Complexity Analysis of the Lasso Regularization Path

Isoform Discovery from RNA-Seq Data

Julien Mairal, Inria, Grenoble

Statslab seminar, Cambridge, 2015

Collaborators

Bin Yu

Elsa Bernard

Laurent Jacob

Jean-Philippe Vert

First part: a curiosity

- J. Mairal and B. Yu. Complexity Analysis of the Lasso Regularization Path. *Proc. ICML*. 2012.

Second part: a useful application of sparsity

- E. Bernard, L. Jacob, J. Mairal, and J-P. Vert. Efficient RNA Isoform Identification and Quantification from RNA-Seq Data with Network Flows. *Bioinformatics*. 2014.

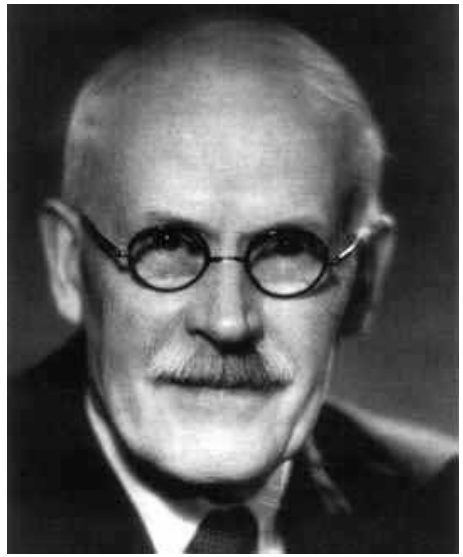
Part I: Complexity Analysis of the Lasso Regularization Path

joint work with Bin Yu from UC Berkeley

- J. Mairal and B. Yu. Complexity Analysis of the Lasso Regularization Path. *Proc. ICML*. 2012.

Early thoughts about parsimony

(a) Dorothy Wrinch
1894–1980



(b) Harold Jeffreys
1891–1989

The existence of simple laws is, then, apparently, to be regarded as a quality of nature; and accordingly we may infer that it is justifiable to prefer a simple law to a more complex one that fits our observations slightly better.

[Wrinch and Jeffreys, 1921]. Philosophical Magazine Series.

Historical overview of parsimony

- 14th century: Ockham's razor;
- 1921: Wrinch and Jeffreys' simplicity principle;
- 1952: Markowitz's portfolio selection;
- 60 and 70's: best subset selection in statistics;
- 70's: use of the ℓ_1-norm for signal recovery in geophysics;
- 90's: wavelet thresholding in signal processing;
- 1996: Olshausen and Field's dictionary learning;
- 1996–1999: Lasso (statistics) and basis pursuit (signal processing);
- 2006: compressed sensing (signal processing) and Lasso consistency (statistics);

What this work is about

- another paper about the Lasso/Basis Pursuit [Tibshirani, 1996, Chen et al., 1999]:

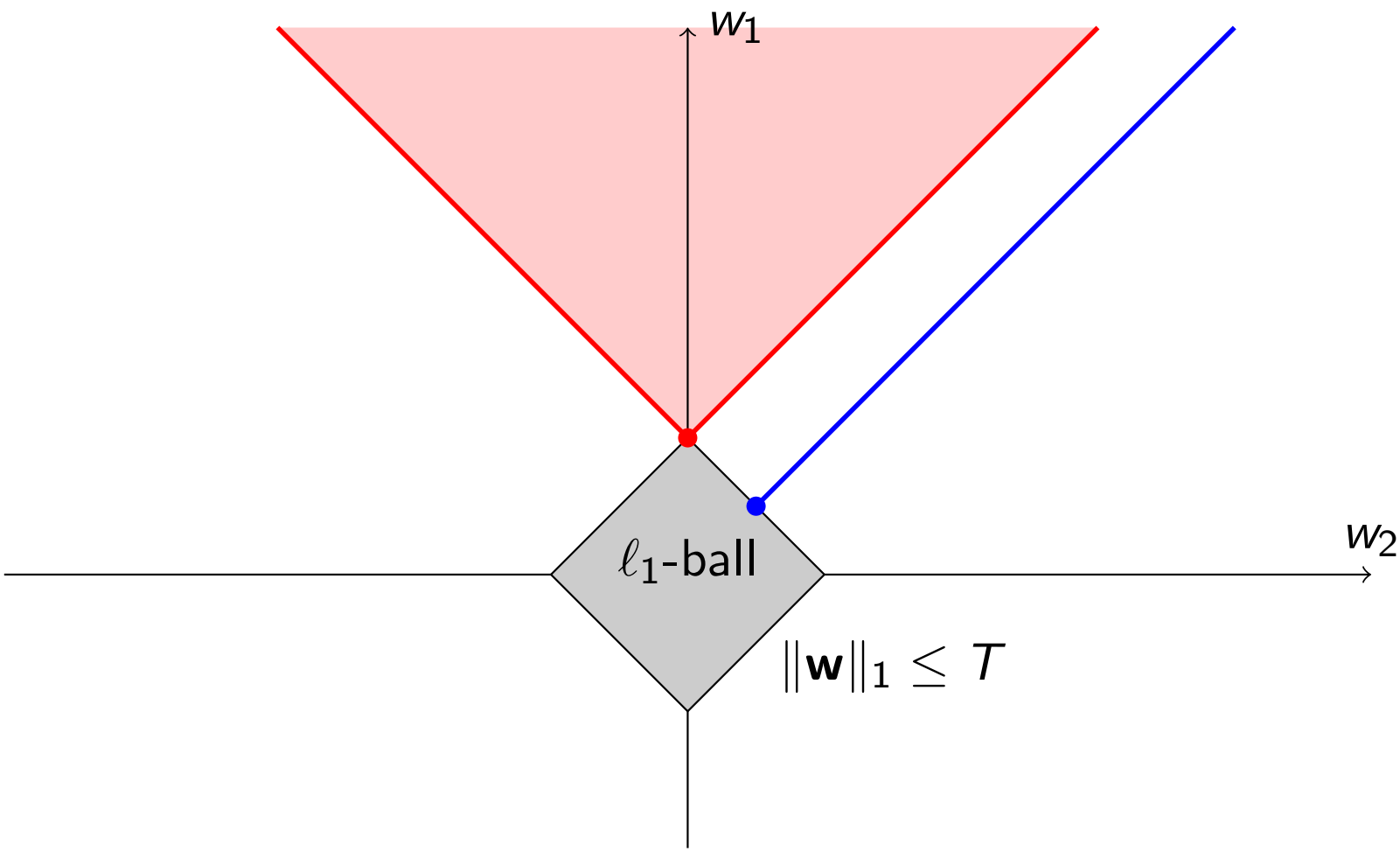
$$\min_{\mathbf{w}\in\mathbb{R}^p} \frac{1}{2}\|\mathbf{y}-\mathbf{X}\mathbf{w}\|_2^2 + \lambda\|\mathbf{w}\|_1; \tag{1}$$

- the first complexity analysis of the homotopy method [Ritter, 1962, Osborne et al., 2000, Efron et al., 2004] for solving (1);

A story similar to

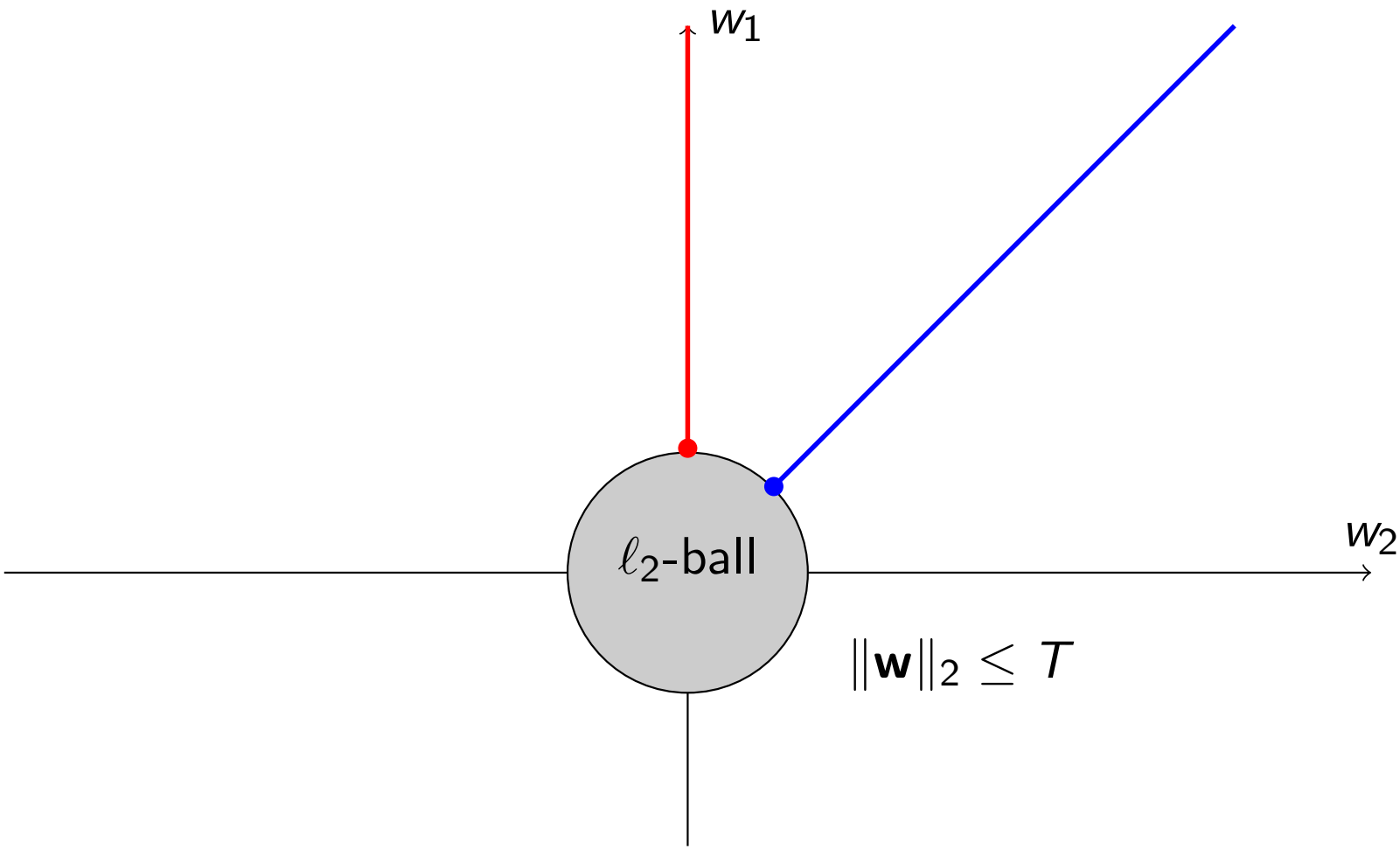
- the simplex algorithm for linear programs [Klee and Minty, 1972];
- the SVM regularization path [Gärtner, Jaggi, and Maria, 2010].

Regularizing with the ℓ_1-norm



The projection onto a convex set is "biased" towards singularities.

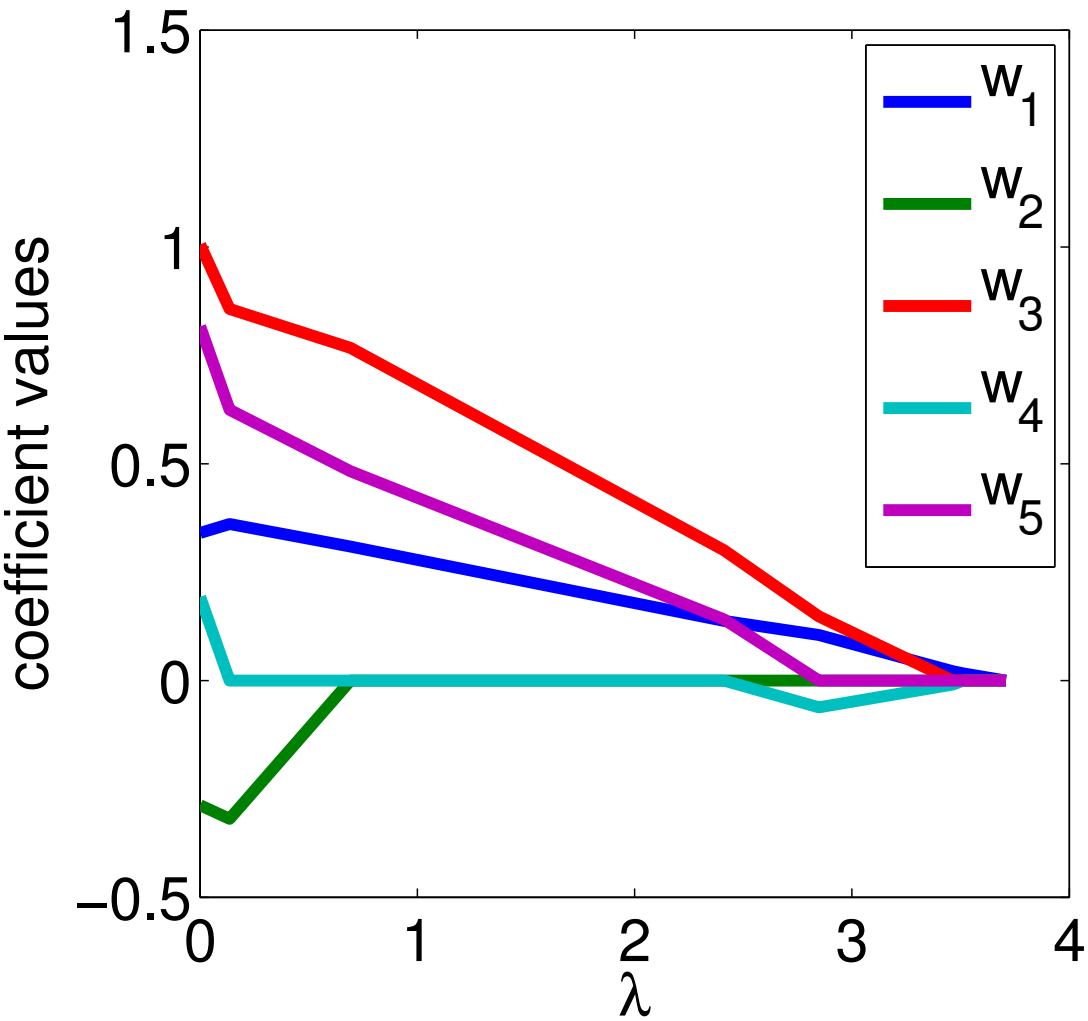
Regularizing with the ℓ_2-norm



The ℓ_2-norm is isotropic.

The Lasso Regularization Path and the Homotopy

Under uniqueness assumption of the Lasso solution, the regularization path is piecewise linear:



Our Main Results

Theorem - worst case analysis

In the worst-case, the regularization path of the Lasso has exactly $(3^p + 1)/2$ *linear segments.*

Proposition - approximate analysis

There exists an ε*-approximate path with* $O(1/\sqrt{\varepsilon})$ *linear segments.*

Brief Introduction to the Homotopy Algorithm

Piecewise linearity

Under uniqueness assumptions of the Lasso solution, the regularization path $\lambda \mapsto \mathbf{w}^\star(\lambda)$ is continuous and piecewise linear.

Brief Introduction to the Homotopy Algorithm

Piecewise linearity

Under uniqueness assumptions of the Lasso solution, the regularization path $\lambda \mapsto \mathbf{w}^\star(\lambda)$ is continuous and piecewise linear.

Recipe of the homotopy method - main ideas

1. finds a trivial solution $\mathbf{w}^\star(\lambda_\infty) = 0$ with $\lambda_\infty = \|\mathbf{X}^\top \mathbf{y}\|_\infty$;
2. compute the direction of the current linear segment of the path;
3. follow the direction of the path by decreasing λ;
4. stop at the next "kink" and go back to 2.

Brief Introduction to the Homotopy Algorithm

Piecewise linearity

Under uniqueness assumptions of the Lasso solution, the regularization path $\lambda \mapsto \mathbf{w}^\star(\lambda)$ is continuous and piecewise linear.

Recipe of the homotopy method - main ideas

1. finds a trivial solution $\mathbf{w}^\star(\lambda_\infty) = 0$ with $\lambda_\infty = \|\mathbf{X}^\top \mathbf{y}\|_\infty$;
2. compute the direction of the current linear segment of the path;
3. follow the direction of the path by decreasing λ;
4. stop at the next "kink" and go back to 2.

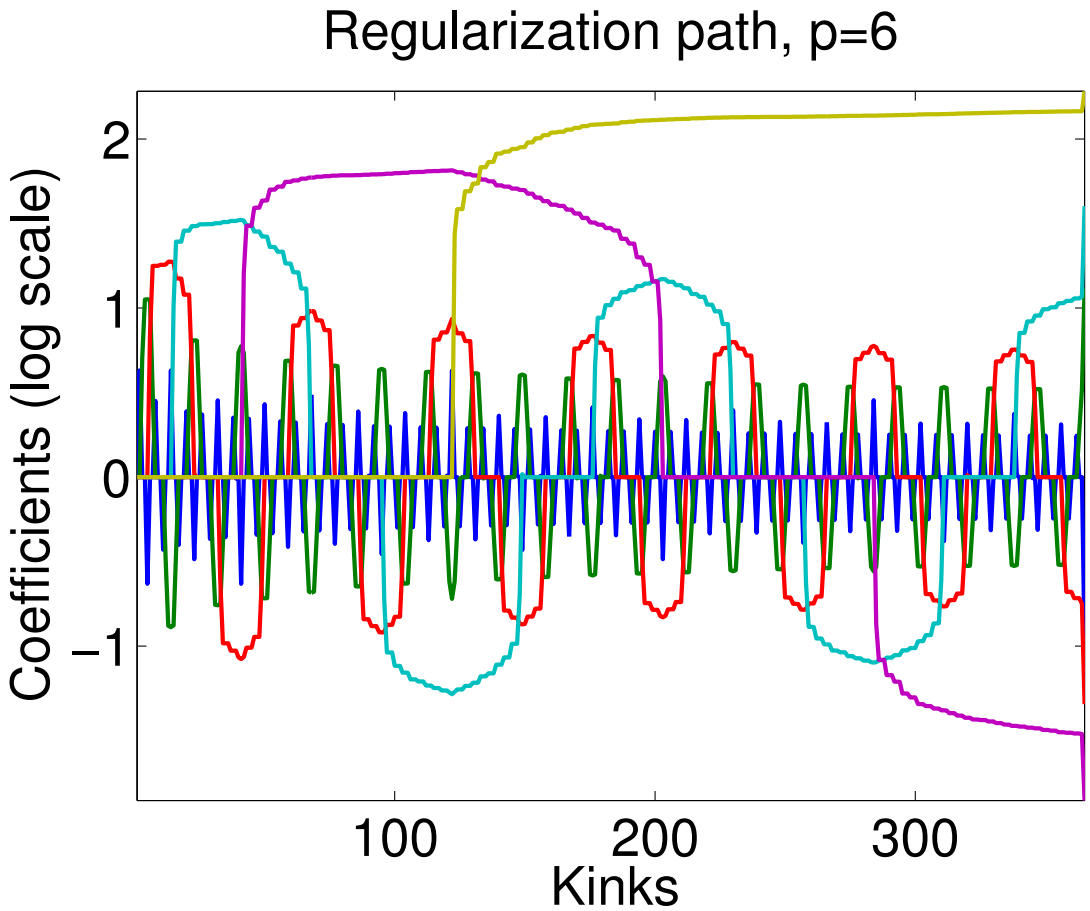
Caveats

- kinks can be very close to each other;
- the direction of the path can involve ill-conditioned matrices;
- worst-case exponential complexity (main result of this work).

Worst case analysis

Theorem - worst case analysis

In the worst-case, the regularization path of the Lasso has exactly $(3^p + 1)/2$ *linear segments.*



Worst case analysis

Consider a Lasso problem ($\mathbf{y} \in \mathbb{R}^n$, $\mathbf{X} \in \mathbb{R}^{n \times p}$).
Define the vector $\tilde{\mathbf{y}}$ in $\mathbb{R}^{n+1}$ and the matrix $\tilde{\mathbf{X}}$ in $\mathbb{R}^{(n+1)\times(p+1)}$ as follows:

$$\tilde{\mathbf{y}} \triangleq \begin{bmatrix} \mathbf{y} \\ y_{n+1} \end{bmatrix}, \quad \tilde{\mathbf{X}} \triangleq \begin{bmatrix} \mathbf{X} & 2\alpha\mathbf{y} \\ 0 & \alpha y_{n+1} \end{bmatrix},$$

where $y_{n+1} \neq 0$ and $0 < \alpha < \lambda_1/(2\mathbf{y}^\top\mathbf{y} + y_{n+1}^2)$.

Adverserial strategy

If the regularization path of the Lasso ($\mathbf{y}$,$\mathbf{X}$) has k linear segments, the path of ($\tilde{\mathbf{y}}, \tilde{\mathbf{X}}$) has $3k - 1$ linear segments.

Worst case analysis

$$\tilde{\mathbf{y}} \triangleq \begin{bmatrix} \mathbf{y} \\ y_{n+1} \end{bmatrix}, \quad \tilde{\mathbf{X}} \triangleq \begin{bmatrix} \mathbf{X} & 2\alpha\mathbf{y} \\ 0 & \alpha y_{n+1} \end{bmatrix},$$

Let us denote by $\{\boldsymbol{\eta}^1, \ldots, \boldsymbol{\eta}^k\}$ the sequence of k sparsity patterns in $\{-1, 0, 1\}^p$ encountered along the path of the Lasso $(\mathbf{y}, \mathbf{X})$.

The new sequence of sparsity patterns for $(\tilde{\mathbf{y}}, \tilde{\mathbf{X}})$ is

$$\Bigg\{ \overbrace{\begin{bmatrix} \boldsymbol{\eta}^1 = 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\eta}^2 \\ 0 \end{bmatrix}, \ldots, \begin{bmatrix} \boldsymbol{\eta}^k \\ 0 \end{bmatrix}}^{\text{first } k \text{ patterns}}, \overbrace{\begin{bmatrix} \boldsymbol{\eta}^k \\ 1 \end{bmatrix}, \begin{bmatrix} \boldsymbol{\eta}^{k-1} \\ 1 \end{bmatrix}, \ldots, \begin{bmatrix} \boldsymbol{\eta}^1 = 0 \\ 1 \end{bmatrix}}^{\text{middle } k \text{ patterns}},$$

$$\underbrace{\begin{bmatrix} -\boldsymbol{\eta}^2 \\ 1 \end{bmatrix}, \begin{bmatrix} -\boldsymbol{\eta}^3 \\ 1 \end{bmatrix}, \ldots, \begin{bmatrix} -\boldsymbol{\eta}^k \\ 1 \end{bmatrix}}_{\text{last } k-1 \text{ patterns}} \Bigg\}.$$

Worst case analysis

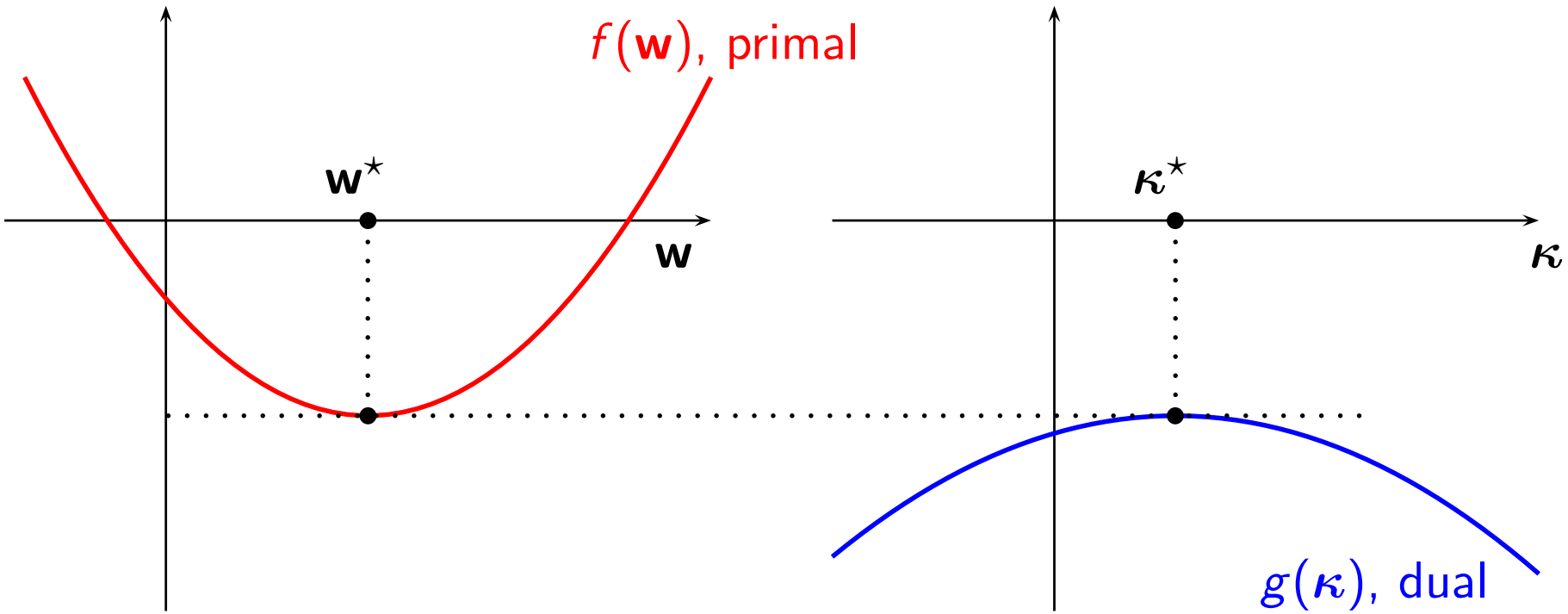
We are now in shape to build a pathological path with $(3^p+1)/2$ linear segments. Note that this lower-bound complexity is tight.

$$\mathbf{y} \triangleq \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad \mathbf{X} \triangleq \begin{bmatrix} \alpha_1 & 2\alpha_2 & 2\alpha_3 & \dots & 2\alpha_p \\ 0 & \alpha_2 & 2\alpha_3 & \dots & 2\alpha_p \\ 0 & 0 & \alpha_3 & \dots & 2\alpha_p \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \alpha_p \end{bmatrix},$$

Approximate Complexity

Refinement of Giesen, Jaggi, and Laue [2010] for the Lasso

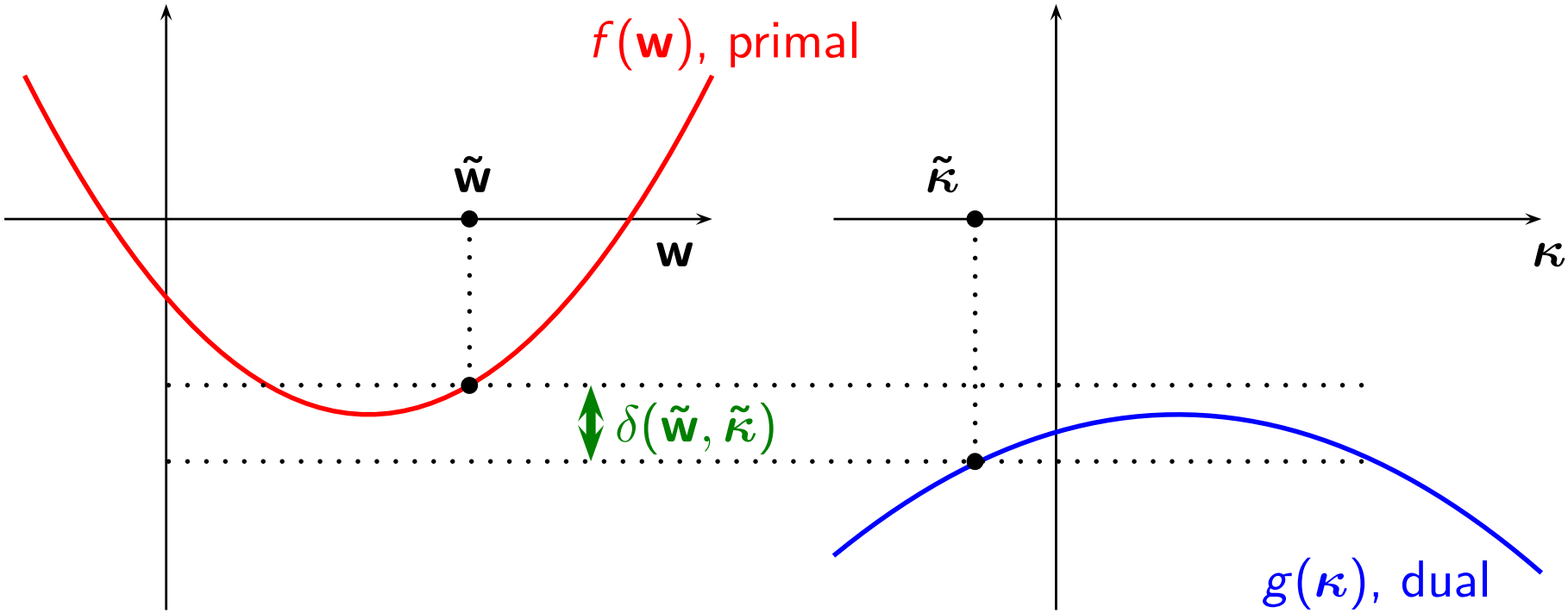
Strong Duality



Strong duality means that $\max_{\boldsymbol{\kappa}} g(\boldsymbol{\kappa}) = \min_{\mathbf{w}} f(\mathbf{w})$

Approximate Complexity

Duality Gaps



Strong duality means that $\max_{\kappa} g(\kappa) = \min_{\mathbf{w}} f(\mathbf{w})$

The duality gap guarantees us that $0 \leq f(\tilde{\mathbf{w}}) - f(\mathbf{w}^\star) \leq \delta(\tilde{\mathbf{w}}, \tilde{\kappa})$.

Approximate Complexity

$$\min_{\mathbf{w}} \left\{ f_\lambda(\mathbf{w}) \triangleq \frac{1}{2}\|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 + \lambda\|\mathbf{w}\|_1 \right\}, \qquad \text{(primal)}$$

$$\max_{\boldsymbol{\kappa}} \left\{ g_\lambda(\boldsymbol{\kappa}) \triangleq -\frac{1}{2}\boldsymbol{\kappa}^\top\boldsymbol{\kappa} - \boldsymbol{\kappa}^\top\mathbf{y} \;\text{ s.t. }\; \|\mathbf{X}^\top\boldsymbol{\kappa}\|_\infty \leq \lambda \right\}. \qquad \text{(dual)}$$

ε-approximate solution

$\mathbf{w}$ satisfies $APPROX_\lambda(\varepsilon)$ when there exists a dual variable $\boldsymbol{\kappa}$ s.t.

$$\delta_\lambda(\mathbf{w}, \boldsymbol{\kappa}) = f_\lambda(\mathbf{w}) - g_\lambda(\boldsymbol{\kappa}) \leq \varepsilon f_\lambda(\mathbf{w}).$$

ε-approximate path

A path $\mathcal{P} : \lambda \mapsto \mathbf{w}(\lambda)$ is an approximate path if it always contains ε-approximate solutions.

(see Giesen et al. [2010] for generic results on approximate paths)

Approximate Complexity

Main relation

$$APPROX_\lambda(0) \Longrightarrow APPROX_{\lambda(1-\sqrt{\varepsilon})}(\varepsilon)$$

Key: find an appropriate dual variable $\boldsymbol{\kappa}(\mathbf{w})$ + simple calculation;

Proposition - approximate analysis

there exists an ε-approximate path with at most $\left\lceil \frac{\log(\lambda_\infty/\lambda_1)}{\sqrt{\varepsilon}} \right\rceil$ *segments.*

Approximate homotopy - main ideas

- Maintain approximate optimality conditions along the path;
- Make steps in λ greater than or equal to $\lambda(1-\theta\sqrt{\varepsilon})$;
- When the kinks are too close to each other, make a large step and switch to first-order method;

A Few Messages to Conclude

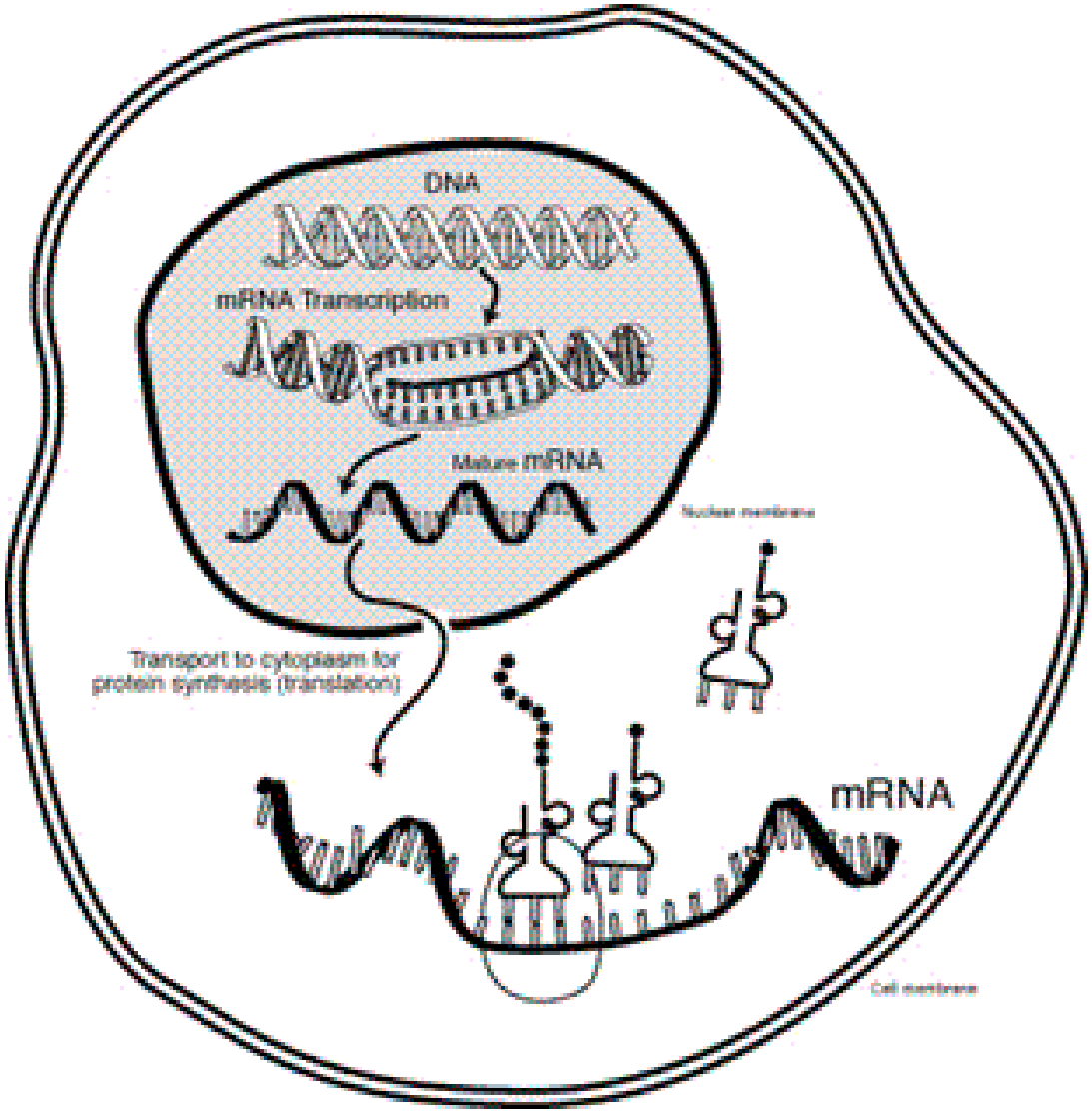
- **Despite its exponential complexity, the homotopy algorithm remains extremely powerful in practice**;
- numerical stability is still an issue of the homotopy algorithm;
- when one does not care about precision, the worst-case complexity of the path can be significantly reduced.

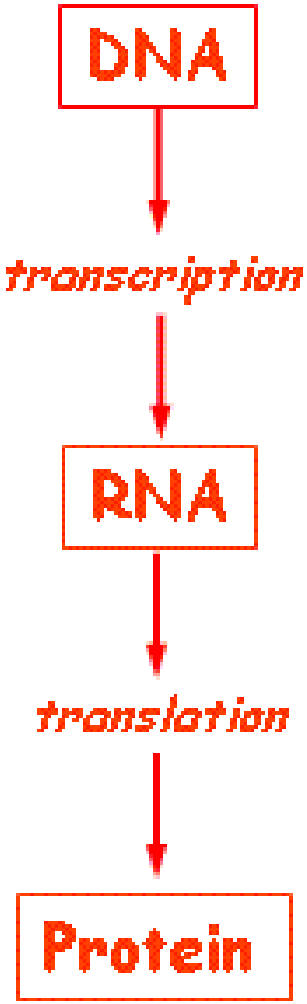
Part II: Isoform Discovery from RNA-Seq Data with Network Flows

joint work with **Elsa Bernard** (Institut Curie), Laurent Jacob (CNRS) and Jean-Philippe Vert (Institut Curie)

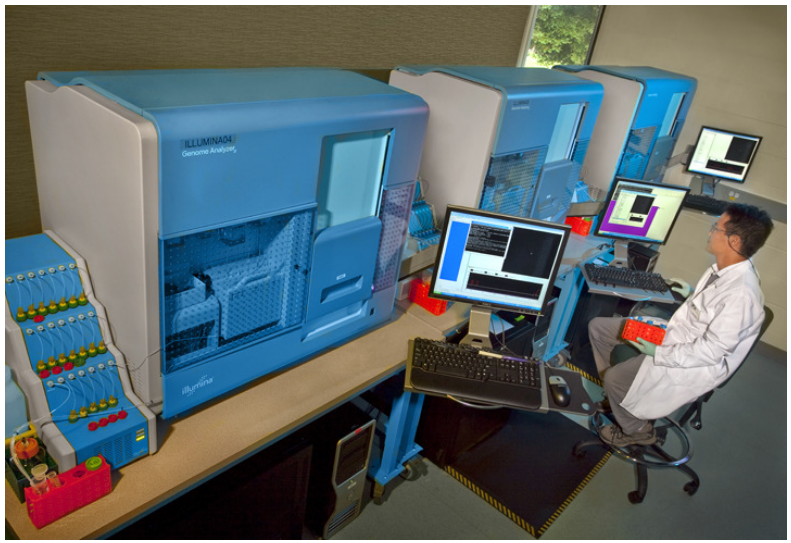
- E. Bernard, L. Jacob, J. Mairal, and J-P. Vert. Efficient RNA Isoform Identification and Quantification from RNA-Seq Data with Network Flows. *Bioinformatics*. 2014.

DNA Transcription/Translation (Central Dogma, 1958)





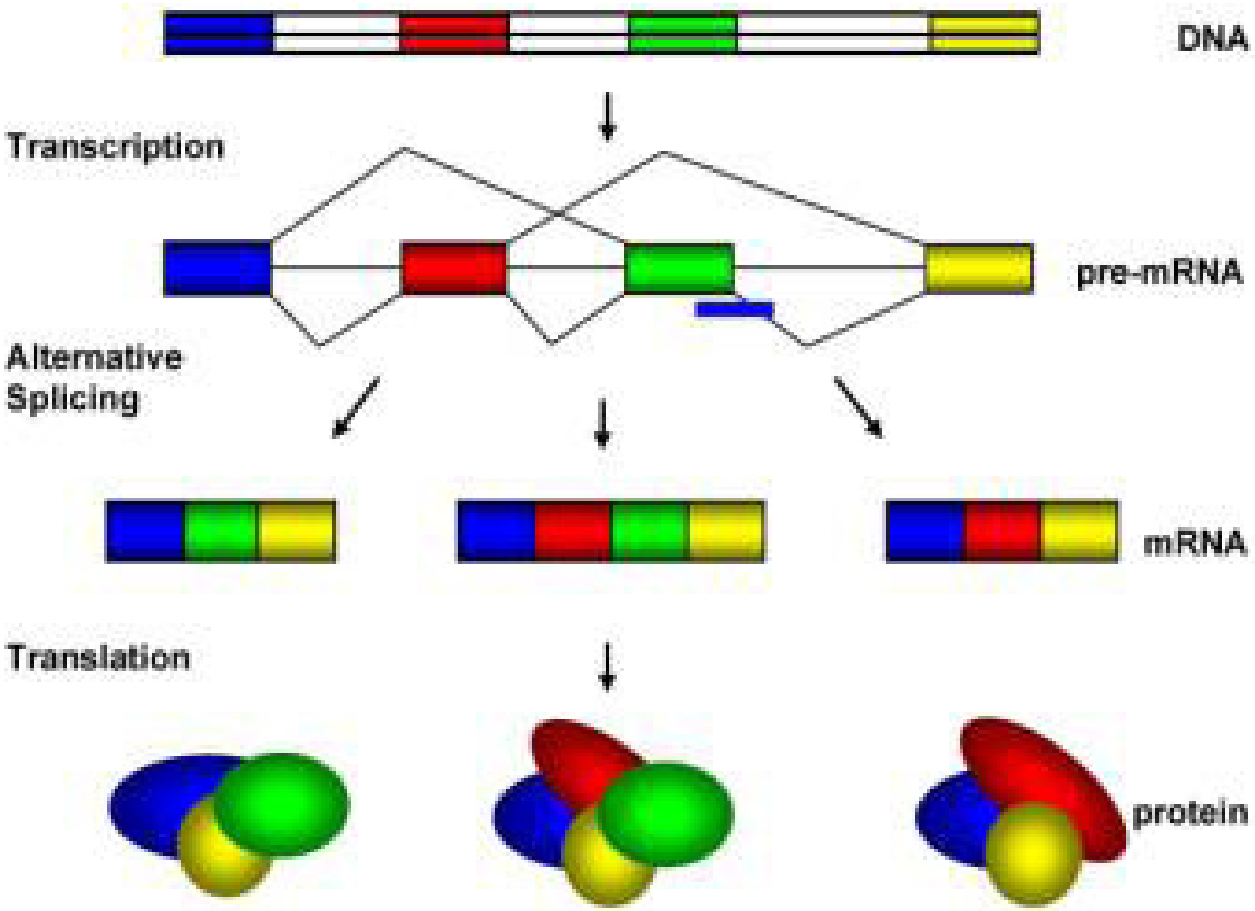
Modern Biology and Challenges



DOE Joint Genome institute

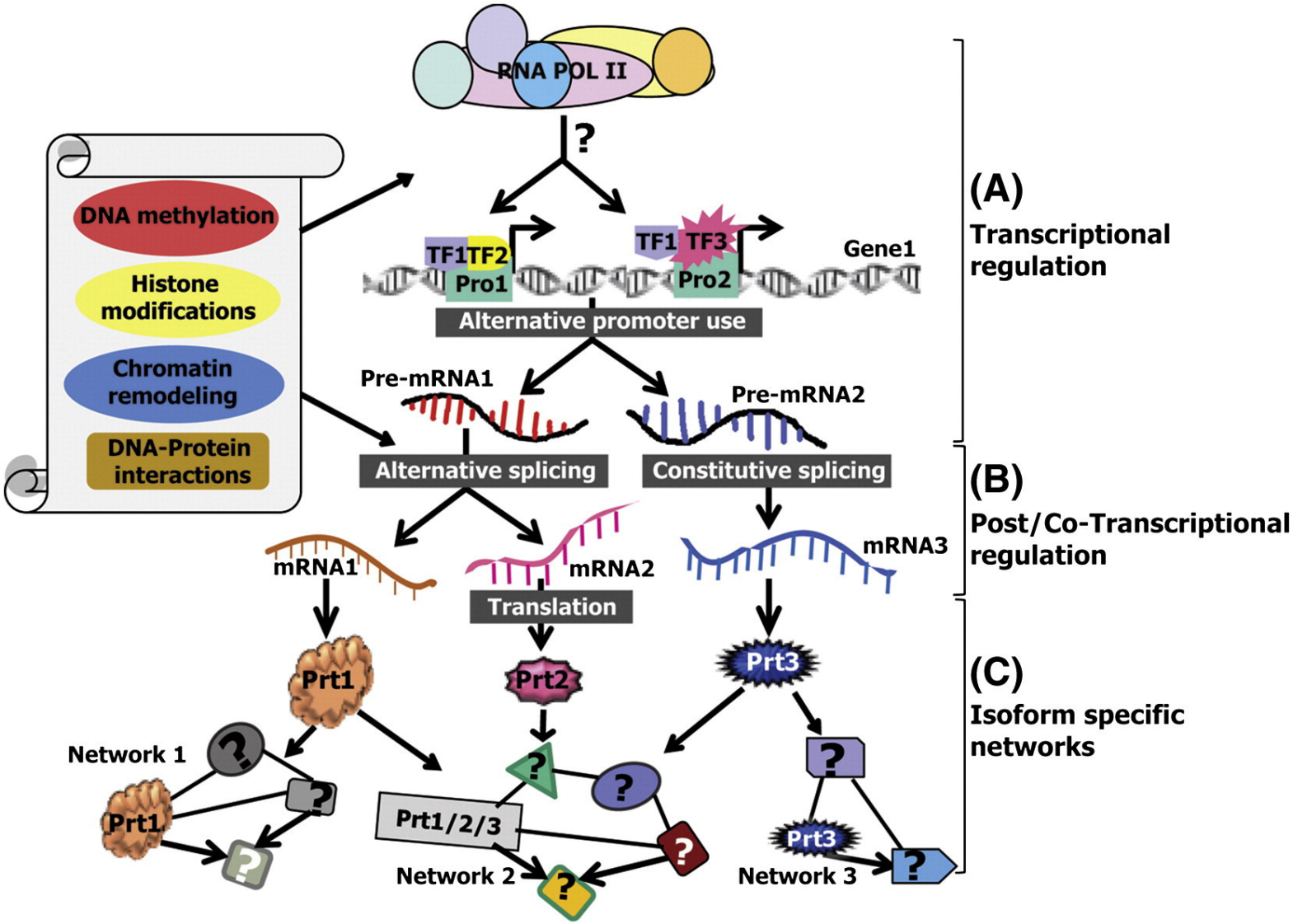
- biology is producing massive amount of data;
- sequencing one genome now costs about 1000$ (vs 0.1 billion $ in 2001), and produces about a few gigabytes of data;
- prediction from DNA data.

Alternative Splicing: 1 Gene = Many Proteins



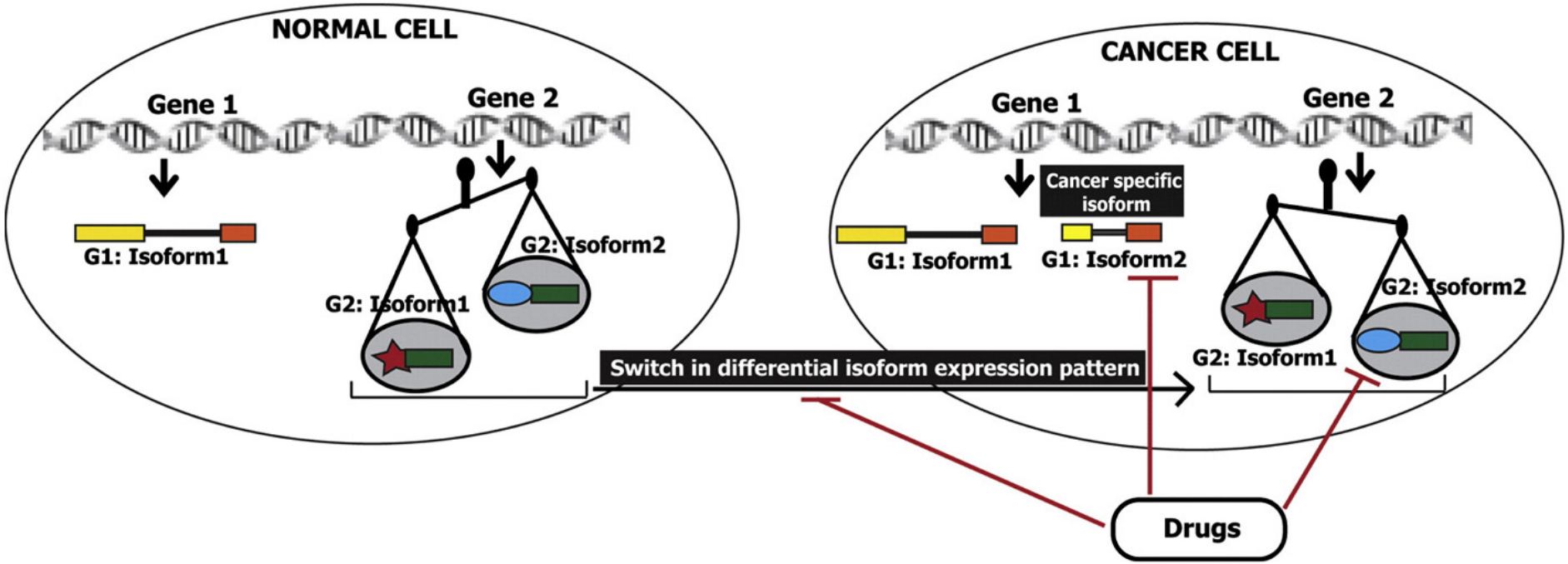
In human, 28k genes give 120k known transcripts (*Pal et al., 2012*)

Importance of Alternative Splicing



(*Pal et al., 2012*)

Opportunities for Drug Developments...

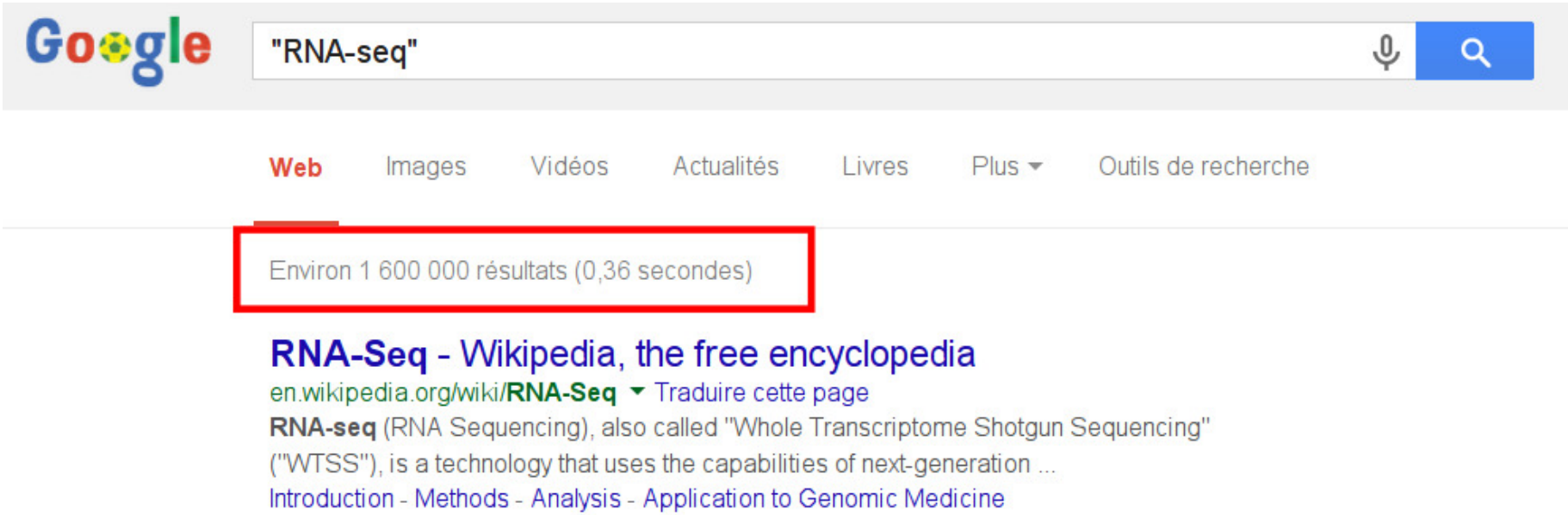


(*Pal et al., 2012*)

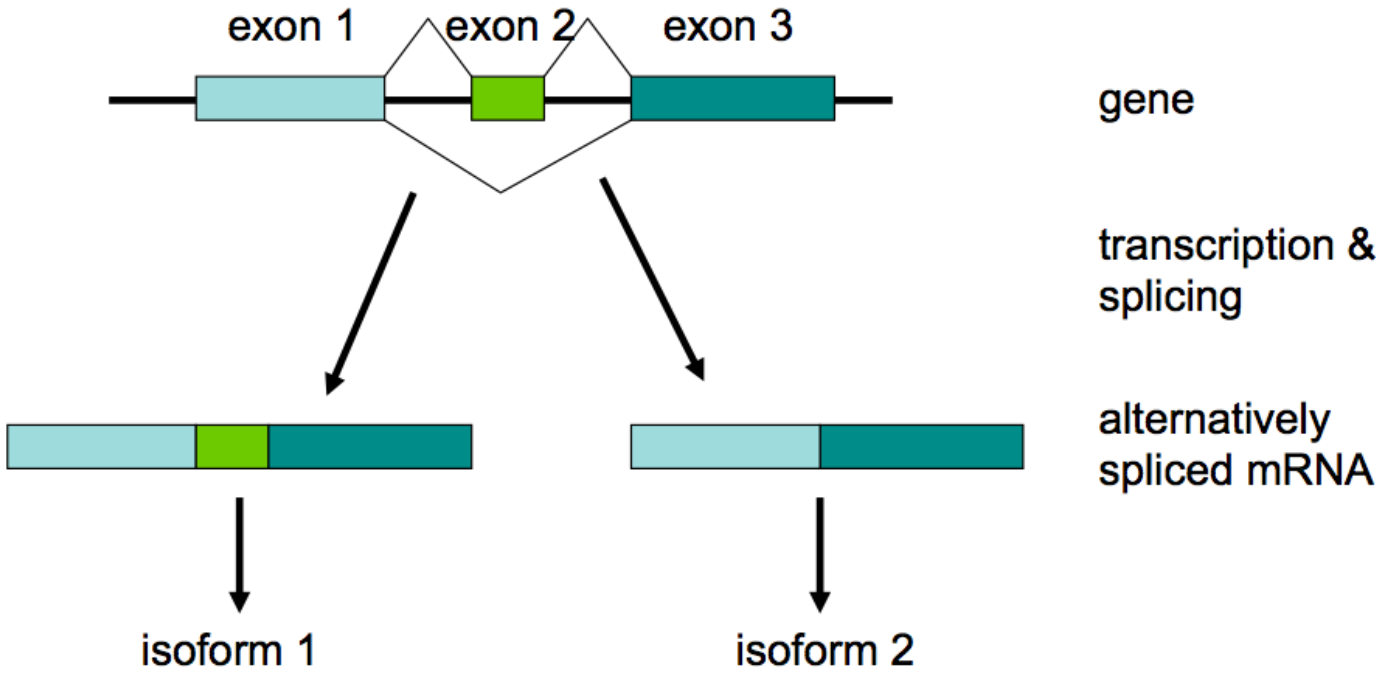
RNA-Seq or Next-Generation Sequencing

What is RNA-Seq?

- RNA-Seq measures abundance of RNA;



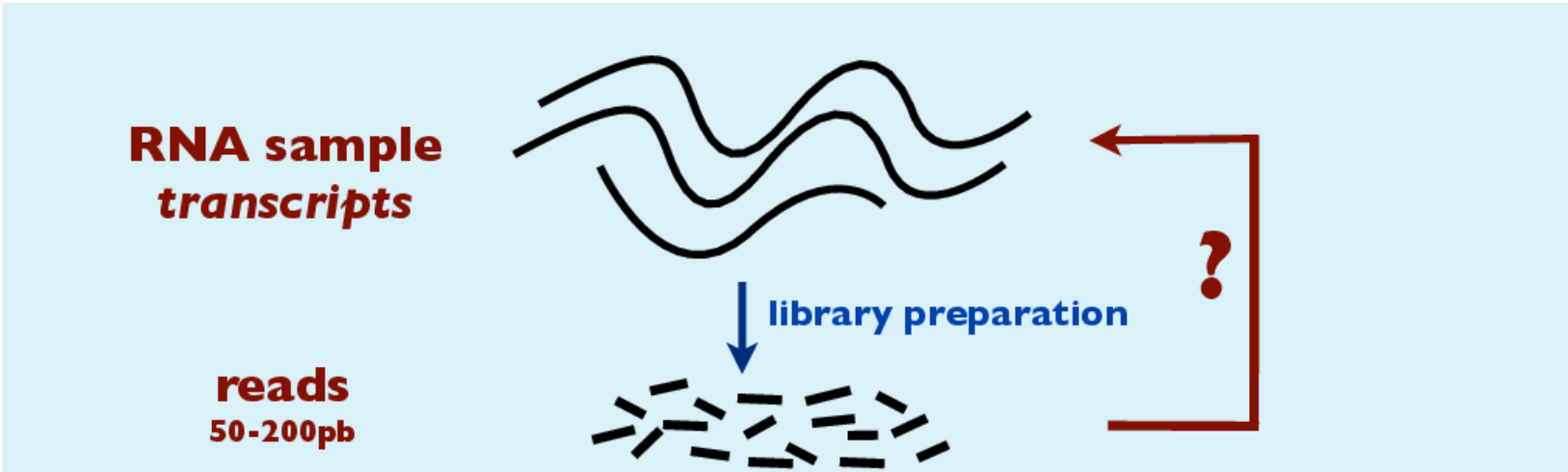
The Isoform Identification and Quantification Problem



Given a biological sample can we:

1. identify the isoform(s) of each gene present in the sample?
2. quantify their abundance?

From RNA-Seq Reads to Isoforms



Transcripts Quantification using annotations

- RQuant (Bohnert et al. 2009)
- FluxCapacitor (Montgomery et al. 2010)
- IsoEM (Nicolae et al. 2011)
- eXpress (Roberts et al. 2013)

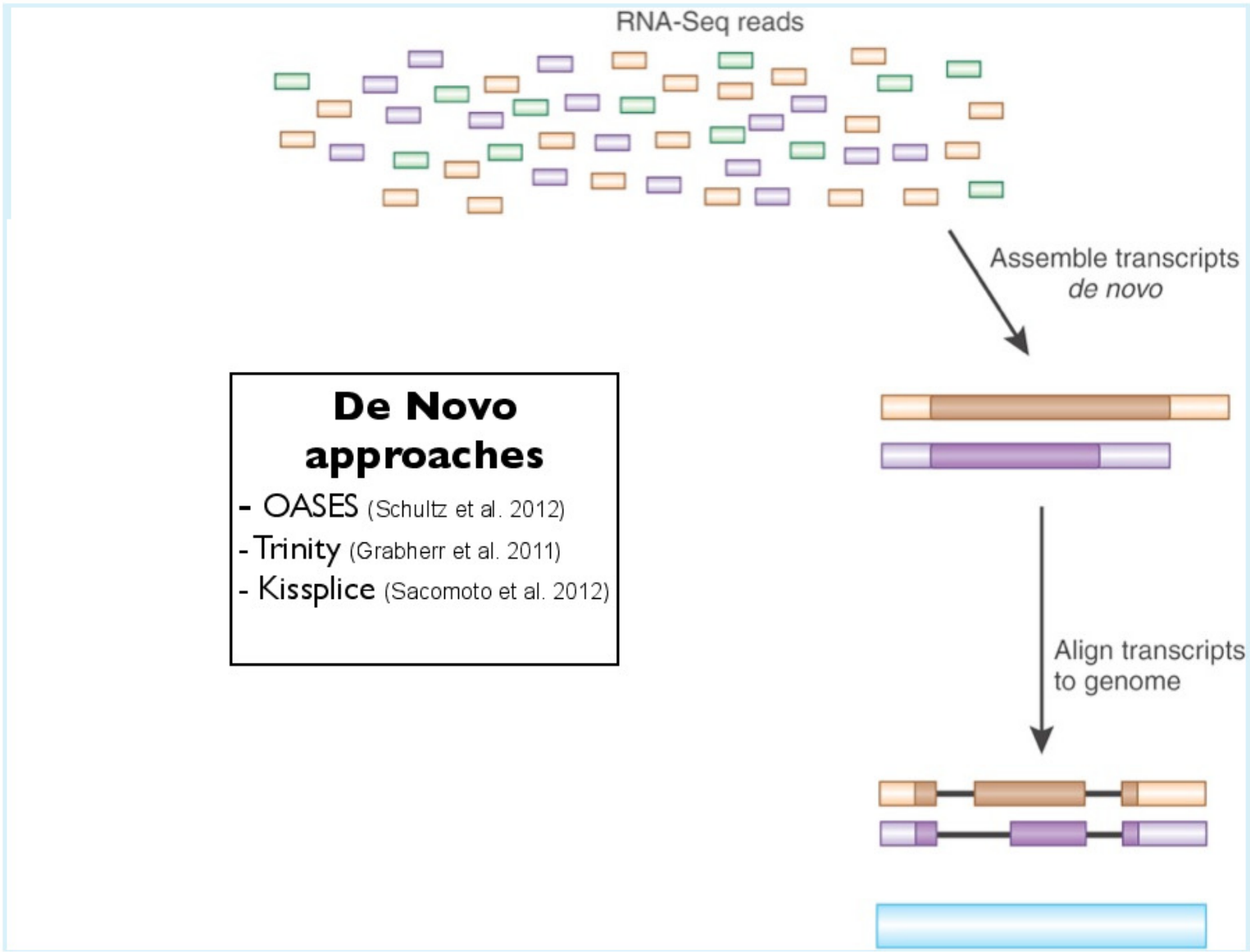
De Novo approaches

- Trinity (Grabherr et al. 2011)
- OASES (Schultz et al. 2012)
- Kissplice (Sacomoto et al. 2012)

Genome-based Transcripts Reconstruction

- Scripture (Guttman et al. 2010)
- Cufflinks (Trapnell et al. 2010)
- IsoLasso (Li et al. 2011a)
- NSMAP (Xia et al. 2011)
- SLIDE (Li et al. 2011b)
- iReckon (Mezlini et al. 2012)
- MiTie (Behr et al. 2013)
- **FlipFlop**

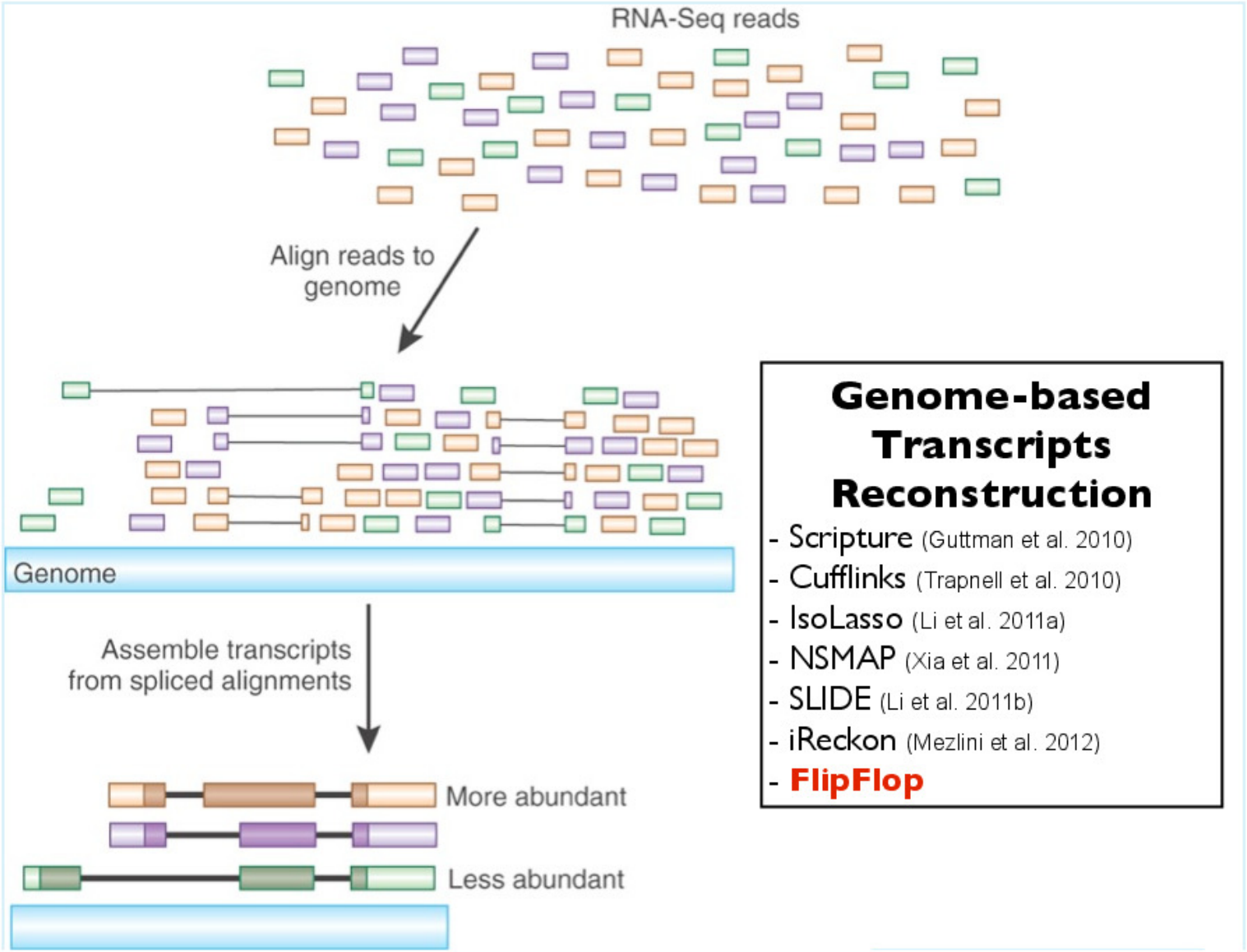
De Novo methods



De Novo approaches

- OASES (Schultz et al. 2012)
- Trinity (Grabherr et al. 2011)
- Kissplice (Sacomoto et al. 2012)

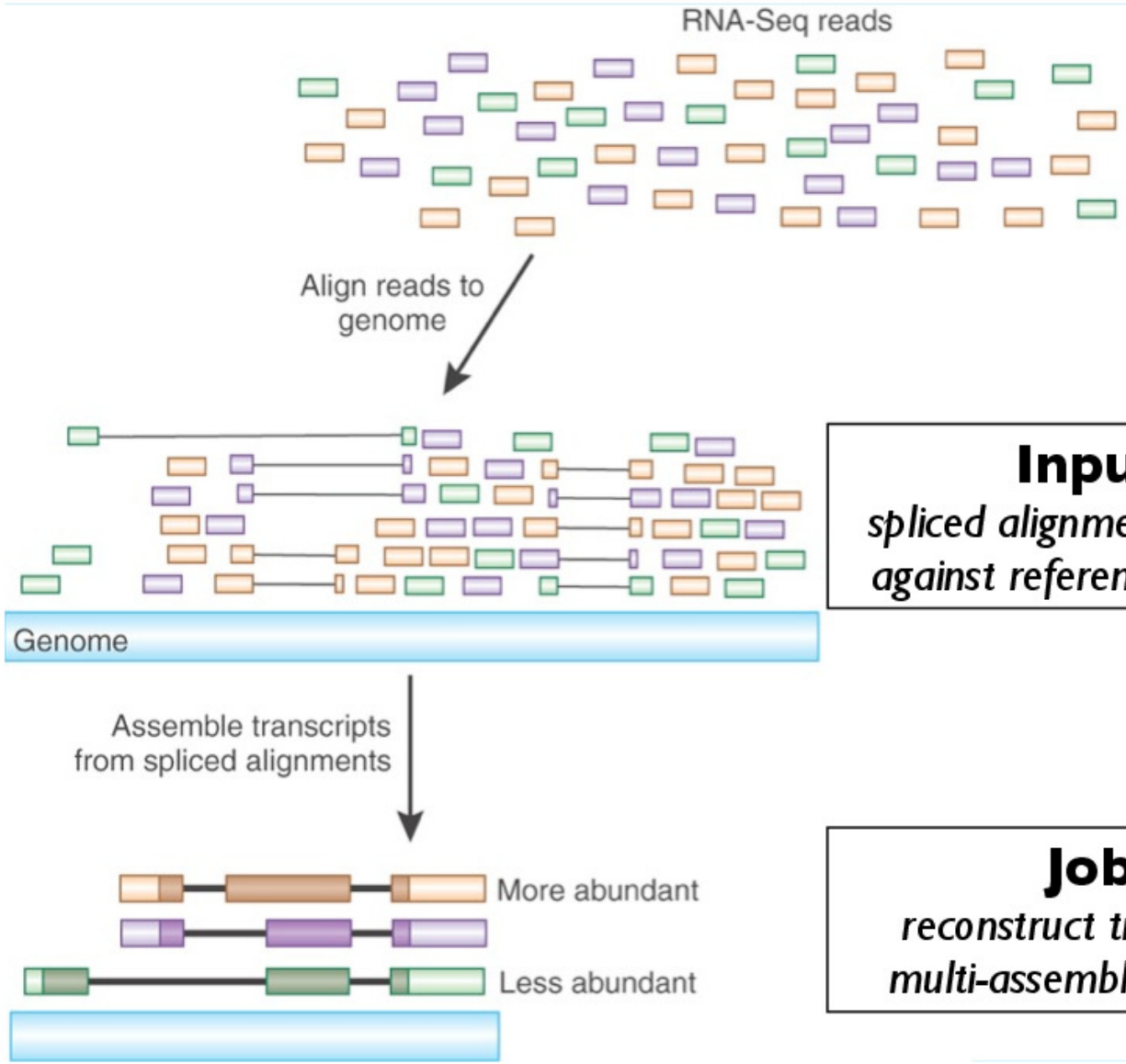
Genome-Based Methods



Genome-based Transcripts Reconstruction

- Scripture (Guttman et al. 2010)
- Cufflinks (Trapnell et al. 2010)
- IsoLasso (Li et al. 2011a)
- NSMAP (Xia et al. 2011)
- SLIDE (Li et al. 2011b)
- iReckon (Mezlini et al. 2012)
- **FlipFlop**

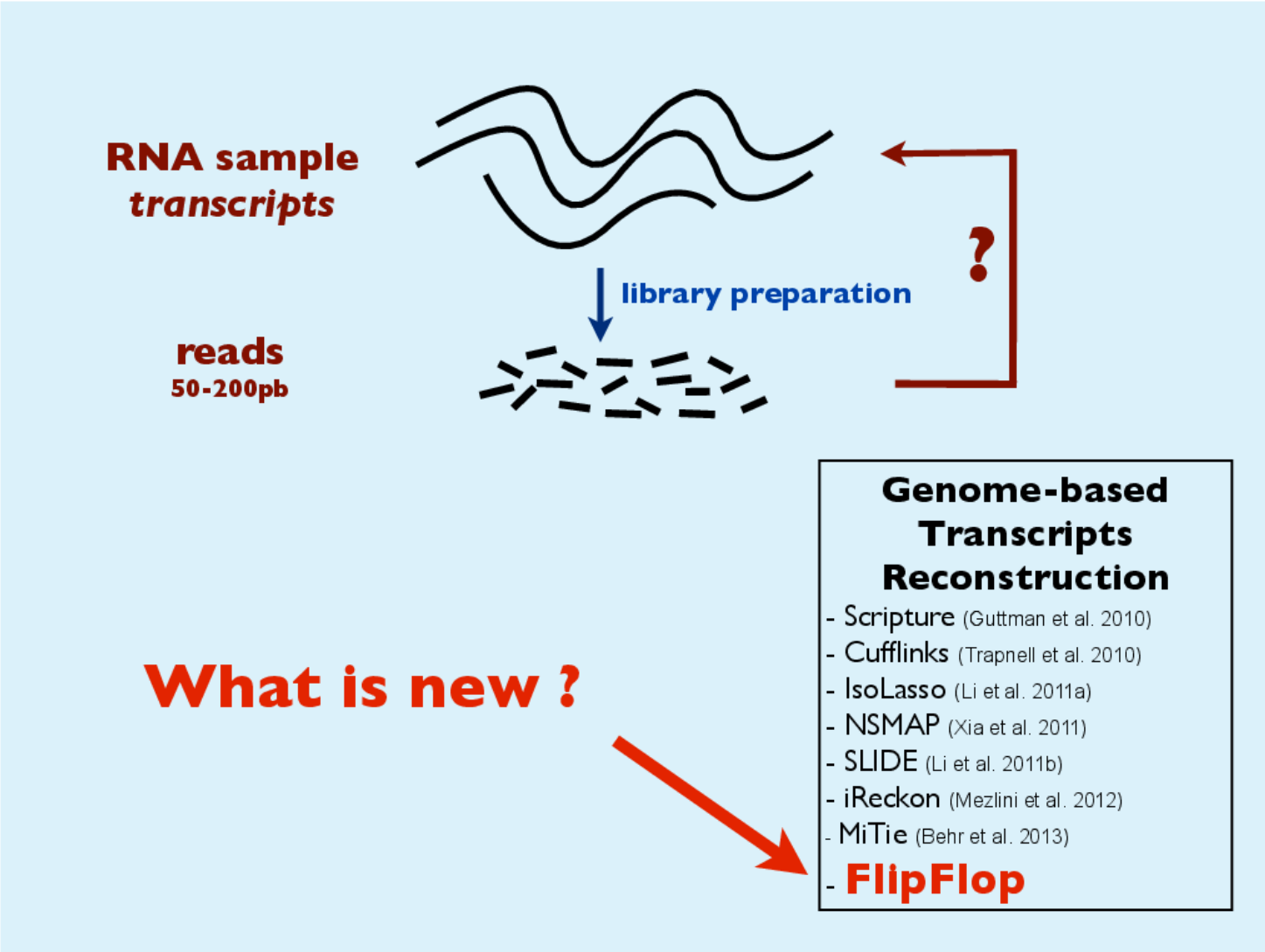
Genome-Based Isoforms Reconstruction



Input:
spliced alignment of reads against reference genome

Job:
reconstruct transcripts multi-assembly problem

Place in the literature



Contributions

- **NO NEED for FILTERING of candidate isoforms**
- **FASTER than existing methods that solve the same problem**

} ***flow method***

- **adapted to LONG READS**
- **R package**

Contributions

- NO NEED for FILTERING of candidate isoforms

- FASTER than existing methods that solve the same problem

- adapted to LONG READS } *particular splicing graph*

- R package

Contributions

- NO NEED for FILTERING of candidate isoforms

- FASTER than existing methods that solve the same problem

- adapted to long reads

- **R package**

Contributions

Home | Install | Help

Home » Bioconductor 2.13 » Software Packages » flipflop

flipflop

Fast lasso-based isoform prediction as a flow problem

Bioconductor version: Release (2.13)

Flipflop discovers which isoforms of a gene are expressed in a given sample together with their abundances, based on RNA-Seq read data.

Author: Elsa Bernard, Laurent Jacob, Julien Mairal and Jean-Philippe Vert

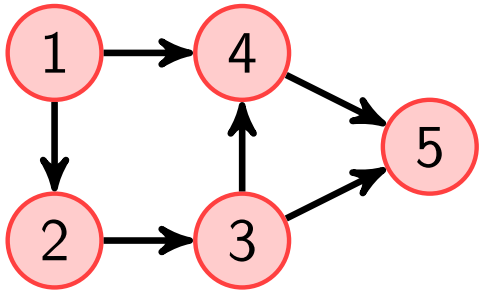
Maintainer: Elsa Bernard <elsa.bernard at mines-paristech.fr>

To install this package, start R and enter:

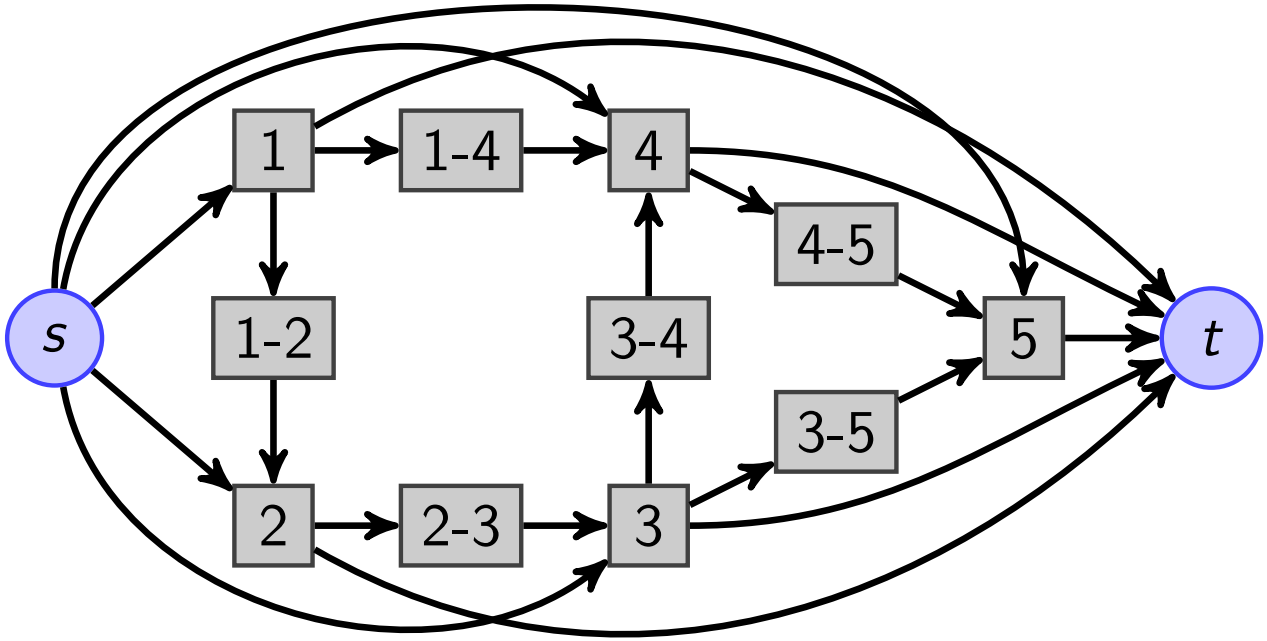
```
source("http://bioconductor.org/biocLite.R")
biocLite("flipflop")
```

Isoforms are Paths in a Graph

- Splicing graph for a gene with 5 exons:

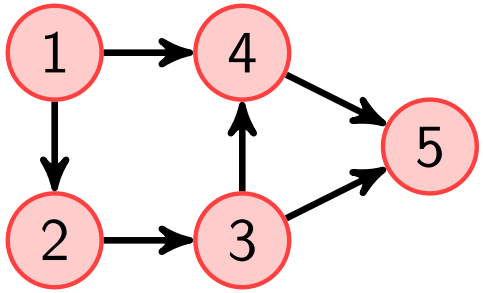


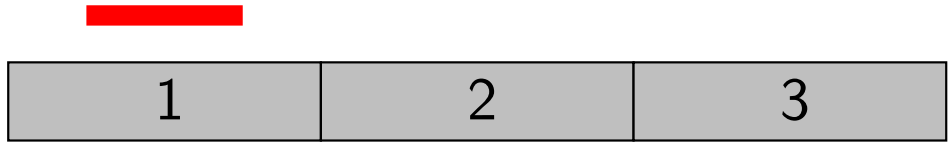
- FlipFlop graph: **1 type of read $\leftrightarrow$ 1 node**



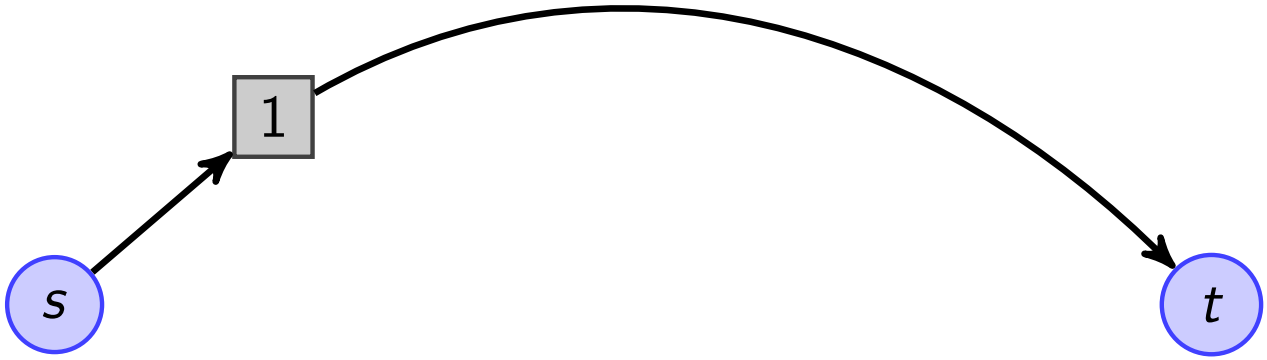
Graph adapted to long reads

- Splicing graph for a gene with 5 exons:



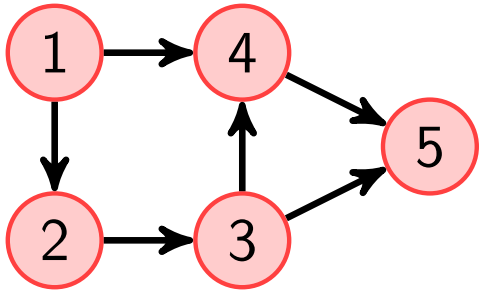


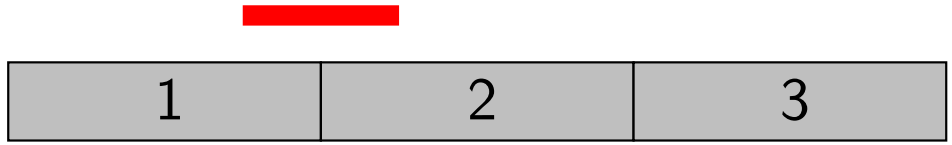
- FlipFlop graph:



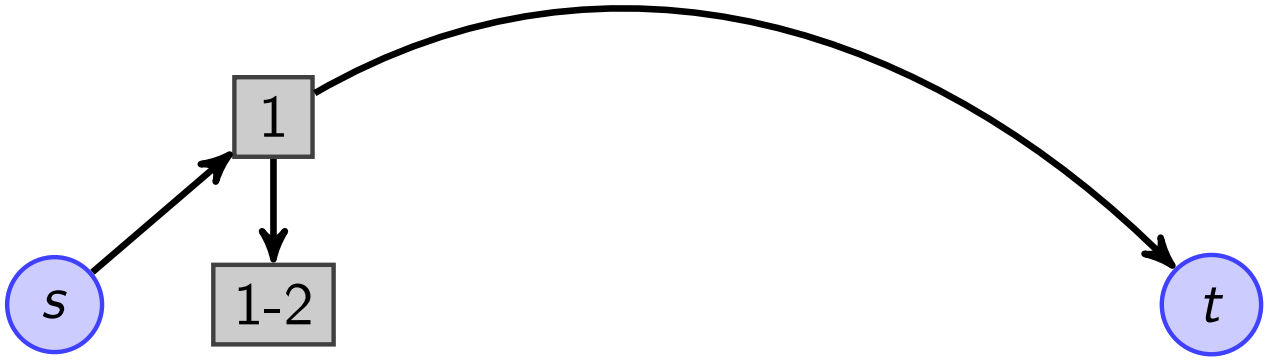
Graph adapted to long reads

- Splicing graph for a gene with 5 exons:



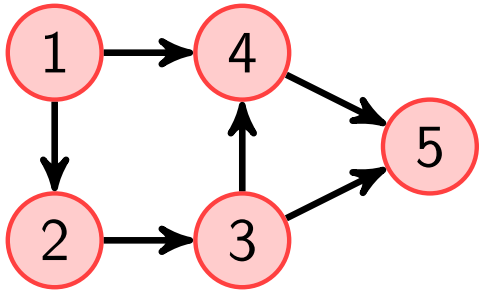


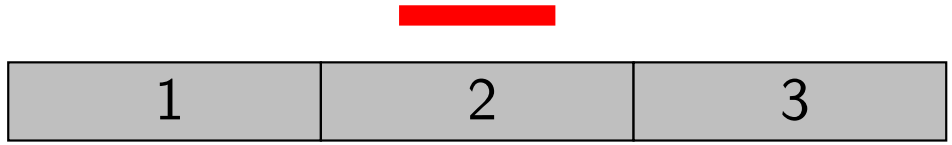
- FlipFlop graph:



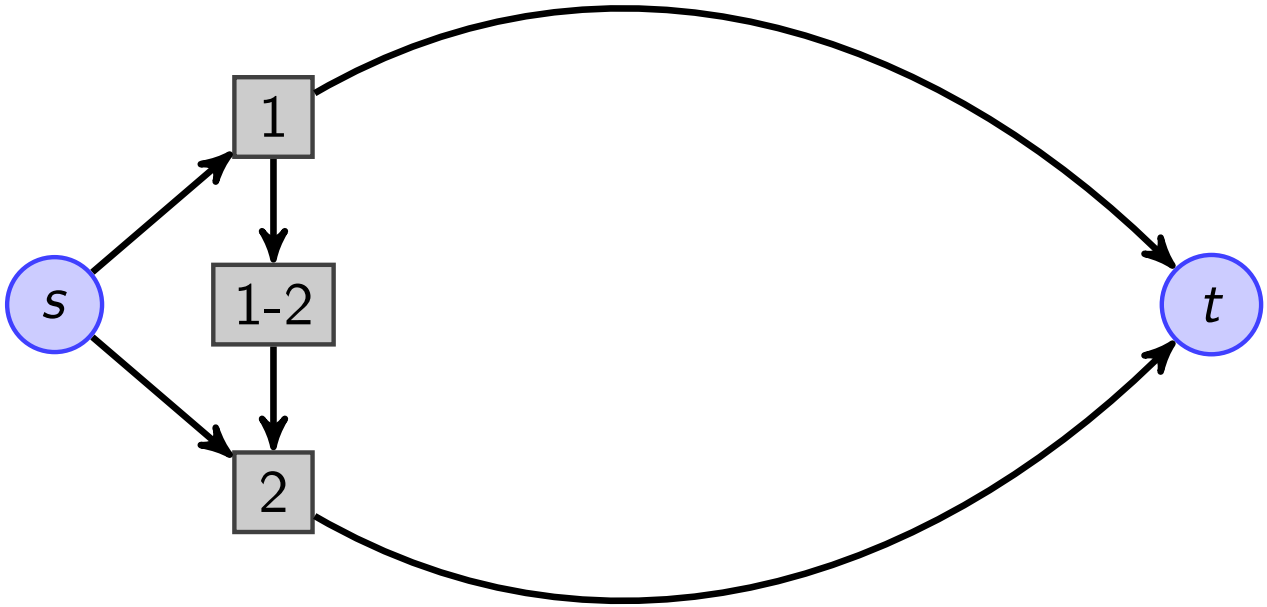
Graph adapted to long reads

- Splicing graph for a gene with 5 exons:



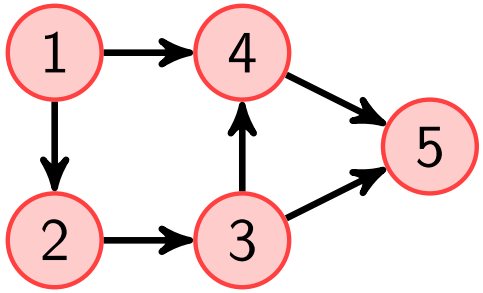


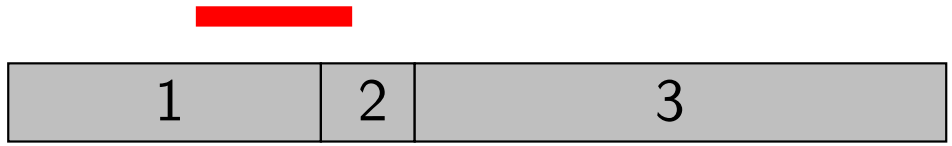
- FlipFlop graph:



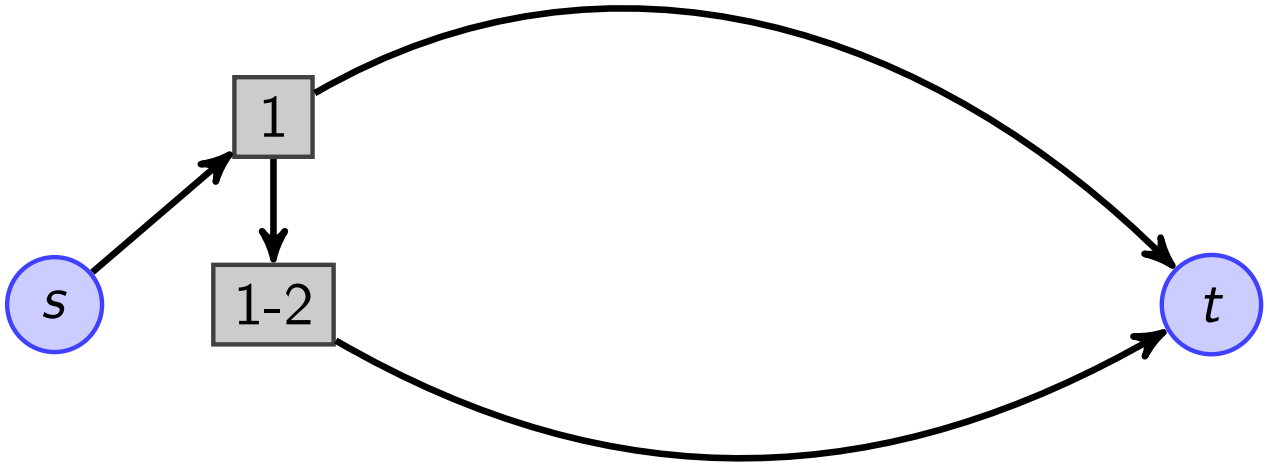
Graph adapted to long reads

- Splicing graph for a gene with 5 exons:



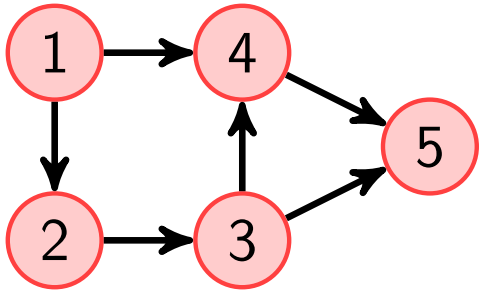


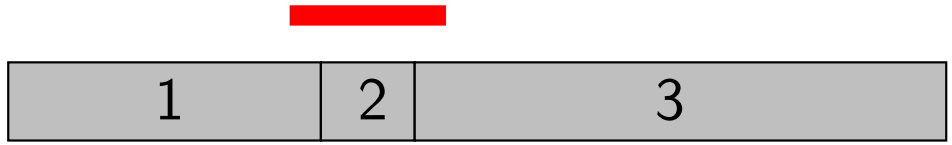
- FlipFlop graph:



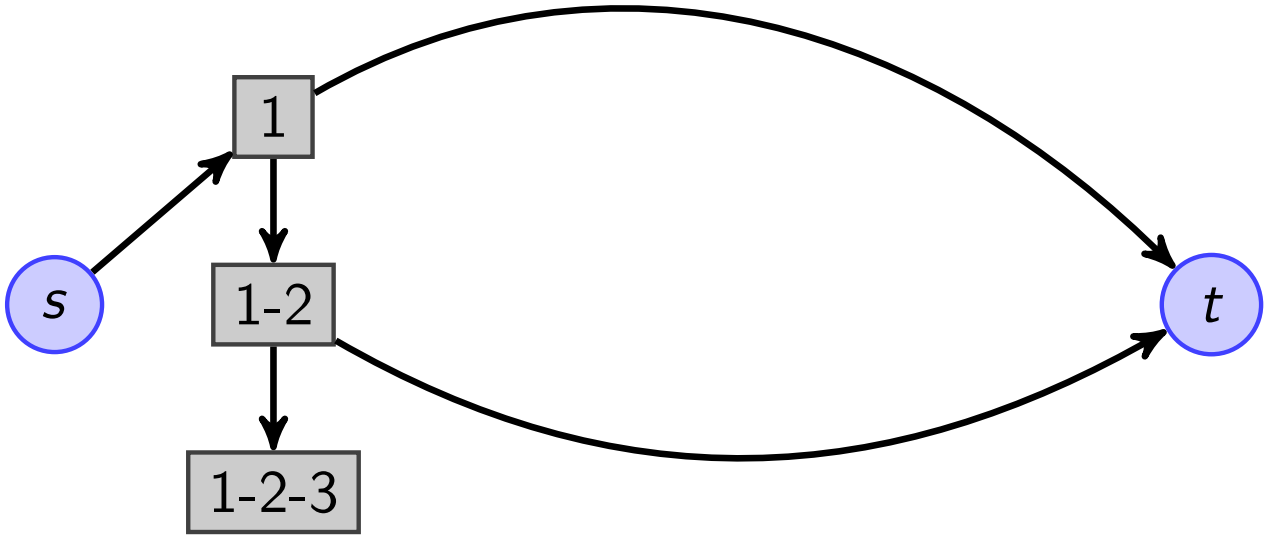
Graph adapted to long reads

- Splicing graph for a gene with 5 exons:



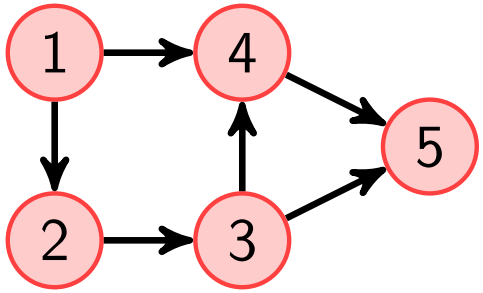


- FlipFlop graph:

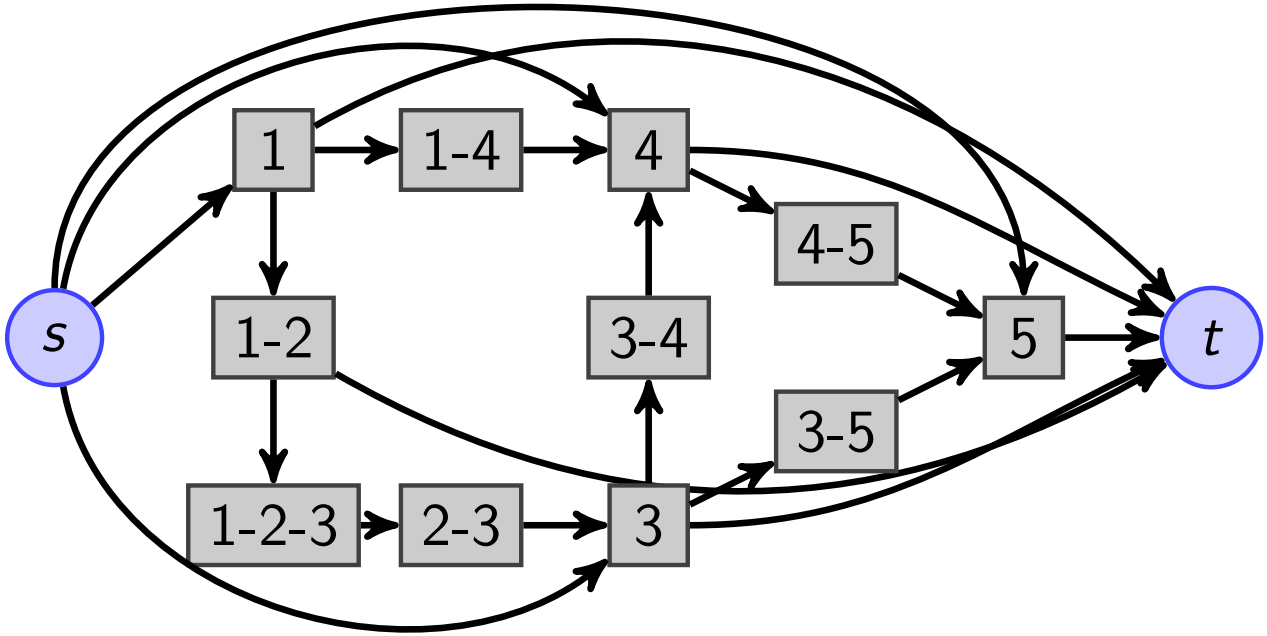


Graph adapted to long reads

- Splicing graph for a gene with 5 exons:

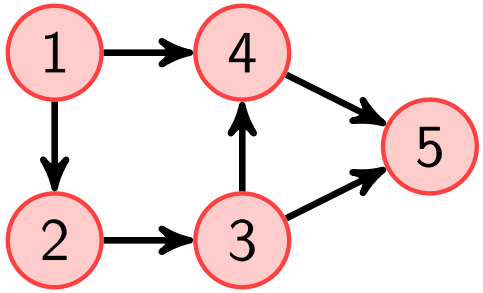


- FlipFlop graph:

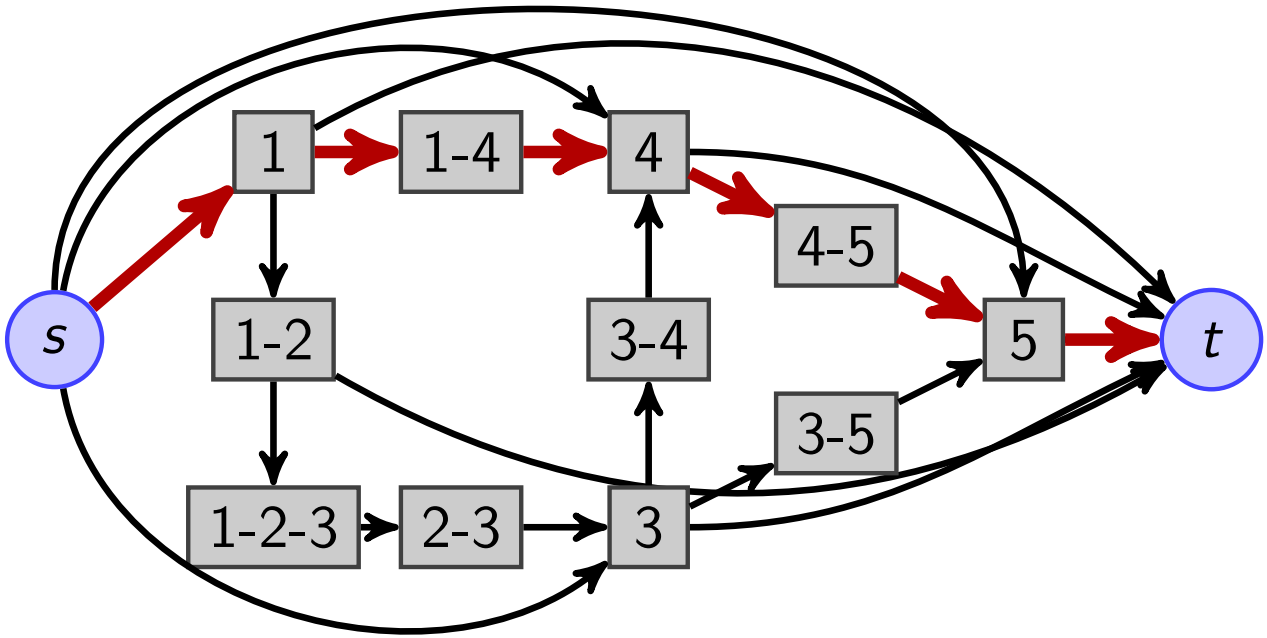


Graph adapted to long reads

- Splicing graph for a gene with 5 exons:

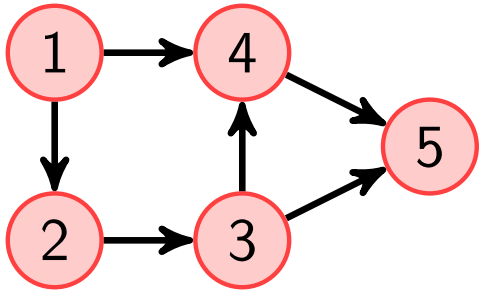


- FlipFlop graph: **one path with abundance β_1**

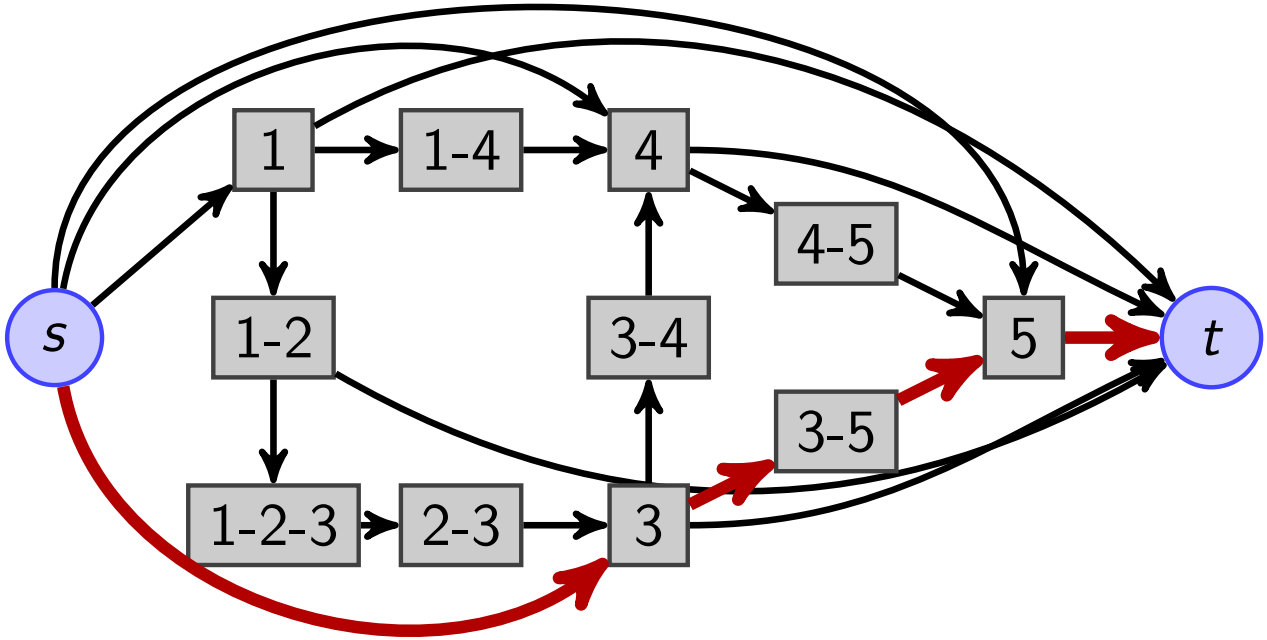


Graph adapted to long reads

- Splicing graph for a gene with 5 exons:



- FlipFlop graph: **another path with abundance β_2 ...**



Select a Small Number of Paths?

$$n \textbf{ exons } \rightarrow \sim 2^n \textbf{ paths/candidate isoforms}$$

feature selection problem with $\sim$ 1000 candidates for 10 exons and $\sim$ 1000000 for 20 exons

Minimal path cover

- Cufflinks

Regularization approach

- IsoLasso, NSMAP, SLIDE, iReckon, MiTie, **FlipFlop**

Select a Small Number of Paths?

Cufflinks strategy

A two-step approach

1. find a set of *minimal paths* to explain read positions (independent from read counts)
2. estimate isoform abundances using read counts

Select a small number of paths?

Regularization approach

1. Suppose there are c **candidate isoforms** (c large)
2. Let β the unknown c-dimensional **vector of abundance**

Select a small number of paths?

Regularization approach

1. Suppose there are c **candidate isoforms** (c large)
2. Let β the unknown c-dimensional **vector of abundance**
3. Let $\mathcal{L}(\beta)$ quantify whether β explains the observed read counts
   - e.g., Poisson negative log-likelihood:

$$\mathcal{L}(\beta) = \sum_{\text{node } u} -\log p(X_u) \quad \text{with} \quad X_u \sim \mathcal{P}(\delta_u) \quad \text{and} \quad \delta_u \propto l_u \sum_{\text{path } p \ni u} \beta_p$$

Select a small number of paths?

Regularization approach

1. Suppose there are c **candidate isoforms** (c large)
2. Let β the unknown c-dimensional **vector of abundance**
3. Let $\mathcal{L}(\beta)$ quantify whether β explains the observed read counts
   - e.g., Poisson negative log-likelihood:

$$\mathcal{L}(\beta) = \sum_{\text{node } u} -\log p(X_u) \quad \text{with} \quad X_u \sim \mathcal{P}(\delta_u) \quad \text{and} \quad \delta_u \propto l_u \sum_{\text{path } p \ni u} \beta_p$$

4. Regularization-based approaches try to solve:

$$\min_{\beta \in \mathbb{R}_+^c} \mathcal{L}(\beta) \text{ such that } \beta \text{ is sparse}$$

Isoform Deconvolution with the ℓ_1-norm

ℓ_1-regularization

Estimate β **sparse** by solving:

$$\min_{\beta \in \mathbb{R}^c_+} \mathcal{L}(\beta) + \lambda \|\beta\|_1 \ ,$$

with $\mathcal{L}$ a convex loss function.

Computationally challenging:

$\rightarrow$ IsoLasso: strong filtering

$\rightarrow$ NSMAP, SLIDE: number of exons cut-off

FlipFlop: Fast Lasso-based Isoform Prediction as a FLOw Problem

$\rightarrow$ no filtering

$\rightarrow$ no exons restrictions

Fast Isoform Deconvolution with the lasso

Theoretical (practical) result

The isoform deconvolution problem

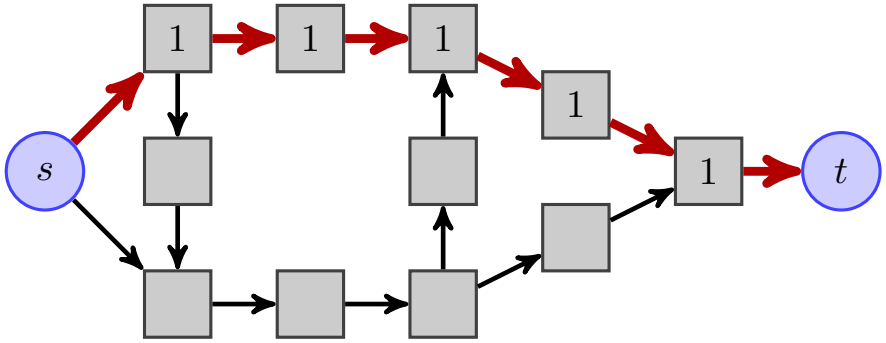
$$\min_{\beta \in \mathbb{R}_+^c} \mathcal{L}(\beta) + \lambda \|\beta\|_1 \ ,$$

can be solved in **polynomial time** with the number of nodes of the splicing graph.

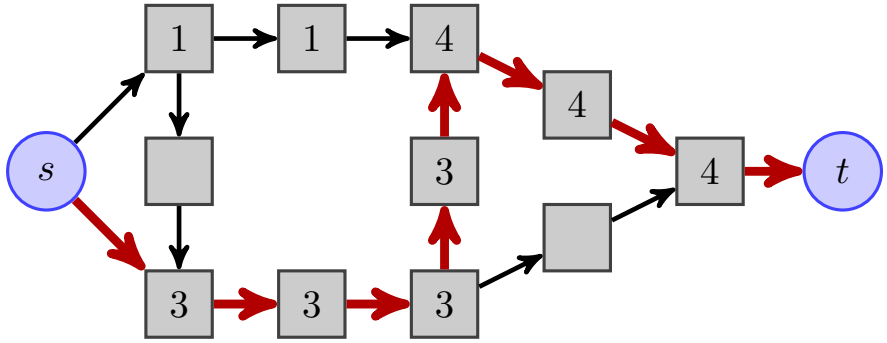
Ideas:

1. the sum of isoform abundances correspond to a **flow** on the graph
2. reformulation as a **convex cost flow problem** (Mairal and Yu, 2012)
3. recover isoforms by flow decomposition algorithm

Combinations of isoforms are flows



(c) Reads at every node corresponding to one isoform.



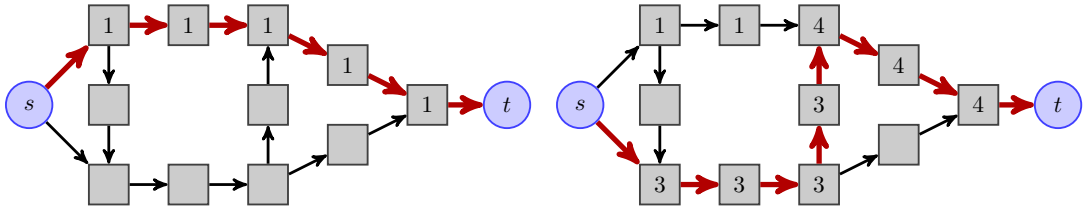
(d) Reads at every node after adding another isoform.

- **Linear combinations of isoforms** $\Rightarrow$ **Flow value on every edges**
- **Flow value on every edges** $\Rightarrow$ **Paths with given value/abundance**

Flow Decomposition
(linear time algorithm)

Flux Capacitor. 2008. A Novel Min-Cost Flow Method for Estimating Transcript Expression with RNA-Seq. RECOMB-2013.

Equivalent flow problem (simpler!)



(e) Reads at every node corresponding to one isoform. (f) Reads at every node after adding another isoform.

- For each edge sum abundances of isoforms that include the edge :

$$f_{uv} = \sum_{\text{path } p \ni (u,v)} \beta_p \quad \text{is a flow}$$

- Moreover

$$\|\beta\|_1 = \sum_{\textbf{path } p} \beta_p = f_t$$

- Therefore

$$\min_{\beta \in \mathbb{R}_+^c} \mathcal{L}(\beta) + \lambda \|\beta\|_1 \quad \text{is equivalent to} \quad \min_{\mathbf{f} \text{ flow}} \tilde{\mathcal{L}}(\mathbf{f}) + \lambda \mathbf{f_t}$$

Technical details

Poisson Loss (with binary matrix $\mathbf{U}$):

$$\mathcal{L}(\mathbf{U}^T\beta) = \sum_{u \in V} \left[Nl_u(\mathbf{U}^T\beta)_u - \mathbf{y}_u \log(Nl_u(\mathbf{U}^T\beta)_u) \right]$$

Flow Decomposition:

$$f_{uv} = \sum_{p \in \mathcal{P}'} \beta_p \mathbf{1}_{\{(u,v) \in p\}}$$

$$\Rightarrow \quad f_v = \sum_{u \in V'} f_{uv} = (\mathbf{U}^T\beta)_v$$

Convex Cost Flow:

$$\min_{f \text{flow}} \quad \sum_{u \in V} [Nl_u f_u - \mathbf{y}_u \log(f_u)] + \lambda f_t$$

Solved using ε-relaxation method (Bertsekas 1998).

Summary

Isoform Detection=Path Selection Problem

$\sim 2^n$ variables (all paths in the splicing graph)

$\Updownarrow$

Equivalent Network Flow Problem

$\sim \frac{n^2}{2}$ variables (all exons and exon-exon junctions in the splicing graph)

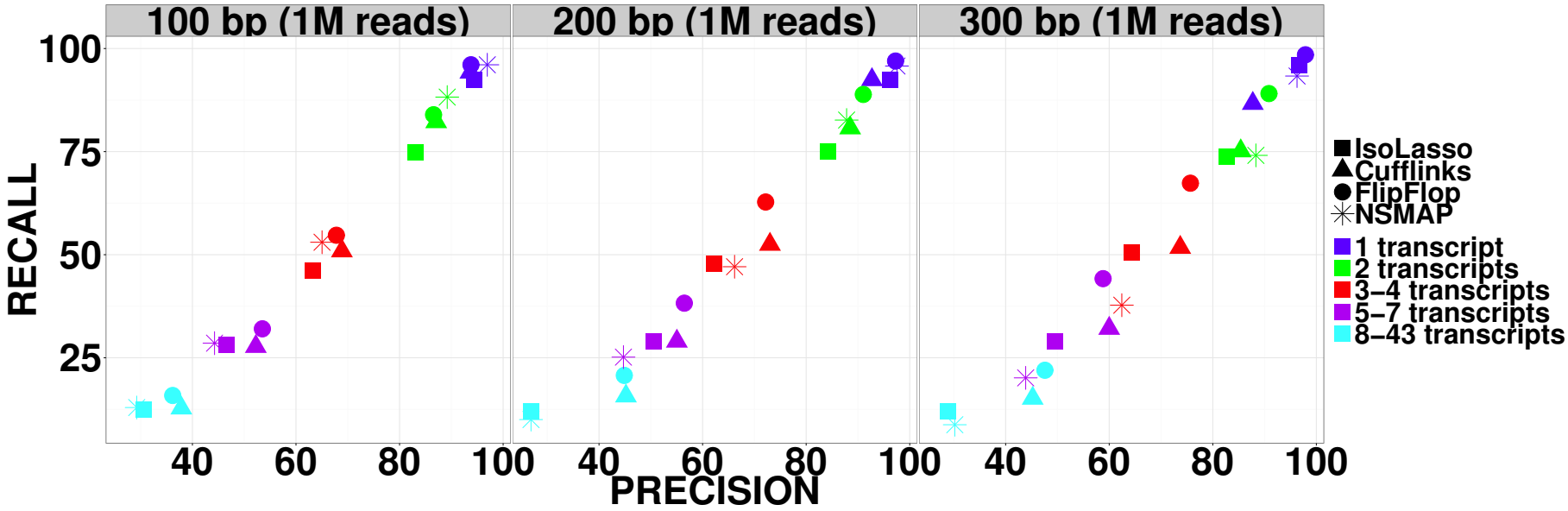
$\downarrow$

Network Flow Algorithms

Efficient Algorithms ! Polynomial Time.

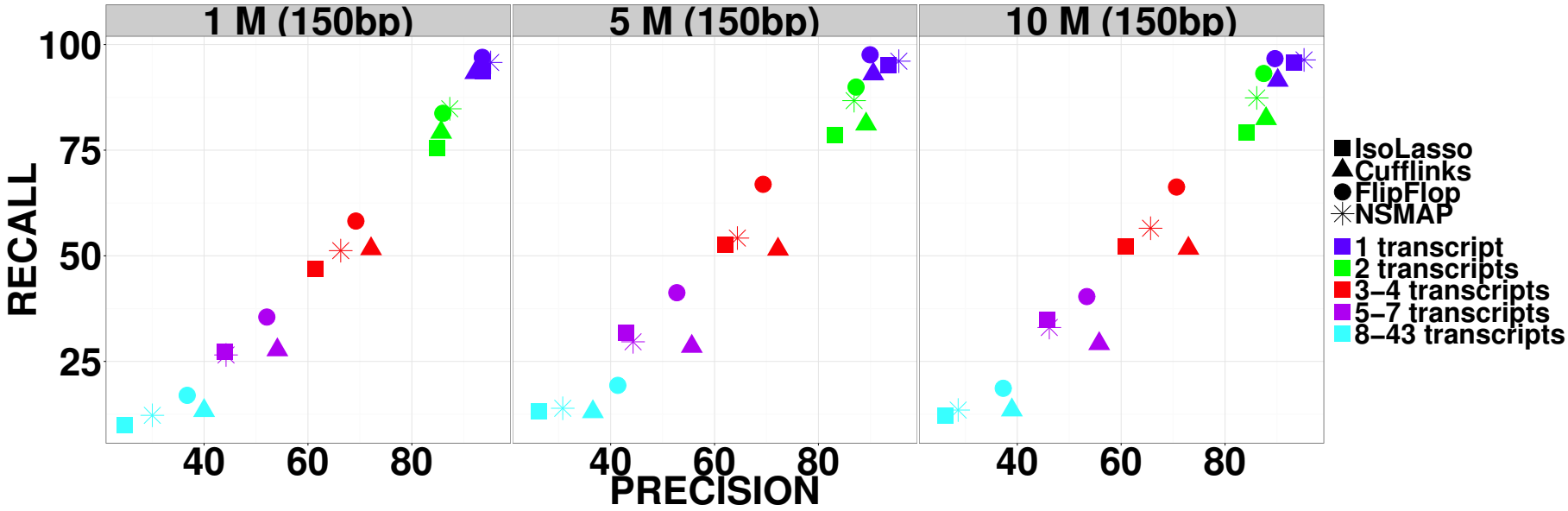
Performance increases with read length

- Human Simulation: hg19, 1137 genes on chr1, 1million reads by transcript levels.
- Simulator: `http://alumni.cs.ucr.edu/~liw/rnaseqreadsimulator.html`



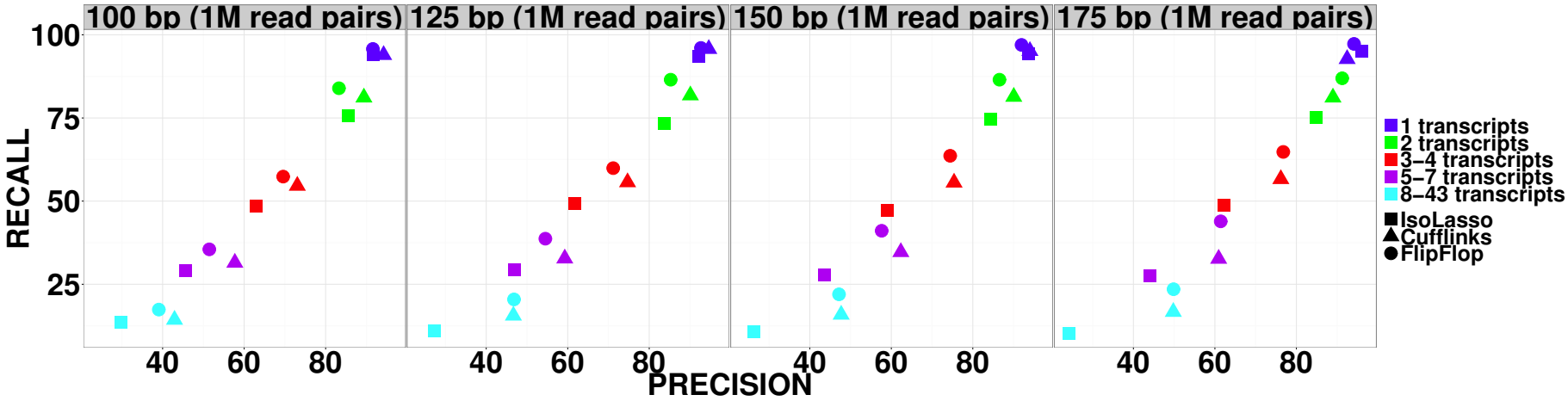
Performance increases with coverage

- Human Simulation: hg19, 1137 genes on chr1, 1million reads by transcript levels.
- Simulator: `http://alumni.cs.ucr.edu/~liw/rnaseqreadsimulator.html`



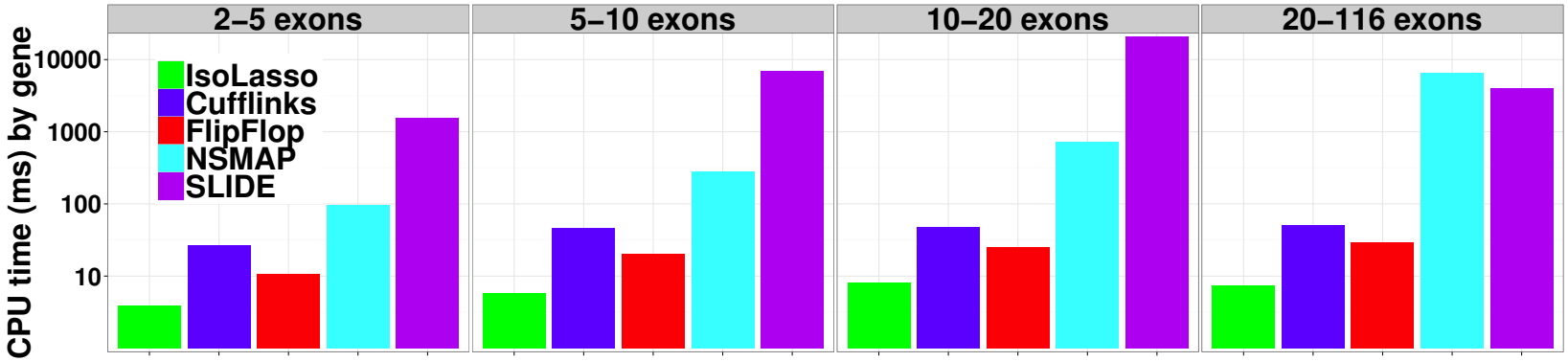
Extension to paired-end reads

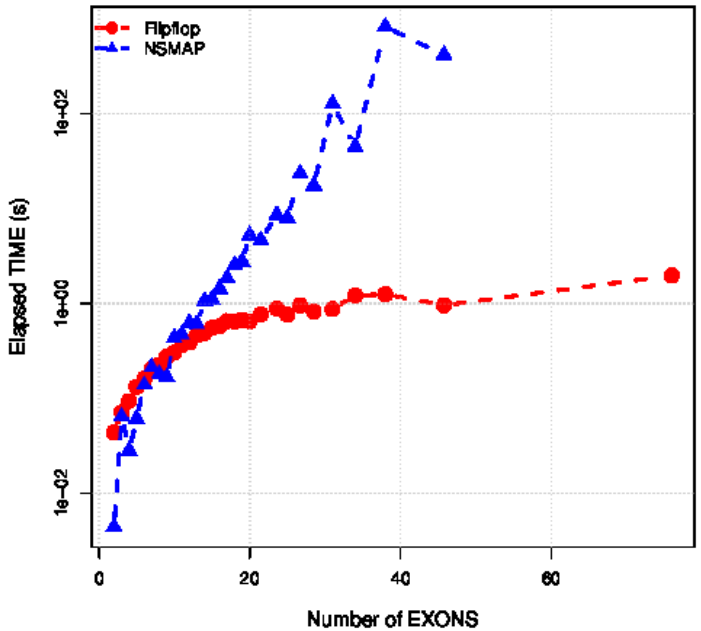
- Human Simulation: hg19, 1137 genes on chr1, 1 million reads by transcript levels.
- Simulator: `http://alumni.cs.ucr.edu/~liw/rnaseqreadsimulator.html`



Speed Trial

- Human Simulation: hg19, 1137 genes on chr1, 1 million reads by transcript levels.
- Simulator: `http://alumni.cs.ucr.edu/~liw/rnaseqreadsimulator.html`



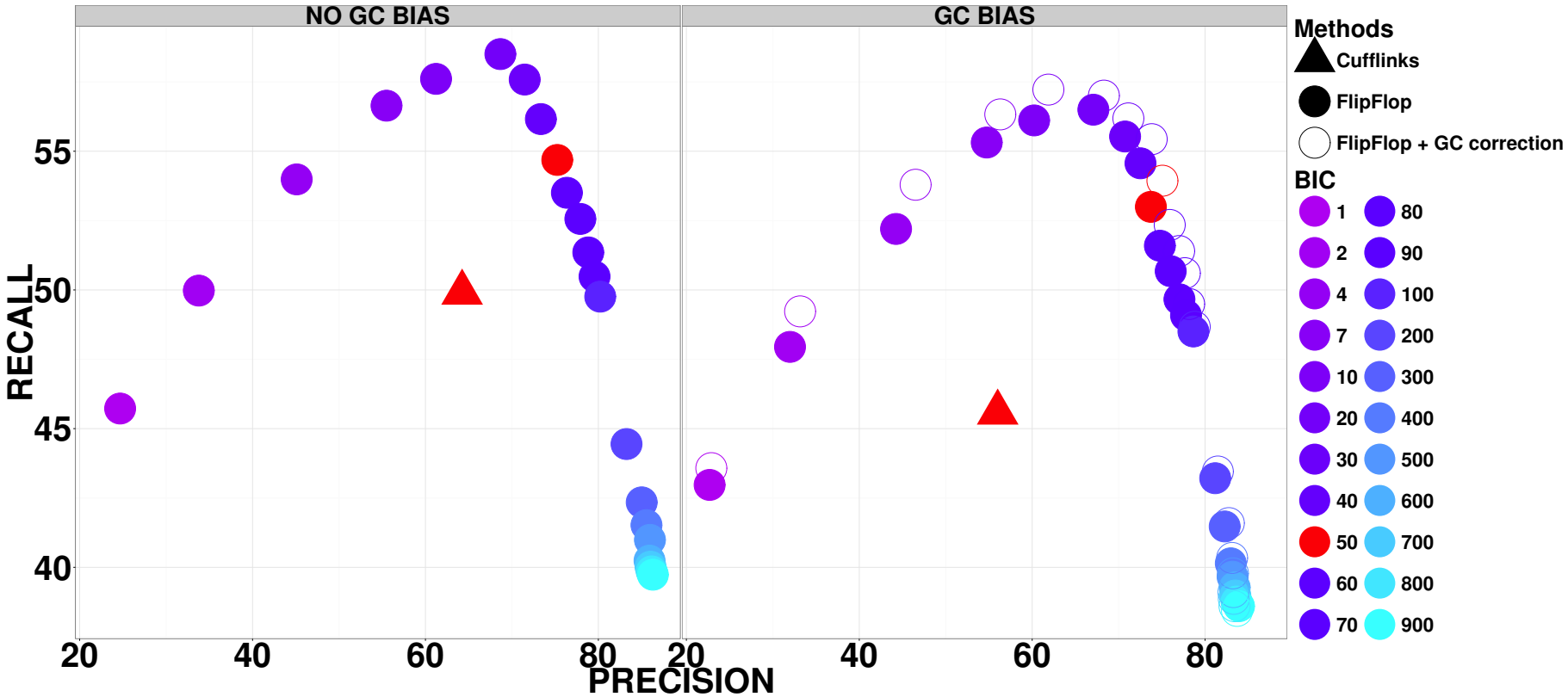


GC bias - Precision-Recall curve

- Human Simulation: hg19, chr1, 150bp single-end reads, 2 million, 4140 transcripts.

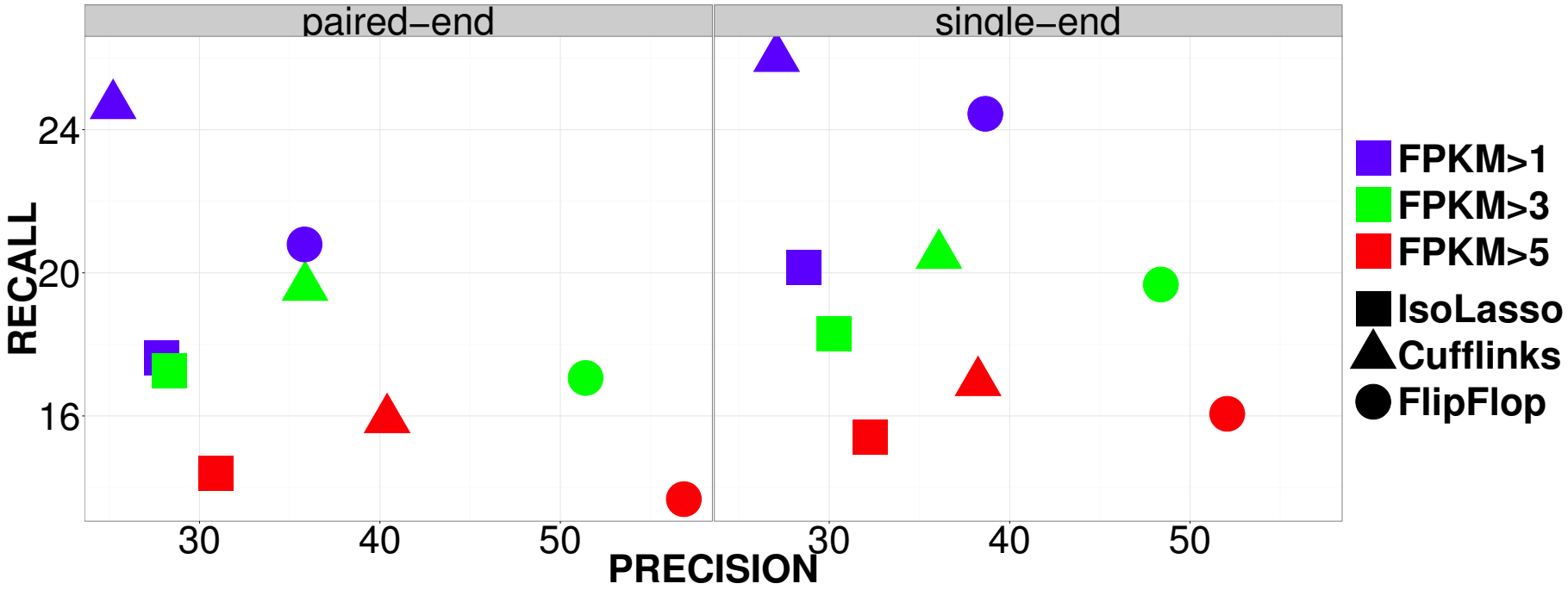
FluxSimulator, Griebel et al, 2012.

Model selection: set of solutions minimizing $\mathcal{L}(\beta) + \lambda\|\beta\|_1$ for different values of $\lambda \rightarrow$ BIC criteria



Real Data

- Human: 50 million 75bp paired-end reads.



Conclusion/Discussion

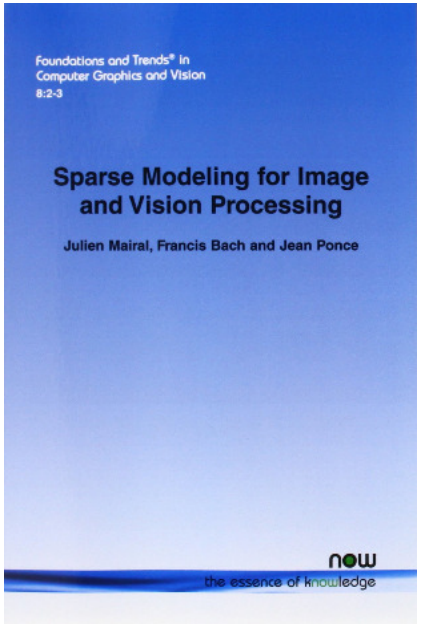
FlipFlop $\rightarrow$ transcripts reconstruction over an exponential number of candidates in polynomial time

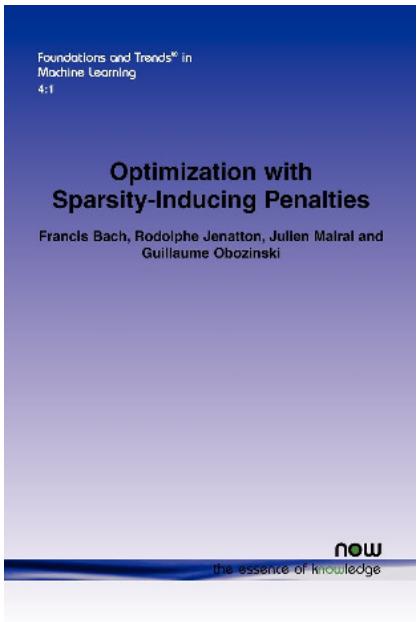
1. **Hard combinatorial ill-posed** prediction problem !
2. **Model Selection:** Cross Validation, Stability Selection?
3. **Multiple-samples:** on-going work with promising preliminary results.
4. **Differential Expression** testing at the isoform level ?

Conclusion/Discussion: get FlipFlop for free!

Advertisement: free monographs

J. Mairal, F. Bach and J. Ponce. *Sparse Modeling for Image and Vision Processing*. Foundations and Trends in Computer Graphics and Vision. 2014.





F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. *Optimization with sparsity-inducing penalties*. Foundations and Trends in Machine Learning, 4(1). 2012.

Advertisement SPAMS toolbox (open-source)

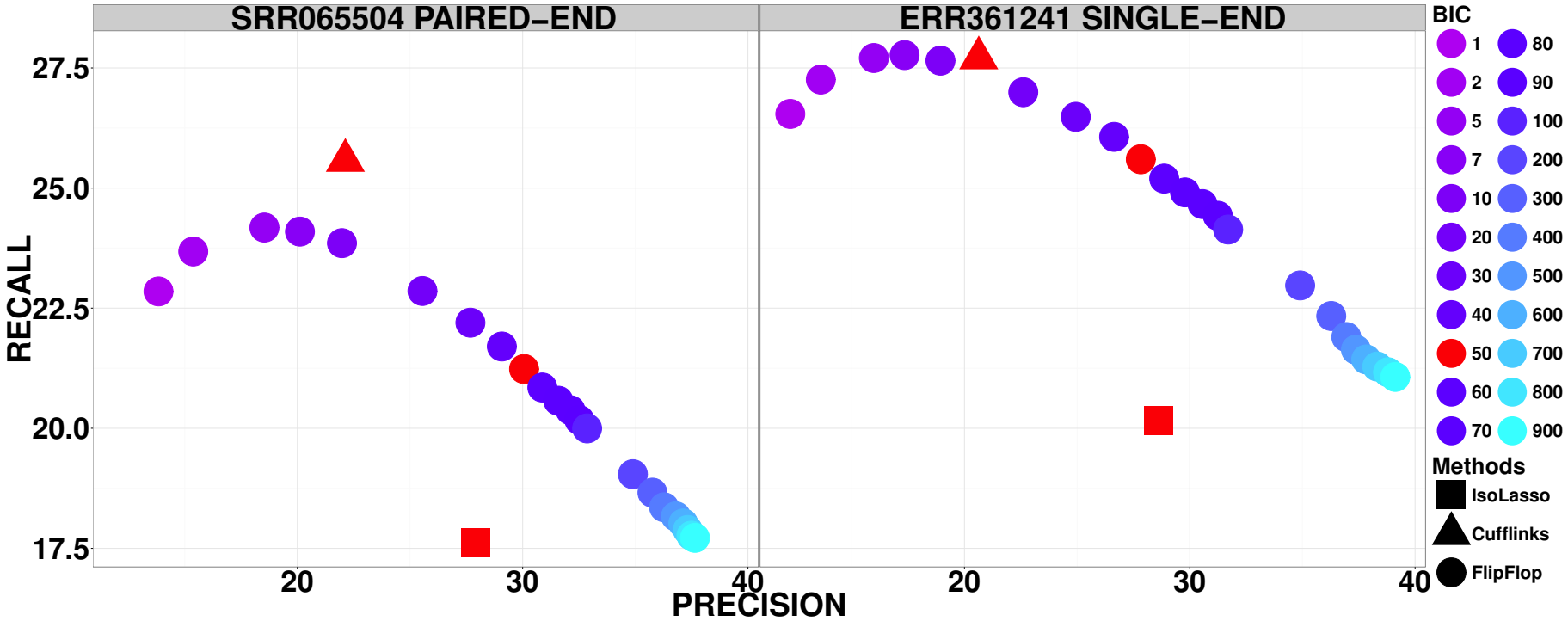
- C++ interfaced with **Matlab, R, Python**.
- proximal gradient methods for ℓ_0, ℓ_1, **elastic-net, fused-Lasso, group-Lasso, tree group-Lasso, tree-ℓ_0, sparse group Lasso, overlapping group Lasso...**
- ...for **square, logistic, multi-class logistic** loss functions.
- handles sparse matrices, provides duality gaps.
- fast implementations of **OMP** and **LARS - homotopy**.
- dictionary learning and matrix factorization (NMF, sparse PCA).
- coordinate descent, block coordinate descent algorithms.
- fast projections onto some convex sets.

Try it! `http://www.di.ens.fr/willow/SPAMS/`

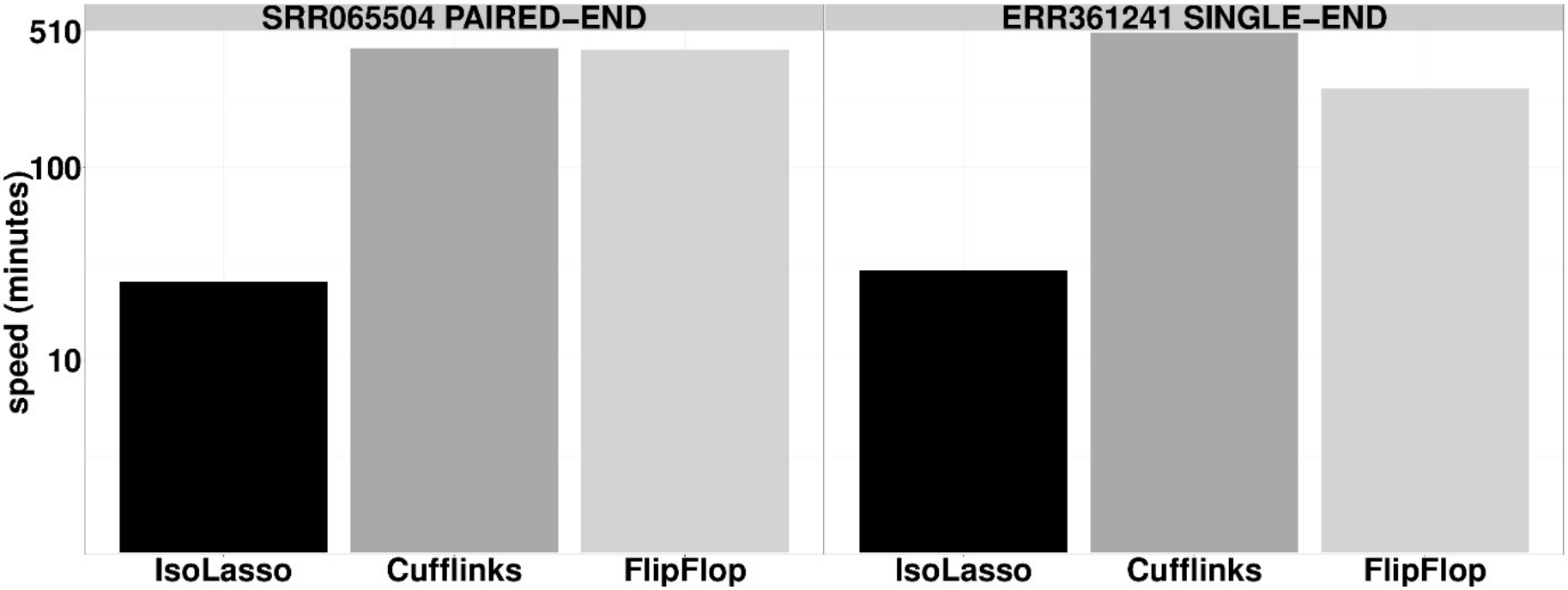
References

- http://cbio.ensmp.fr/flipflop/
- SParse Modelling Software SPAMS
  http://lear.inrialpes.fr/people/mairal/software.php
- H. Jiang and W. H. Wong. *Bioinformatics*, 25(8):1026–1032, 2009.
- C. Trapnell et al. *Nat Biotechnol*, 28(5):511–515, 2010.
- Z. Xia et al. *BMC Bioinformatics*, 12:162, 2011.
- W. Li et al. *J Comput Biol*, 18:1693–1707, 2011.
- J.J. Li et al. *P Natl Acad Sci USA*, 108(50):19867–19872, 2011.
- R. K. Ahuja et al. *Prentice Hall*, 1993.
- D. P. Bertsekas. *Athena Scientific*, 1998.
- J. Mairal and B. Yu. *JMLR*, 2013.

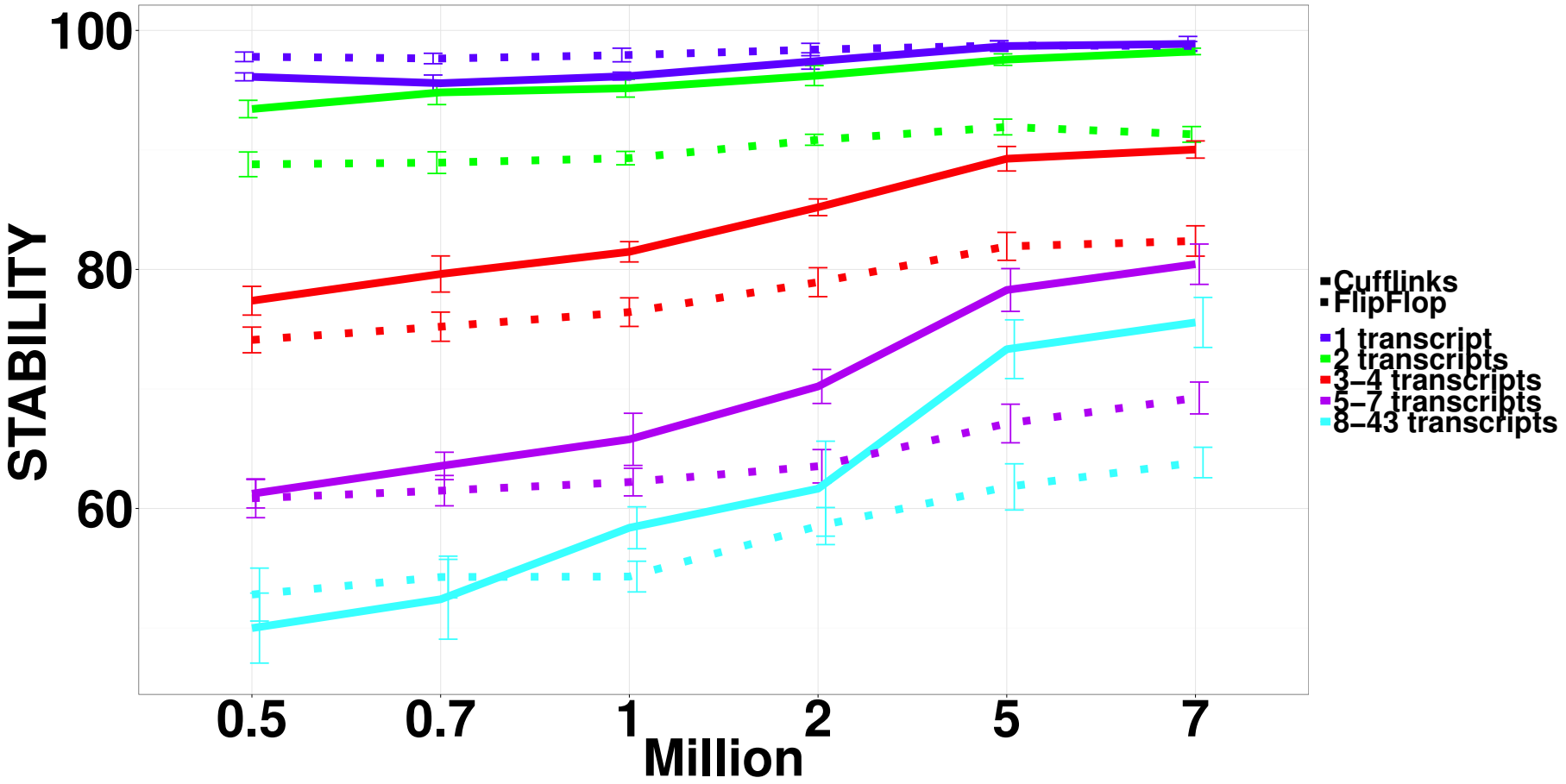
Precision-Recall curves on real data



Speed comparison on real data

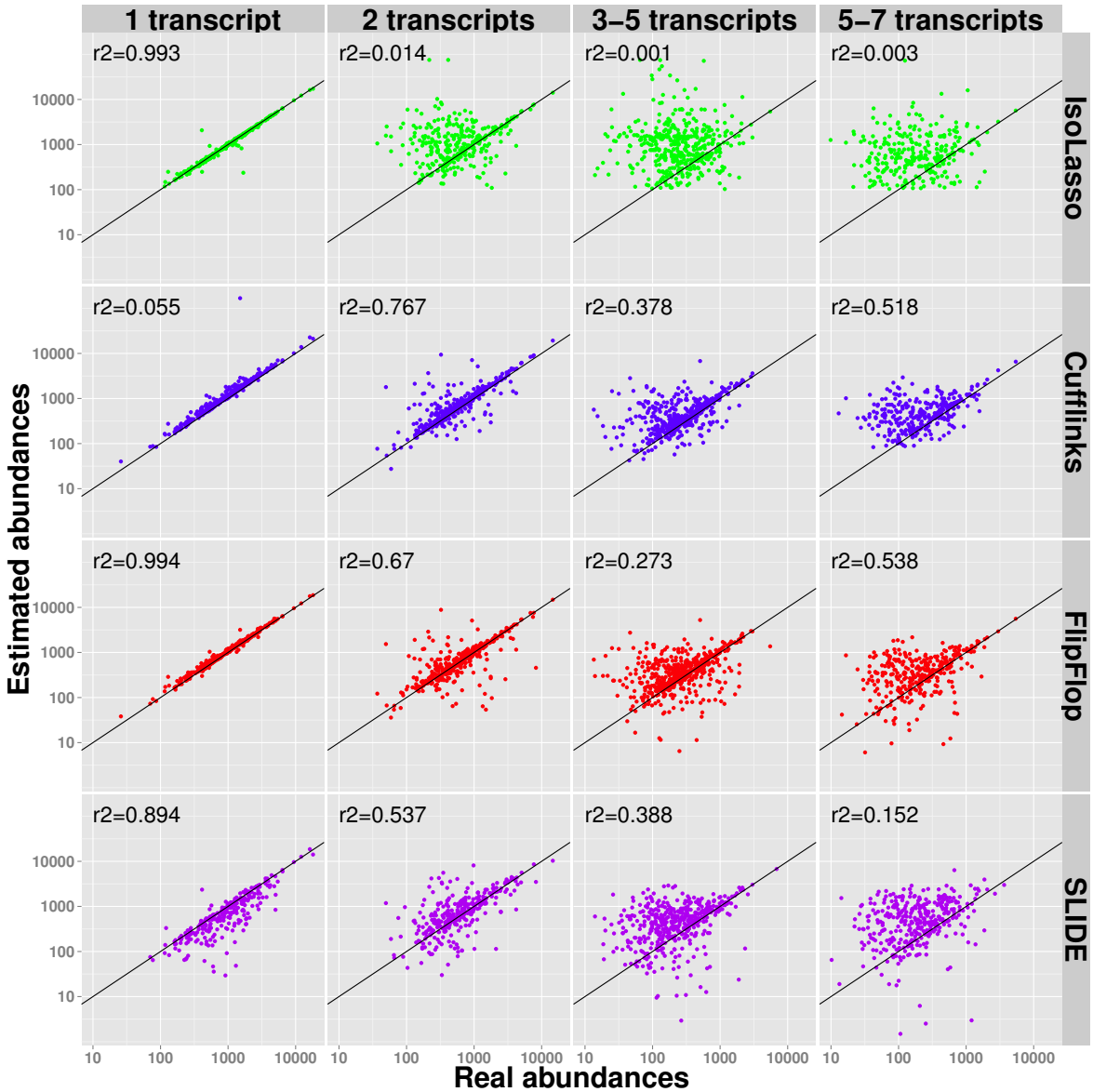


Stability study



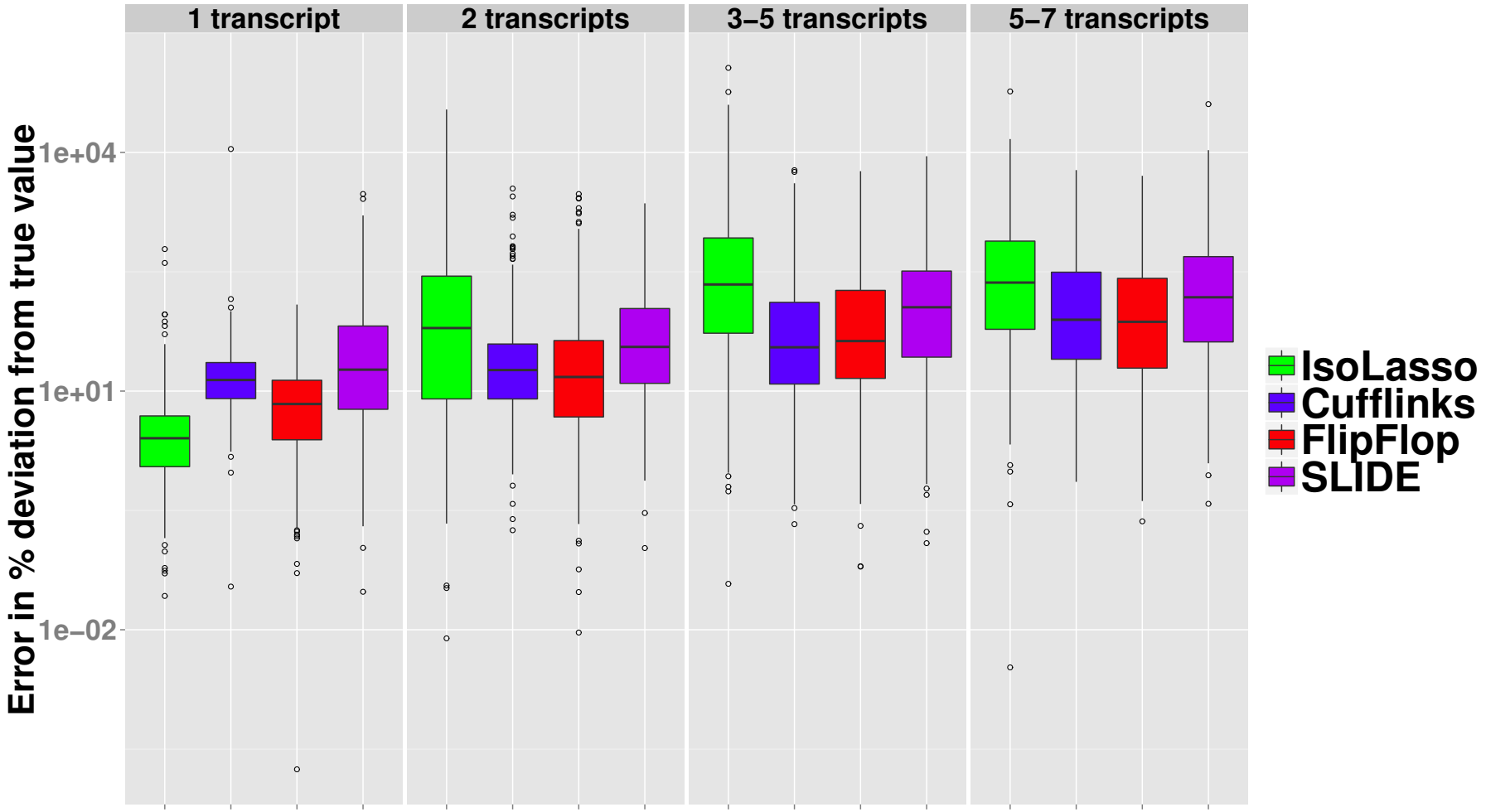
Human Simulation: Abundances

hg19, 1137 genes on chr1, 1million 75 bp single-end reads by transcript levels.



Simulation: Deviation

hg19, 1137 genes on chr1, 1million 75 bp single-end reads by transcript levels.



References I

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20:33–61, 1999.

B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of statistics*, 32(2):407–499, 2004.

B. Gärtner, M. Jaggi, and C. Maria. An exponential lower bound on the complexity of regularization paths. *preprint arXiv:0903.4817v2*, 2010.

J. Giesen, M. Jaggi, and S. Laue. Approximating parameterized convex optimization problems. In *Algorithms - ESA*, Lectures Notes Comp. Sci. 2010.

V. Klee and G. J. Minty. How good is the simplex algorithm? In O. Shisha, editor, *Inequalities*, volume III, pages 159–175. Academic Press, New York, 1972.

References II

M. R. Osborne, B. Presnell, and B. A. Turlach. On the Lasso and its dual. *Journal of Computational and Graphical Statistics*, 9(2):319–37, 2000.

K. Ritter. Ein verfahren zur lösung parameterabhängiger, nichtlinearer maximum-probleme. *Mathematical Methods of Operations Research*, 6(4):149–166, 1962.

R. Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society. Series B*, 58(1):267–288, 1996.

D. Wrinch and H. Jeffreys. XLII. On certain fundamental principles of scientific inquiry. *Philosophical Magazine Series 6*, 42(249):369–390, 1921.

Worst case analysis - Backup Slide

$$\tilde{\mathbf{y}} \triangleq \begin{bmatrix} \mathbf{y} \\ y_{n+1} \end{bmatrix}, \quad \tilde{\mathbf{X}} \triangleq \begin{bmatrix} \mathbf{X} & 2\alpha\mathbf{y} \\ 0 & \alpha y_{n+1} \end{bmatrix},$$

Some intuition about the adverserial strategy:

1. the patterns of the new path must be $[\boldsymbol{\eta}^{i\top}, 0]^\top$ or $[\pm\boldsymbol{\eta}^{i\top}, 1]^\top$;
2. the factor α ensures the $(p+1)$-th variable to enter late the path;
3. after the k first kinks, we have $\mathbf{y} \approx \mathbf{X}\mathbf{w}^\star(\lambda)$ and thus

$$\tilde{\mathbf{X}} \begin{bmatrix} \mathbf{w}^\star(\lambda) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ y_{n+1} \end{bmatrix} \approx \tilde{\mathbf{y}} \approx \tilde{\mathbf{X}} \begin{bmatrix} -\mathbf{w}^\star(\lambda) \\ 1/\alpha \end{bmatrix}.$$

Worst case analysis - Backup Slide 2

$$\min_{\tilde{\mathbf{w}}\in\mathbb{R}^p,\tilde{w}\in\mathbb{R}} \frac{1}{2}\left\|\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\begin{bmatrix}\tilde{\mathbf{w}}\\ \tilde{w}\end{bmatrix}\right\|_2^2 + \lambda\left\|\begin{bmatrix}\tilde{\mathbf{w}}\\ \tilde{w}\end{bmatrix}\right\|_1 =,$$

$$\min_{\tilde{\mathbf{w}}\in\mathbb{R}^p,\tilde{w}\in\mathbb{R}} \frac{1}{2}\|(1-2\alpha\tilde{w})\mathbf{y} - \mathbf{X}\tilde{\mathbf{w}}\|_2^2 + \frac{1}{2}(y_{n+1} - \alpha y_{n+1}\tilde{w})^2 + \lambda\|\tilde{\mathbf{w}}\|_1 + \lambda|\tilde{w}|.$$

is equivalent to

$$\min_{\tilde{\mathbf{w}}'\in\mathbb{R}^p} \frac{1}{2}\|\mathbf{y} - \mathbf{X}\tilde{\mathbf{w}}'\|_2^2 + \frac{\lambda}{|1-2\alpha\tilde{w}^\star|}\|\tilde{\mathbf{w}}'\|_1,$$

and then

$$\tilde{\mathbf{w}}^\star = \begin{cases} (1-2\alpha\tilde{w}^\star)\mathbf{w}^\star\left(\frac{\lambda}{|1-2\alpha\tilde{w}^\star|}\right) & \text{if } \tilde{w}^\star \neq \frac{1}{2\alpha} \\ 0 & \text{otherwise} \end{cases}.$$