RNA isoforms discovery from RNA-seq data

FlipFlop: Fast Lasso based Isoform Prediction as a FLOw Problem

Elsa Bernard\textsuperscript{1}, Laurent Jacob\textsuperscript{2}, Julien Mairal\textsuperscript{3}, Jean-Philippe Vert\textsuperscript{1}

\textsuperscript{1}Mines ParisTech, Institut Curie, INSERM U900
\textsuperscript{2}CNRS - LBBE Laboratory, Lyon
\textsuperscript{3}Inria - LEAR, Grenoble

Persyvact workshop on probabilistic graphical models - July 2014
Team Players

Elsa Bernard
Laurent Jacob
Jean-Philippe Vert


There will be a probabilistic model and a graph
\neq

Bayesian graphical model or
Markov Random Field
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;

RNA-Seq - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/RNA-Seq
RNA-seq (RNA Sequencing), also called "Whole Transcriptome Shotgun Sequencing" ("WTSS"), is a technology that uses the capabilities of next-generation ... Introduction - Methods - Analysis - Application to Genomic Medicine
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

**De Novo approaches**
- Trinity (Grabherr et al. 2011)
- OASES (Schultz et al. 2012)
- Kissplice (Sacomoto et al. 2012)

**Transcripts Quantification using annotations**
- RQuant (Bohnert et al. 2009)
- FluxCapacitor (Montgomery et al. 2010)
- IsoEM (Nicolae et al. 2011)
- eXpress (Roberts et al. 2013)

**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 approaches

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

- 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

**RNA sample**
- transcripts

reads 50-200pb

library preparation

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**

What is new?
Contributions

- NO NEED for FILTERING of candidate isoforms
- FASTER than existing methods that solve the same problem
- adapted to LONG READS
- R package

flow method
Contributions

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

- R package
Contributions

Bioconductor

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:

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

1. Formulation as a Path Selection Problem

2. Sparse Probabilistic Model and Optimization: FlipFlop

3. Results and Perspectives
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:

```
1 → 4 → 5
\downarrow
2 → 3
```

- 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:

```
1 → 4 → 5
\downarrow \quad \downarrow \quad \downarrow
2 → 3
```

- FlipFlop graph:
Graph adapted to long reads

- Splicing graph for a gene with 5 exons:

```
1 -> 4
|   | 5
|   |
2 -> 3
```

- FlipFlop graph: one path with abundance $\beta_1$
Graph adapted to long reads

- Splicing graph for a gene with 5 exons:

- FlipFlop graph: another path with abundance $\beta_2$ ...
Select a Small Number of Paths?

\[ n \text{ exons} \rightarrow \sim 2^n \text{ paths/candidate isoforms} \]

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 \( \beta \) 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 $\beta$ the unknown $c$-dimensional vector of abundance
3. Let $L(\beta)$ quantify whether $\beta$ explains the observed read counts
   - e.g., Poisson negative log-likelihood:
     \[
     L(\beta) = \sum_{\text{node } u} - \log p(X_u) \text{ with } X_u \sim \mathcal{P}(\delta_u) \text{ and } \delta_u \propto l_u \sum_{\text{path } p \ni u} \beta_p
     \]
Select a small number of paths?

Regularization approach

1. Suppose there are $c$ candidate isoforms ($c$ large)
2. Let $\beta$ the unknown $c$-dimensional vector of abundance
3. Let $L(\beta)$ quantify whether $\beta$ explains the observed read counts
   - e.g., Poisson negative log-likelihood:
     \[ L(\beta) = \sum_{\text{node } u} -\log p(X_u) \text{ with } X_u \sim \mathcal{P}(\delta_u) \text{ and } \delta_u \propto l_u \sum_{\text{path } p \ni u} \beta_p \]
4. Regularization-based approaches try to solve:
   \[
   \min_{\beta \in \mathbb{R}_+^c} L(\beta) \text{ such that } \beta \text{ is sparse}
   \]
Isoform Deconvolution with the $\ell_1$-norm

$\ell_1$-regularization

Estimate $\beta$ sparse by solving:

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

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

**Computationally challenging:**

→ IsoLasso: strong filtering
→ NSMAP, SLIDE: number of exons cut-off

**FlipFlop:** Fast Lasso-based Isoform Prediction as a FLOw Problem
→ no filtering
→ no exons restrictions
Regularizing with the $\ell_1$-norm

The projection onto a convex set is “biased” towards singularities.
Regularizing with the $\ell_2$-norm

The $\ell_2$-norm is isotropic.

$\|\beta\|_2 \leq T$
Regularizing with the $\ell_\infty$-norm

The $\ell_\infty$-norm encourages $|\beta_1| = |\beta_2|$.
In 3D.
Copyright G. Obozinski
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

(a) Reads at every node corresponding to one isoform.
(b) 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)

Equivalent flow problem (simpler!)

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

\[ f_{uv} = \sum_{p \ni (u, v)} \beta_p \]  

is a flow

Moreover

\[ \|\beta\|_1 = \sum_{p} \beta_p = f_t \]

Therefore

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

is equivalent to \[ \min_{f \text{ flow}} \tilde{\mathcal{L}}(f) + \lambda f_t \]
Technical details

Poisson Loss (with binary matrix $U$):

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

Flow Decomposition:

$$
f_{uv} = \sum_{p \in P'} \beta_p 1\{(u,v) \in p\}
\Rightarrow f_v = \sum_{u \in V'} f_{uv} = (U^T \beta)_v
$$

Convex Cost Flow:

$$
\min_{f_{\text{flow}}} \sum_{u \in V} \left[ Nl_u f_u - y_u \log(f_u) \right] + \lambda f_t
$$

Solved using $\varepsilon$-relaxation method (Bertsekas 1998).
Summary

Isoform Detection = Path Selection Problem
~ $2^n$ variables (all paths in the splicing graph)

$\leftrightarrow$

Equivalent Network Flow Problem
~ $\frac{n^2}{2}$ variables (all exons and exon-exon junctions in the splicing graph)

↓

Network Flow Algorithms
Efficient Algorithms! Polynomial Time.
Performance increases with read length

- Human Simulation: hg19, 1137 genes on chr1, 1 million reads by transcript levels.

![Graph showing performance increase with read length](image-url)
Performance increases with coverage

- Human Simulation: hg19, 1137 genes on chr1, 1 million reads by transcript levels.

---

### 1 M (150bp) vs 5 M (150bp) vs 10 M (150bp)

#### PRECISION vs RECALL

- IsoLasso
- Cufflinks
- FlipFlop
- NSMAP

- 1 transcript
- 2 transcripts
- 3–4 transcripts
- 5–7 transcripts
- 8–43 transcripts
Extension to paired-end reads OK

- Human Simulation: hg19, 1137 genes on chr1, 1 million reads by transcript levels.

![Graph showing precision and recall for different read lengths and transcript numbers.](image-url)
Speed Trial

- Human Simulation: hg19, 1137 genes on chr1, 1 million reads by transcript levels.

<table>
<thead>
<tr>
<th>Number of Exons</th>
<th>CPU Time (ms) by gene</th>
</tr>
</thead>
<tbody>
<tr>
<td>2−5 exons</td>
<td></td>
</tr>
<tr>
<td>5−10 exons</td>
<td></td>
</tr>
<tr>
<td>10−20 exons</td>
<td></td>
</tr>
<tr>
<td>20−116 exons</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tool</th>
<th>2−5 exons</th>
<th>5−10 exons</th>
<th>10−20 exons</th>
<th>20−116 exons</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsoLasso</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cufflinks</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FlipFlop</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NSMAP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLIDE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Graphs showing CPU time for different exon numbers:

- Number of EXONS vs. Elapsed TIME (s)
- CPU time (ms) by gene for different exon numbers
GC bias - Precision-Recall curve

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

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

- http://cbio.ensmp.fr/flipflop/
- SParse Modelling Software SPAMS
  http://lear.inrialpes.fr/people/mairal/software.php
Precision-Recall curves on real data
Speed comparison on real data

The image shows a bar chart comparing the speed of different tools (IsoLasso, Cufflinks, FlipFlop) on two types of data: SRR065504 PAIRED-END and ERR361241 SINGLE-END. The Y-axis represents time in minutes, ranging from 10 to 510 minutes. The chart indicates that FlipFlop generally performs faster than IsoLasso and Cufflinks for both types of data.
Stability study

- Cufflinks
- FlipFlop
- 1 transcript
- 2 transcripts
- 3–4 transcripts
- 5–7 transcripts
- 8–43 transcripts

STABILITY vs Million
Human Simulation: Abundances

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

<table>
<thead>
<tr>
<th>1 transcript</th>
<th>2 transcripts</th>
<th>3−5 transcripts</th>
<th>5−7 transcripts</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r^2=0.994$</td>
<td>$r^2=0.67$</td>
<td>$r^2=0.388$</td>
<td>$r^2=0.152$</td>
</tr>
<tr>
<td>$r^2=0.894$</td>
<td>$r^2=0.537$</td>
<td>$r^2=0.388$</td>
<td>$r^2=0.003$</td>
</tr>
<tr>
<td>$r^2=0.994$</td>
<td>$r^2=0.767$</td>
<td>$r^2=0.538$</td>
<td>$r^2=0.518$</td>
</tr>
<tr>
<td>$r^2=0.099$</td>
<td>$r^2=0.055$</td>
<td>$r^2=0.003$</td>
<td>$r^2=0.001$</td>
</tr>
</tbody>
</table>

**IsoLasso**

**Cufflinks**

**FlipFlop**

**SLIDE**
Simulation: Deviation

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

<table>
<thead>
<tr>
<th></th>
<th>1 transcript</th>
<th>2 transcripts</th>
<th>3–5 transcripts</th>
<th>5–7 transcripts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error in % deviation from true value</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IsoLasso</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cufflinks</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FlipFlop</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLIDE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

J. Mairal

FlipFlop 39/39
Part IV: Back to Structured Sparsity
(depending on time)
What about more complicated norms?

Copyright G. Obozinski
What about more complicated norms?
Group Lasso
[Turlach et al., 2005, Yuan and Lin, 2006]

the $\ell_1/\ell_q$-norm: $\Omega(\beta) = \sum_{g \in G} \|\beta_g\|_q$.

- $G$ is a **partition** of $\{1, \ldots, p\}$;
- $q = 2$ or $q = \infty$ in practice;
- can be interpreted as the $\ell_1$-norm of $[\|\beta_g\|_q]_{g \in G}$.

$$
\Omega(\beta) = \|\beta_{\{1,2}\}\|_2 + |\beta_3|.
$$
Structured sparsity with overlapping groups

**Warning:** Under the name “structured sparsity” appear in fact significantly different formulations!

1. **non-convex**
   - zero-tree wavelets [Shapiro, 1993];
   - predefined collection of sparsity patterns: [Baraniuk et al., 2010];
   - select a union of groups: [Huang et al., 2009];
   - structure via Markov Random Fields: [Cehver et al., 2008];

2. **convex (norms)**
   - tree-structure: [Zhao et al., 2009];
   - select a union of groups: [Jacob et al., 2009];
   - zero-pattern is a union of groups: [Jenatton et al., 2009];
   - other norms: [Micchelli et al., 2010].
Group Lasso with overlapping groups
[Jenatton, Audibert, and Bach, 2009]

\[ \Omega(\beta) = \sum_{g \in G} \| \beta_g \|_q. \]

What happens when the groups overlap?

- the pattern of non-zero variables is an intersection of groups;
- the zero pattern is a union of groups.

\[ \Omega(\beta) = \| \beta \|_2 + |\beta_2| + |\beta_3|. \]
Hierarchical Norms
[Zhao, Rocha, and Yu, 2009]

A node can be active only if its ancestors are active.
The selected patterns are rooted subtrees.
Modelling Patterns as Unions of Groups
the non-convex penalty of Huang, Zhang, and Metaxas [2009]

Warning: different point of view than the two previous slides

\[ \varphi(\beta) \triangleq \min_{\mathcal{J} \subseteq \mathcal{G}} \left\{ \sum_{g \in \mathcal{J}} \eta_g \text{ s.t. } \text{Supp}(\beta) \subseteq \bigcup_{g \in \mathcal{J}} g \right\}. \]

- the penalty is non-convex.
- is NP-hard to compute (set cover problem).
- The pattern of non-zeroes in \( \beta \) is a union of (a few) groups.

It can be rewritten as a boolean linear program:

\[ \varphi(\beta) = \min_{x \in \{0,1\}^{|\mathcal{G}|}} \left\{ \eta^T x \text{ s.t. } Nx \geq \text{Supp}(\beta) \right\}. \]
Modelling Patterns as Unions of Groups

convex relaxation and the penalty of Jacob, Obozinski, and Vert [2009]

The penalty of Huang et al. [2009]:

$$\varphi(\beta) = \min_{x \in \{0,1\}^{|g|}} \left\{ \eta^T x \text{ s.t. } Nx \geq \text{Supp}(\beta) \right\}.$$  

A convex LP-relaxation:

$$\psi(\beta) \triangleq \min_{x \in \mathbb{R}_+^{|g|}} \left\{ \eta^T x \text{ s.t. } Nx \geq |\beta| \right\}.$$  

**Lemma:** $\psi$ is the penalty of Jacob et al. [2009] with the $\ell_\infty$-norm:

$$\psi(\beta) = \min_{(\xi_g \in \mathbb{R}^p)_{g \in \mathcal{G}}} \sum_{g \in \mathcal{G}} \eta_g \|\xi_g\|_\infty \text{ s.t. } \beta = \sum_{g \in \mathcal{G}} \xi_g \text{ and } \forall g, \text{ Supp}(\xi_g) \subseteq g,$$  

J. Mairal

FlipFlop
Modelling Patterns as Unions of Groups

The norm of Jacob et al. [2009] in 3D

\( \psi(\beta) \) with \( G = \{\{1, 2\}, \{2, 3\}, \{1, 3\}\} \).
Graph sparsity
\(G = (V, E), \text{ with } V = \{1, \ldots, p\}\)
Graph sparsity

Encouraging patterns with a small number of connected components
Formulation

\[ \min_{\beta \in \mathbb{R}^p} \left( R(\beta) \right) \quad + \quad \lambda \Omega(\beta) , \]

\( \Omega \) should encourage connected patterns in the graph.

- the penalty of Huang et al. [2009]:

\[ \varphi(\beta) = \min_{\eta \in \{0,1\}^{|G|}} \left\{ \eta^\top x \quad \text{s.t.} \quad Nx \geq \text{Supp}(\beta) \right\} . \]

- a convex LP-relaxation (penalty of Jacob et al. [2009]):

\[ \psi(\beta) \triangleq \min_{\eta \in \mathbb{R}^{|G|}_+} \left\{ \eta^\top x \quad \text{s.t.} \quad Nx \geq |\beta| \right\} . \]
Structured sparsity for graphs
Group structure for graphs.

Natural choices to encourage connectivity in the graph is to define $\mathcal{G}$ as

1. pairs of vertices linked by an arc. **only models local interactions**;
2. all connected subgraphs up to a size $L$. **cumbersome/intractable**;
3. all connected subgraphs. **intractable**.

**Question**

*Can we replace connected subgraphs by another structure which (i) is rich enough to model long-range interactions in the graph, and (ii) leads to computationally feasible penalties?*
A solution when the graph is a DAG (Mairal and Yu, 2012)

1. Define $G$ to be the set of all paths in the DAG.
2. Define $\eta_g$ to be $\gamma + |g|$ (the cost of selecting a path $g$).

\[ \varphi(\beta) = (\gamma + 3) + (\gamma + 3) \]
Graph sparsity for DAGs

Decomposability of the weights $\eta_g = \gamma + |g|$
Equivalence to network flows

An optimization problem on paths might be transformed into an equivalent flow problem.

Proposition 1

$$\varphi(\beta) = \min_{f \in \mathcal{F}} \sum_{(u,v) \in E'} f_{uv} c_{uv} \quad \text{s.t.} \quad s_j(f) \geq 1, \; \forall j \in \text{Supp}(\beta),$$

Proposition 2

$$\psi(\beta) = \min_{f \in \mathcal{F}} \sum_{(u,v) \in E'} f_{uv} c_{uv} \quad \text{s.t.} \quad s_j(f) \geq |\beta_j|, \; \forall j \in \{1, \ldots, p\},$$

$$\varphi(\beta), \psi(\beta)$$ and similarly the proximal operators, the dual norm of $$\psi$$ can be computed in polynomial time using network flow optimization.
References I


References II


