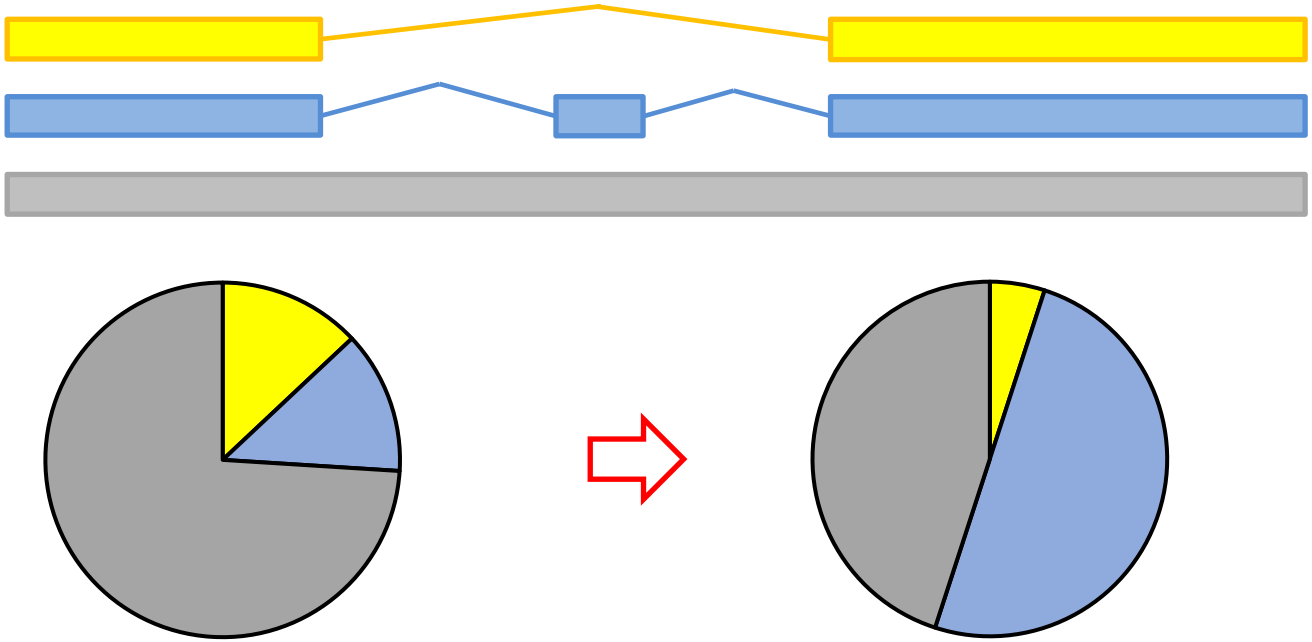


- We say a gene/isoform was differentially expressed if there is a difference generally between its expression levels in treatment and that in control.
- In statistics, the *null hypothesis* is

$$mean_{\text{treatment}} - mean_{\text{control}} = 0$$

Applications with isoform expression levels

- Can we apply the statistics idea of DEG discovery to discover *preference changes* of isoforms?



Applications with isoform expression levels

- KNOWledge and KNOW-how can help
  - 1. most DEG discovery based on log-count-per-million so the null hypothesis is actually

    $logCPM_{\text{treatment}} - logCPM_{\text{control}} = 0$
  - 2. the term “preference” means taking gene expression level as the background

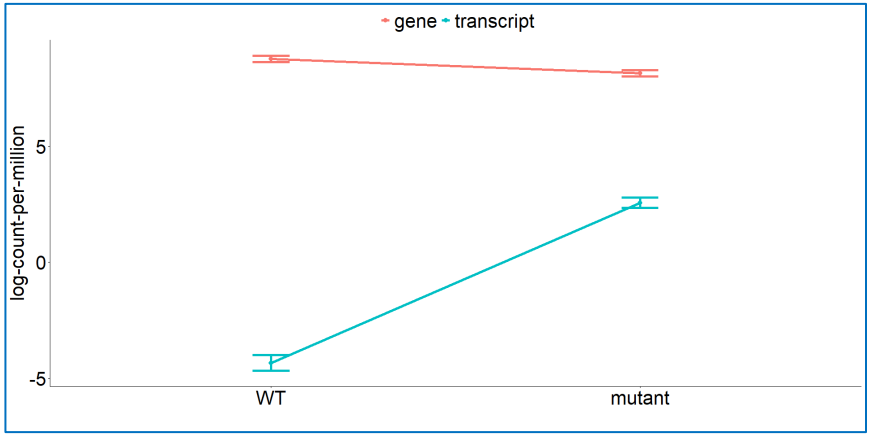
    Preference: $CPM_{\text{transcript}} / CPM_{\text{gene}}$
  - 3. a reasonable interpretation of “preference change” can be “fold-change”

    $(CPM_{\text{transcript,treat}} / CPM_{\text{gene,treat}}) / (CPM_{\text{transcript,ctrl}} / CPM_{\text{gene,ctrl}})$
  - 4. take log

    $(logCPM_{\text{transcript,treat}} - logCPM_{\text{gene,treat}}) - (logCPM_{\text{transcript,ctrl}} - logCPM_{\text{gene,ctrl}})$
  - 5. This is *difference of differences* and the *interaction term analysis* is exactly for it

Applications with isoform expression levels

- In walkthrough we have R code for the interaction term analysis
  - also an example that can be visually confirmed.
- Here is a real example of detecting differentially *preferred* isoforms



In this example, gene expression levels are about the same as isoform expression level altered.

NOTE: the interaction term idea can deal with cases even for gene expression level altered, *with respect to biological replicates*.

Applications with isoform expression levels

- Short conclusions
  - Isoform-based methods can be used for
    - detecting differentially expressed isoforms and
    - detecting differentially preferred isoforms.
  - AI can help us, and AI+knowledge can help us better.

Walk-through of the isoform-based algorithms

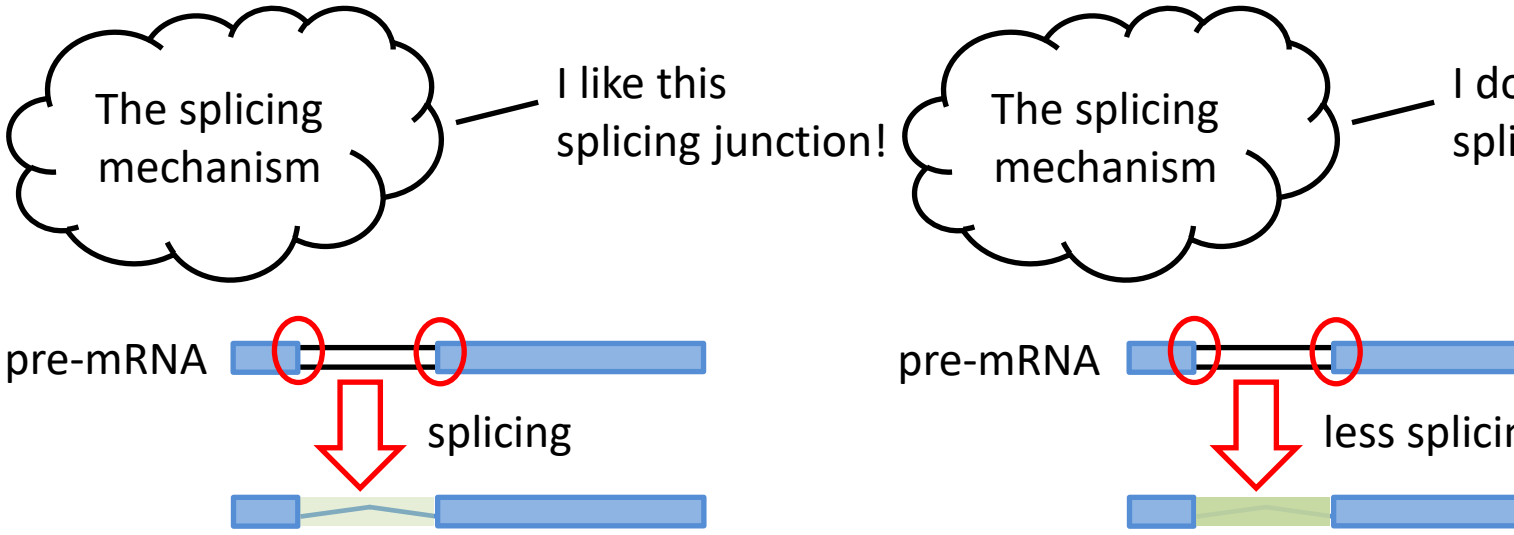
- Switch to file AS_20250930_walkthrough.pptx

Theories of event-based algorithms

- Cautions
  - This part contains methods that I have been applying for years in my works
    - But not general descriptions of event-based algorithms
  - All mentioned methods have been incorporated in a (few) number of papers
  - Software repository: RackJ
    - https://github.com/wdlingit/rackj/
    - Direct binary download: https://downloads.sourceforge.net/project/rackj/0.99c/rackJ.tar.gz

Theories of event-based algorithms

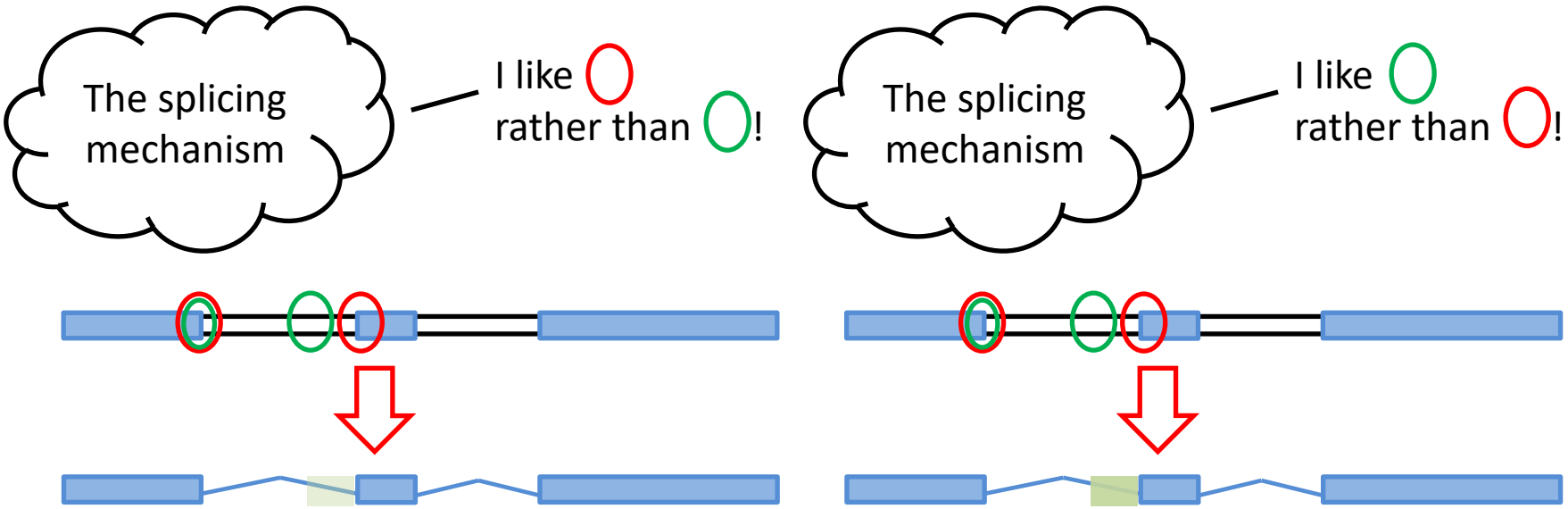
- The underlying thinking of the methods to be described is
  - to taking *preference* of the splicing mechanism into consideration



Alternative intron-retention

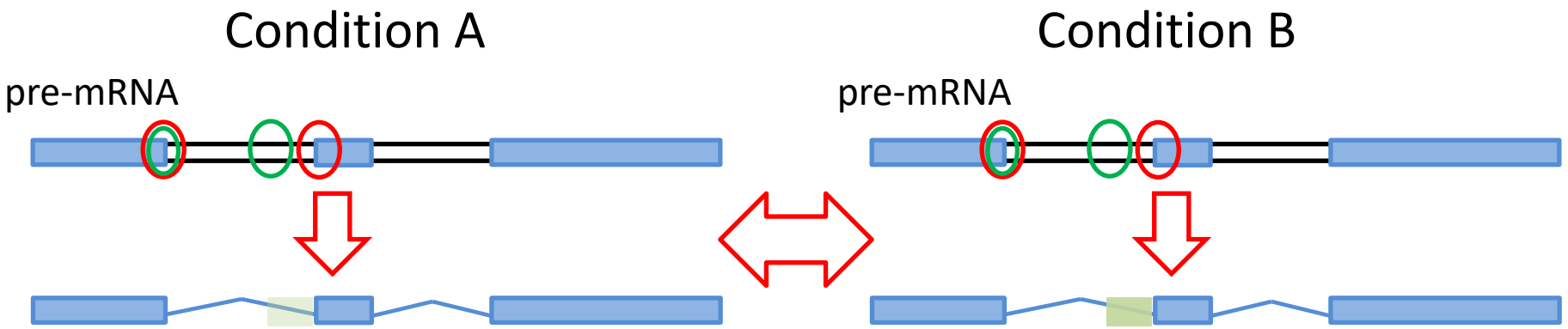
Theories of event-based algorithms

- Taking *preference* of the splicing mechanism into consideration.
  - another example on alternative accepter



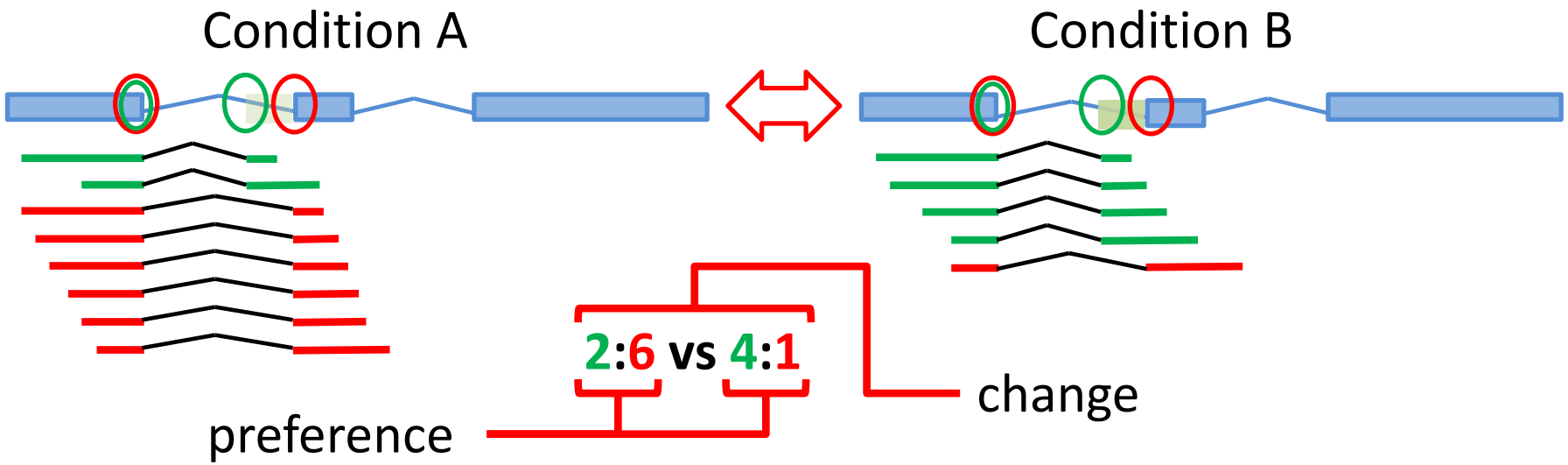
Theories of event-based algorithms

- Revisit the term “alternative”
  - change of splicing preference between two conditions
- The term “preference” means
  - the possibility of choosing something against some *background*.



Theories of event-based algorithms

- Take alternative donor/acceptor events as an example
  - The *preference* can be somehow measured by read counts
  - The *change of preference* can be measured by some statistical tests



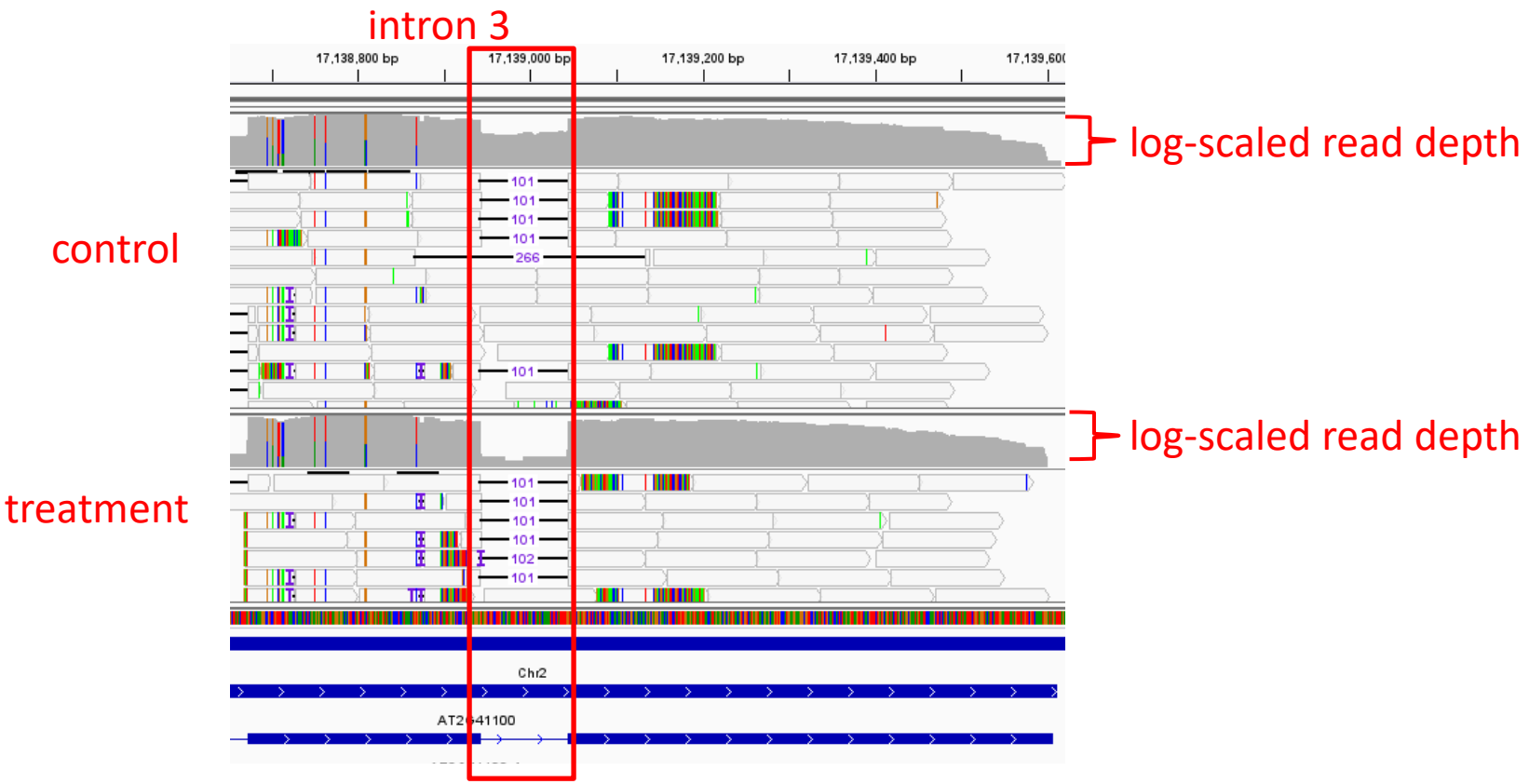
Theories of event-based algorithms

- In next slides
  - We show cases of alternative splicing comparisons of the example data
  - with visualization and explanation

Theories of event-based algorithms

- Alternative intron-retention

#GeneID	intronNo	intronLen	intronC	intronT	exonC	exonT	chiSquared	P-value
AT2G41100	3	101	34.0	1.9	223.4	160.2	19.6	9.41E-06



Theories of event-based algorithms

- Alternative intron-retention

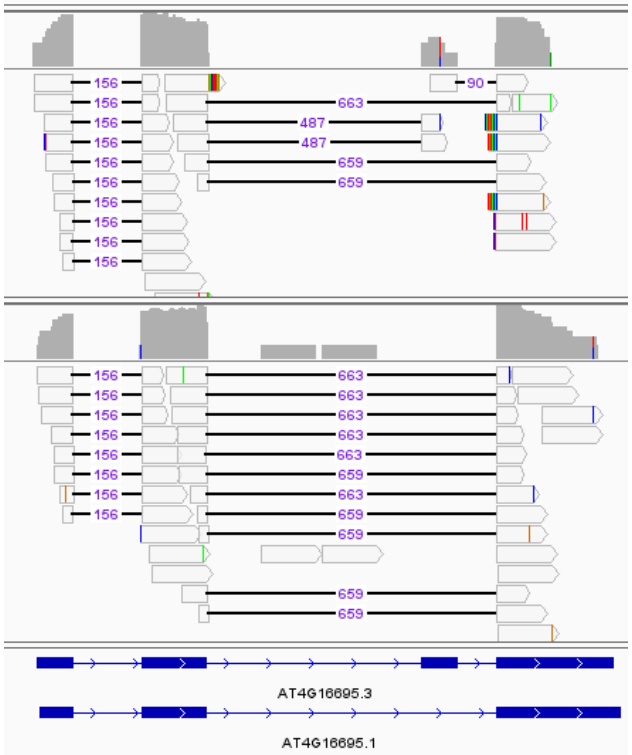
#GeneID	intronNo	intronLen	intronC	intronT	exonC	exonT	chiSquared	P-value
AT2G41100	3	101	34.0	1.9	223.4	160.2	19.6	9.41E-06

  - We computed read depths of an intron region (34.0 & 1.9) and took read depths of neighboring exons (223.4 & 160.2) as the background
  - Chi-squared test of *goodness of fit* was used to see if intron read depths are following the background
  - In English, to see if the chance of retaining the intron was changed between the two conditions.

Theories of event-based algorithms

- Alternative exon-skipping

#GeneID	exonPair	control	treatment	xControl	xTreatment	xChiSquared	P-value
AT4G16695	2<=>4	3	11	3	0	10.45249	0.001225



Theories of event-based algorithms

- Alternative exon-skipping

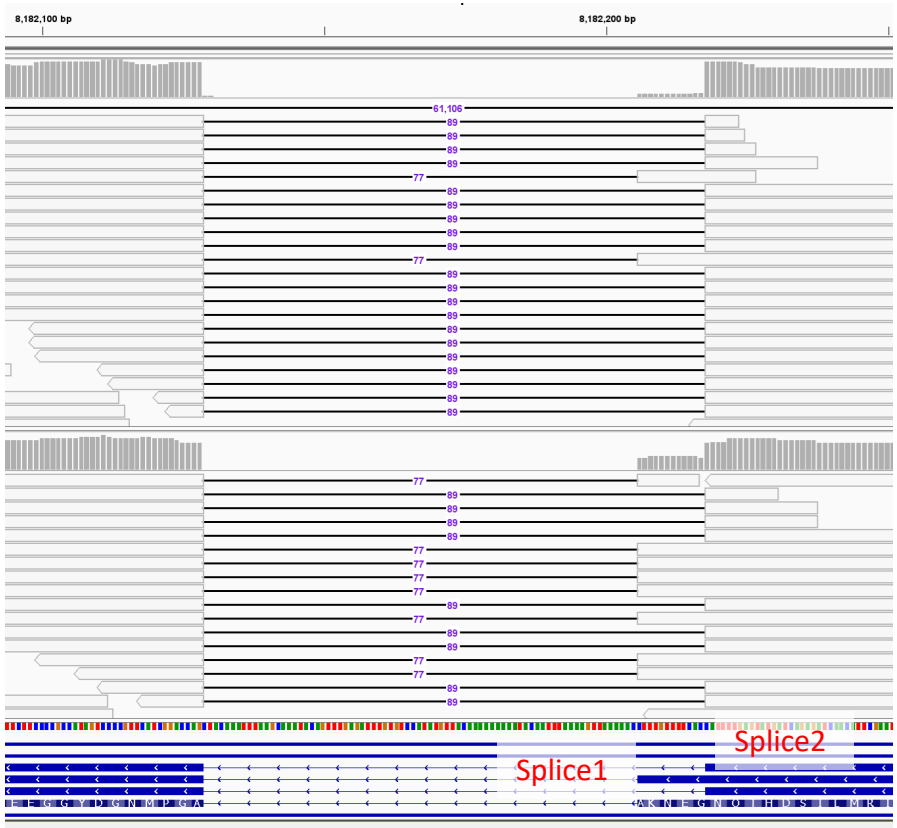
#GeneID	exonPair	control	treatment	xControl	xTreatment	xChiSquared	P-value
AT4G16695	2<=>4	3	11	3	0	10.45249	0.001225

  - We counted reads that are supporting the exon-skipping event (3 & 11) and reads not supporting the event (3 & 0)
  - Chi-squared test of *goodness of fit* was used to see if any of the two sets of numbers are not following the other
  - In English, to see if the chance of skipping (or not skipping) an exon was changed between the two conditions.

Theories of event-based algorithms

- Alternative donor/accepter change

#Genec	Splice1	Splice2	Ctr Splice1	Trt Splice1	Ctr SpliceO	Trt SpliceO	p-value
AT1G23080	2(0)-3(0)	2(0)-3(-12)	2	8	20	9	0.011



Theories of event-based algorithms

- Alternative donor/accepter change

#Genec	Splice1	Splice2	Ctr Splice1	Trt Splice1	Ctr SpliceO	Trt SpliceO	p-value
AT1G23080	2(0)-3(0)	2(0)-3(-12)	2	8	20	9	0.011

  - We counted reads that are supporting junction *splice1* "2(0)-3(0)"(2 & 8) and splice reads from the same exon pairs but not supporting *splice1* (20 & 9)
  - Fisher exact test was used to see if any of the two sets of numbers are not following the other
  - In English, to see if the chance of picking *splice1* as the splicing junction was changed between the two conditions.

Theories of event-based algorithms

- A short note
  - For the three types of AS comparisons
    - Intron retention
    - Exon skipping
    - Alternative donor/accepter
  - The applied statistical tests hold *the same null hypothesis*
    - the preference of the splicing event is the same between the two conditions
    - A literal interpretation on a significant P-value: it is *unlikely* the preference is the same between the two conditions

Theories of event-based algorithms

- Short conclusions
  - Event-based algorithms, at least as we presented, take RNAseq evidences *directly* for statistical comparisons
  - The presented event-based methods take the preference of the splicing mechanism into consideration
  - Our recent development also enables comparisons between sample groups
    - A choice of not merging biological replicates and taking replication into consideration

Walk-through of the event-based algorithms

- Switch to file AS_20250930_walkthrough.pptx

Discussions

- Isoform-based algorithms vs event-based algorithms, which kind of method to use?
  - This depends on your research purpose
    - Isoform-based algorithms predicts expression levels of transcripts
      - Overall results of splicing events per gene
    - Event-based algorithms should report changes that focus on splicing events
    - There should be no problem to do both of them at the same time
      - Always study the results carefully

Discussions

- Can we incorporate technologies like nanopore or PacBio in alternative splicing analyses?
  - The key should be the quality of results.
  - *Currently*, sequencing *error rates* of nanopore & PacBio were considered higher than that of Illumina
  - This may affect fitting of mapping records to exon boundaries
    - => alternative donor/accepter detection, and may be small exons

Appendix 1: dealing with long reads

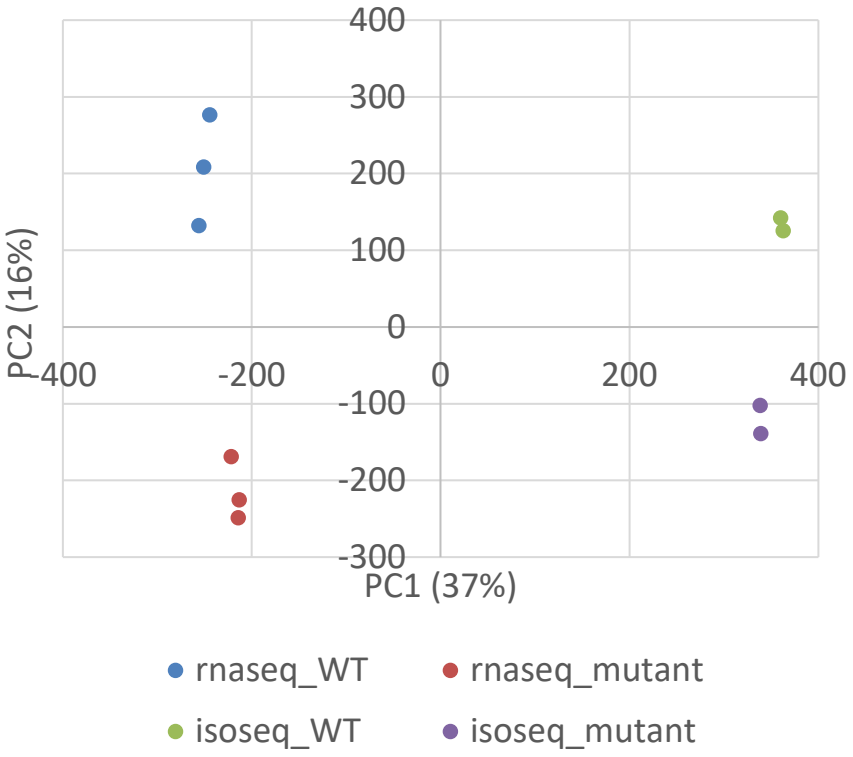
- We have a walkthrough for dealing with long read datasets of multiple samples
  - https://github.com/wdlingit/cop/wiki/Summarize-ISOseq-isoforms
  - Based on methods applied in Huang *et al.*, Genome Biology. 2022
- In short, the procedure summarizes isoforms across all samples and generate a count matrix of all isoforms.

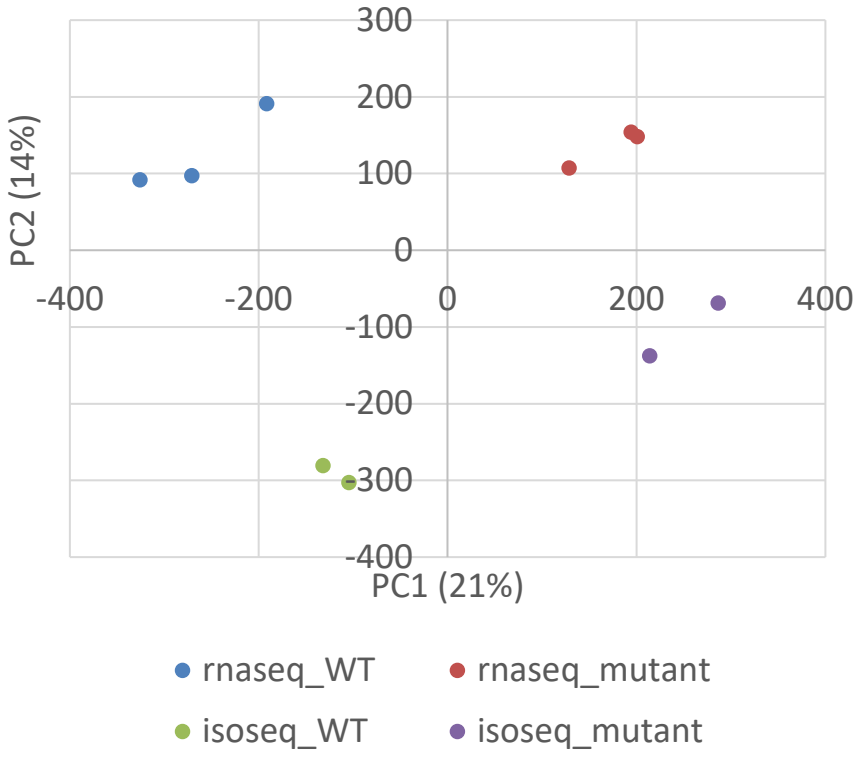
Appendix 2: dealing with short reads and long reads at the same time

- The following steps are from my currently best practice
  1. Apply the procedure in the last page for long reads
     - so that we can update genome annotation and have a read count table of long reads
  2. Use StringTie with the updated genome annotation for short reads
     - so that we have a read count table of short reads of the same isoforms with the table of long reads
  3. Combine the two tables and perform some necessary batch-effect correction
     - Long read and short read are of different technologies. There should be some batch effect.
  4. Apply the corrected values for downstream computations.

Appendix 2: dealing with short reads and long reads at the same time

- Before and after batch correction





Finally

- Thank you for your attentions.
- I am willing to answer and/or discuss questions via email or in some other interactive form.
  - Please don’t hesitate to let me know if you have any questions.