Segmentation of Counting Processes and Dynamical Models

PhD Thesis Defense

Mokhtar Zahdi Alaya

June 27, 2016

Plan

1. Motivations
2. Learning the intensity of time events with change-points
   - Piecewise constant intensity
   - Estimation procedure
   - Change-points detection + Numerical experiments
3. Binarsity
   - Features binarization
   - Binarsity penalization
   - Generalized linear models + binarsity
4. High-dimensional time-varying Aalen and Cox models
   - Weighted $(\ell_1 + \ell_1)$-TV penalization
   - Theoretical guaranties
   - Algorithm + Numerical experiments
5. Conclusion + Perspectives

Plan

Weighted and unweighted TV

- For a chosen positive vector of weights $\hat{w}$, we define the (discrete) weighted total-variation (TV) by

$$\|\beta\|_{\mathrm{TV},\hat{w}} = \sum_{j=2}^{p} \hat{w}_j |\beta_j - \beta_{j-1}|, \text{ for all } \beta \in \mathbb{R}^p.$$

- If $\hat{w} \equiv 1$, then we define the unweighted TV by

$$\|\beta\|_{\mathrm{TV}} = \sum_{j=2}^{p} |\beta_j - \beta_{j-1}|, \text{ for all } \beta \in \mathbb{R}^p.$$

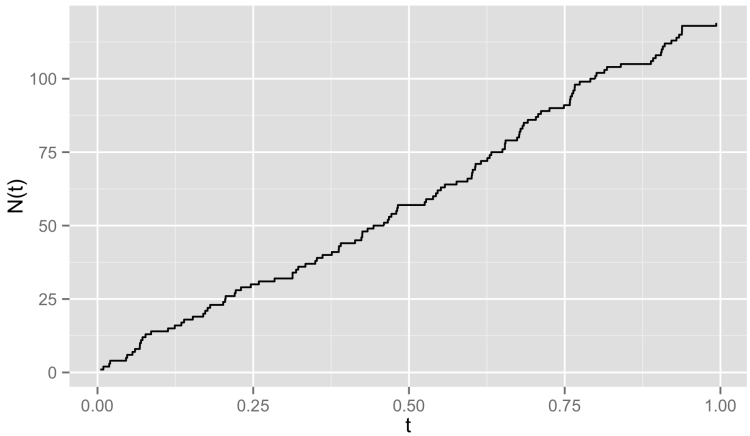
Motivations for using TV

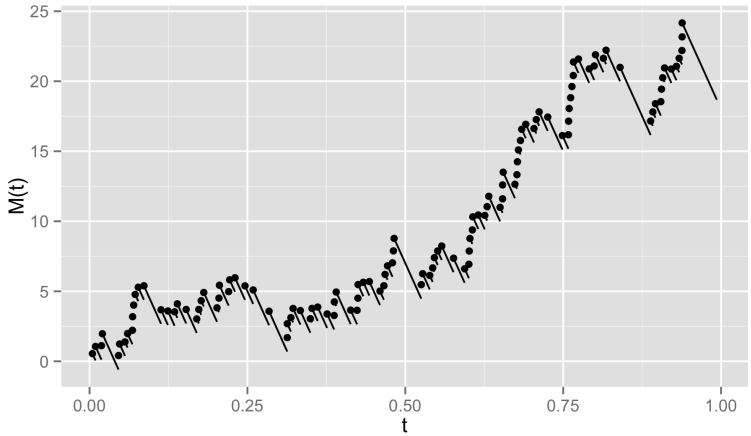
- Appropriate for multiple change-points estimation.
  $\longrightarrow$ Partitioning a nonstationary signal into several contiguous stationary segments of variable duration [Harchaoui and Lévy-Leduc (2010)].
- Widely used in sparse signal processing and imaging (2D) [Chambolle et al. (2010)].
- Enforces sparsity in the discrete gradient, which is desirable for applications with features ordered in some meaningful way [Tibshirani et al. (2005)].

Plan

Counting process: stochastic setup

- $N = \{N(t)\}_{0 \le t \le 1}$ is a counting process.





- Doob-Meyer decomposition:

$$N(t) = \underbrace{\Lambda_0(t)}_{\text{compensator}} + \underbrace{M(t),}_{\text{martingale}} \quad 0 \le t \le 1.$$

- The intensity of N is defined by

$$\lambda_0(t)dt = d\Lambda_0(t) = \mathbb{P}[N \text{ has a jump in } [t, t+dt) | \mathcal{F}(t)],$$

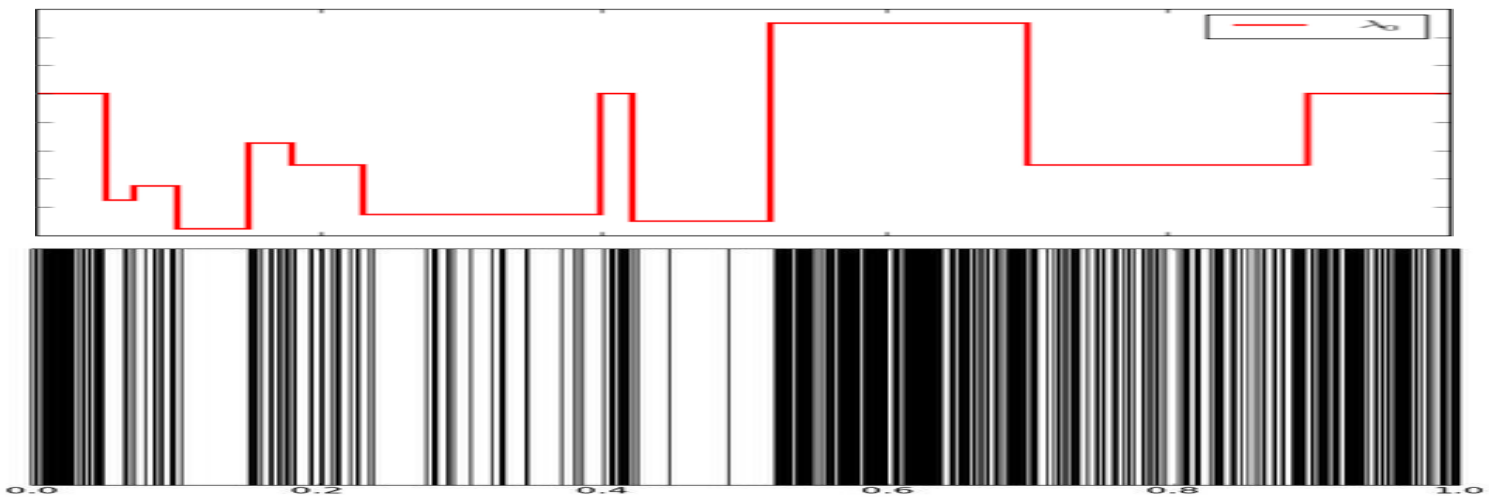
where $\mathcal{F}(t) = \sigma(N(s), s \le t)$.

Piecewise constant intensity

- Assume that

$$\lambda_0(t) = \sum_{\ell=1}^{L_0} \beta_{0,\ell} \mathbb{1}_{(\tau_{0,\ell-1}, \tau_{0,\ell}]}(t),\ 0 \leq t \leq 1.$$

- $\{\tau_{0,0} = 0 < \tau_{0,1} < \cdots < \tau_{0,L_0-1} < \tau_{0,L_0} = 1\}$: set of true change-points.
- $\{\beta_{0,\ell} : 1 \leq \ell \leq L_0\}$: set of jump sizes of λ_0.
- L_0 : number of true change-points.



Assumption on observations

Data

We observe n i.i.d copies of N on $[0,1]$, denoted $N_1, \ldots, N_n$.

- We define $\bar{N}_n(I) = \frac{1}{n}\sum_{i=1}^{n} N_i(I)$, $N_i(I) = \int_I dN_i(t)$, for any interval $I \subset [0,1]$.
- This assumption is equivalent to observing a single process N with intensity $n\lambda_0$ (only used to have a notion of growing observations with an increasing n).

A procedure based on total-variation penalization

- We introduce the least-squares functional

$$R_n(\lambda) = \int_0^1 \lambda(t)^2 dt - \frac{2}{n} \sum_{i=1}^{n} \int_0^1 \lambda(t) dN_i(t),$$

[Reynaud-Bouret (2003, 2006), Gaïffas and Guilloux (2012)].

- Fix $m = m_n \geq 1$, an integer that shall go to infinity as $n \to \infty$.
- We approximate λ_0 in the set of nonnegative piecewise constant functions on $[0, 1]$ given by

$$\Lambda_m = \left\{ \lambda_\beta = \sum_{j=1}^{m} \beta_{j,m} \lambda_{j,m} : \beta = [\beta_{j,m}]_{1 \leq j \leq m} \in \mathbb{R}_+^m \right\},$$

where

$$\lambda_{j,m} = \sqrt{m} \mathbb{1}_{I_{j,m}} \quad \text{et} \quad I_{j,m} = \left(\frac{j-1}{m}, \frac{j}{m} \right].$$

A procedure based on total-variation penalization

- The estimator of λ_0 is defined by

$$\hat{\lambda} = \lambda_{\hat{\beta}} = \sum_{j=1}^{m} \hat{\beta}_{j,m} \lambda_{j,m}.$$

where $\hat{\beta}$ is giving by

$$\hat{\beta} = \underset{\beta \in \mathbb{R}_+^m}{\operatorname{argmin}} \left\{ R_n(\lambda_\beta) + \|\beta\|_{\mathrm{TV},\hat{w}} \right\}.$$

- We consider the dominant term

$$\hat{w}_j \approx \sqrt{\frac{m \log m}{n} \bar{N}_n\Big(\big(\frac{j-1}{m}, 1\big]\Big)}.$$

Oracle inequality with fast rate

- The linear space Λ_m is endowed by the norm

$$\|\lambda\| = \sqrt{\int_0^1 \lambda^2(t)dt}.$$

- Let $\hat{S}$ to be the support of the discrete gradient of $\hat{\beta}$,

$$\hat{S} = \{j : \hat{\beta}_{j,m} \neq \hat{\beta}_{j-1,m} \text{ for } j = 2, \ldots, m\}.$$

- Let $\hat{L}$ to be the estimated number of change-points defined by:

$$\hat{L} = |\hat{S}|.$$

Oracle inequality with fast rate

The estimator $\hat{\lambda}$ satisfies the following:

Theorem 1

Fix $x > 0$ and let the data-driven weights $\hat{w}$ defined as above. Assume that $\hat{L}$ satisfies $\hat{L} \leq L_{\max}$. Then, we have

$$\begin{aligned}\|\hat{\lambda} - \lambda_0\|^2 \leq \inf_{\beta \in \mathbb{R}_+^m} \left\|\lambda_\beta - \lambda_0\right\|^2 + 6(L_{\max} + 2(L_0 - 1)) \max_{1 \leq j \leq m} \hat{w}_j^2 \\ + C_1 \frac{\|\lambda_0\|_\infty \left(x + L_{\max}(1 + \log m)\right)}{n} \\ + C_2 \frac{m\left(x + L_{\max}(1 + \log m)\right)^2}{n^2},\end{aligned}$$

with a probability larger than $1 - L_{\max} e^{-x}$.

Oracle inequality with fast rate

- Let $\Delta_{\beta,\max} = \max_{1\leq \ell,\ell' \leq L_0} |\beta_{0,\ell} - \beta_{0,\ell'}|$, be the maximum of jump size of λ_0.

Corollary

We have

$$\|\lambda_\beta - \lambda_0\|^2 \leq \frac{2(L_0 - 1)\Delta^2_{\beta,\max}}{m}.$$

- Our procedure has a fast rate of convergence of order

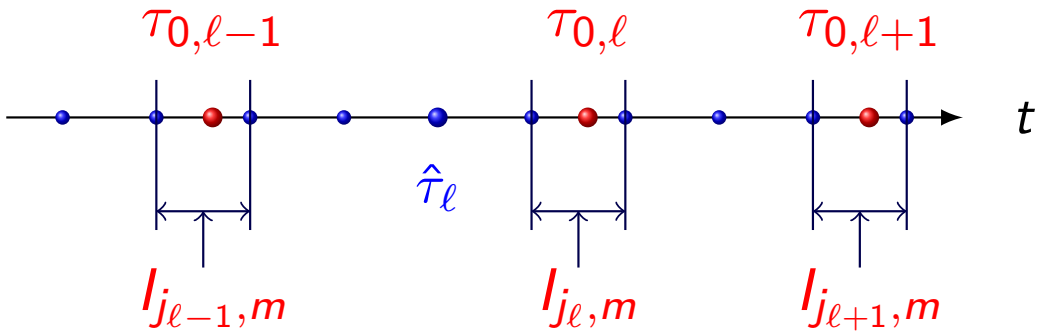
$$\frac{(L_{\max} \vee L_0) m \log m}{n}.$$

- An optimal tradeoff between approximation and complexity is given by the choice:

$$\text{If } L_{\max} = O(m) \Rightarrow m \approx n^{1/3}.$$
$$\text{If } L_{\max} = O(1) \Rightarrow m \approx n^{1/2}.$$

Consistency of change-points detection

- There is an unavoidable non-parametric bias of approximation.
- The *approximate change-points sequence* $(\frac{j_\ell}{m})_{0\leq \ell \leq L_0}$ is defined as the *right-hand side boundary* of the unique interval $I_{j_\ell,m}$ that contains the true change-point $\tau_{0,\ell}$.
- $\tau_{0,\ell} \in \left(\frac{j_\ell - 1}{m}, \frac{j_\ell}{m}\right]$, for $\ell = 1, \ldots, L_0 - 1$, where $j_0 = 0$ and $j_{L_0} = m$ by convention.



- Let $\hat{S} = \{\hat{j}_1, \ldots, \hat{j}_{\hat{L}}\}$ with $\hat{j}_1 < \cdots < \hat{j}_{\hat{L}}$, and $\hat{j}_0 = 0$ and $\hat{j}_{\hat{L}+1} = m$.
- We define simply

$$\hat{\tau}_\ell = \frac{\hat{j}_\ell}{m} \text{ for } \ell = 0, \ldots, \hat{L} + 1.$$

Consistency of change-points detection

- We can't recover the exact position of two change-points if they lie on the same interval $I_{j,m}$.

Minimal distance between true change-points

Assume that there is a positive constant $c \geq 8$ such that

$$\min_{1 \leq \ell \leq L_0} |\tau_{0,\ell} - \tau_{0,\ell-1}| > \frac{c}{m}.$$

$\longrightarrow$ The change-points of λ_0 are sufficiently far apart.
$\longrightarrow$ There cannot be more than one change-point in the "high-resolution" intervals $I_{j,m}$.

- The procedure will be able to recover the (unique) intervals $I_{j_\ell,m}$, for $\ell = 0, \ldots, L_0$, where the change-point belongs.

Consistency of change-points detection

- $\Delta_{j,\min} = \min_{1 \leq \ell \leq L_0 - 1} |j_{\ell+1} - j_\ell|$, the minimum distance between two consecutive terms in the change-points of λ_0.
- $\Delta_{\beta,\min} = \min_{1 \leq q \leq m-1} |\beta_{0,q+1,m} - \beta_{0,q,m}|$, the smallest jump size of the projection $\lambda_{0,m}$ of λ_0 onto Λ_m.
- $(\varepsilon_n)_{n \geq 1}$, a non-increasing and positive sequence that goes to zero as $n \to \infty$.

Technical Assumptions

We assume that $\Delta_{j,\min}$, $\Delta_{\beta,\min}$ and $(\varepsilon_n)_{n \geq 1}$ satisfy

$$\frac{\sqrt{nm}\varepsilon_n \Delta_{\beta,\min}}{\sqrt{\log m}} \to \infty \quad \text{and} \quad \frac{\sqrt{n}\Delta_{j,\min}\Delta_{\beta,\min}}{\sqrt{m \log m}} \to \infty, \quad \text{as} \quad n \to \infty.$$

Consistency of change-points detection

Theorem 2

Under the given Assumptions, and if $\hat{L} = L_0 - 1$, then the change-points estimators $\{\hat{\tau}_1, \ldots, \hat{\tau}_{\hat{L}}\}$ satisfy

$$\mathbb{P}\Big[\max_{1 \leq \ell \leq L_0 - 1} |\hat{\tau}_\ell - \tau_{0,\ell}| \leq \varepsilon_n\Big] \to 1, \text{ as } n \to \infty.$$

- If $m \approx n^{1/3}$, Theorem 2 holds with $\varepsilon_n \approx n^{-1/3}$, $\Delta_{\beta,\min} = n^{-1/6}$ et $\Delta_{j,\min} \geq 6$.
- $m \approx n^{1/2}$, Theorem 2 holds with $\varepsilon_n \approx n^{-1/2}$, $\Delta_{\beta,\min} = n^{-1/6}$ et $\Delta_{j,\min} \geq 6$.

Proximal operator + algorithm

- We are interested in computing a solution

$$x^\star = \underset{x \in \mathbb{R}^p}{\operatorname{argmin}} \{g(x) + h(x)\},$$

  where g is smooth and h is simple (prox-calculable).
- The proximal operator prox_h of a proper, lower semi-continuous, convex function $h : \mathbb{R}^m \to (-\infty, \infty]$, is defined as

$$\operatorname{prox}_h(v) = \underset{x \in \mathbb{R}^m}{\operatorname{argmin}} \left\{ \frac{1}{2}\|v - x\|_2^2 + h(x) \right\}, \text{ for all } v \in \mathbb{R}^m.$$

- Proximal gradient descent (PGD) algorithm is based on

$$x^{(k+1)} = \operatorname{prox}_{\varepsilon_k h}\left(x^{(k)} - \varepsilon_k \nabla g(x^{(k)})\right).$$

[Daubechies et al. (2004) (ISTA) , Beck and Teboulle (2009) (FISTA)]

Proximal operator of the weighted TV penalization

- We have

$$\hat{\beta} = \underset{\beta \in \mathbb{R}_+^m}{\operatorname{argmin}} \left\{ \frac{1}{2} \|\mathbf{N} - \beta\|_2^2 + \|\beta\|_{\mathrm{TV},\hat{w}} \right\},$$

  where $\mathbf{N} = [\mathbf{N}_j]_{1 \leq j \leq m} \in \mathbb{R}_+^m$ is given by

$$\mathbf{N} = \left(\sqrt{m} \bar{N}_n(I_{1,m}), \ldots, \sqrt{m} \bar{N}_n(I_{m,m}) \right).$$

- Then

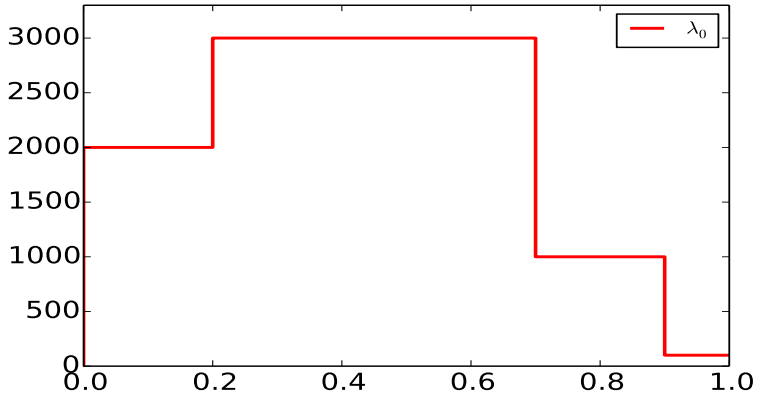
$$\hat{\beta} = \mathrm{prox}_{\|\cdot\|_{\mathrm{TV},\hat{w}}}(\mathbf{N}).$$

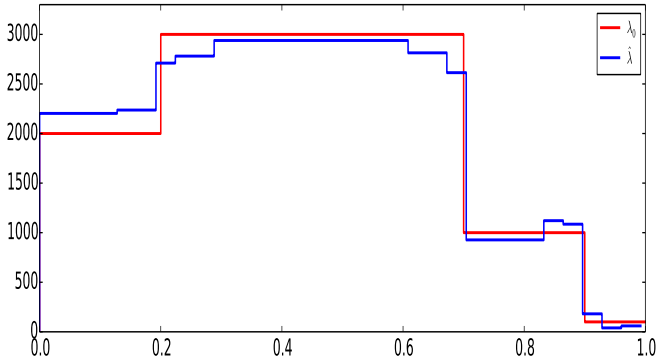
- Modification of Condat's algorithm [Condat (2013)].
- If we have a feasible dual variable $\hat{u}$, we can compute the primal solution $\hat{\beta}$, by Fenchel duality.
- The Karush-Kuhn-Tucker (KKT) optimality conditions characterize the unique solutions $\hat{\beta}$ and $\hat{u}$.

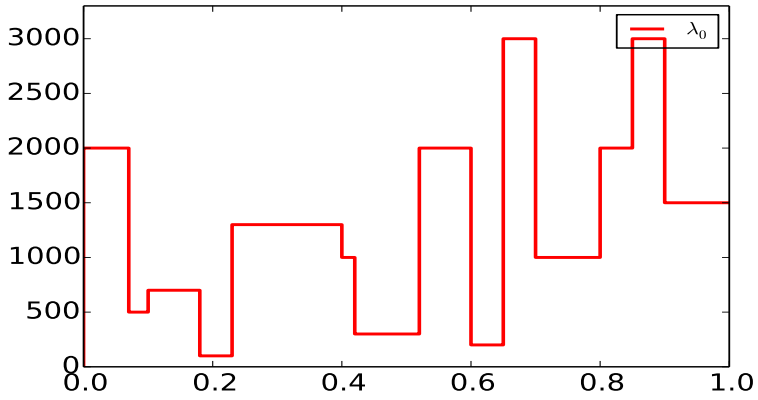
Algorithm 1: $\hat{\beta} = \text{prox}_{\|\cdot\|_{\text{TV},\hat{w}}}(\mathbf{N})$

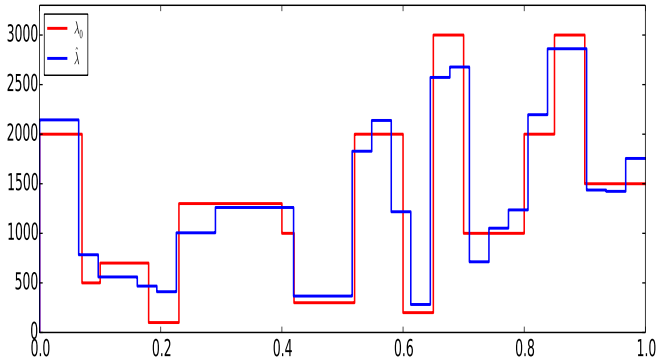
1. **set** $k = k_0 = k_- = k_+ \leftarrow 1$; $\beta_{\min} \leftarrow \mathbf{N}_1 - \hat{w}_2$; $\beta_{\max} \leftarrow \mathbf{N}_1 + \hat{w}_2$; $\theta_{\min} \leftarrow \hat{w}_2$; $\theta_{\max} \leftarrow -\hat{w}_2$;
2. **if** $k = m$ **then**
   $\hat{\beta}_m \leftarrow \beta_{\min} + \theta_{\min}$;
3. **if** $\mathbf{N}_{k+1} + \theta_{\min} < \beta_{\min} - \hat{w}_{k+2}$ **then** `/* negative jump */`
   $\hat{\beta}_{k_0} = \cdots = \hat{\beta}_{k_-} \leftarrow \beta_{\min}$; $k = k_0 = k_- = k_+ \leftarrow k_- + 1$;
   $\beta_{\min} \leftarrow \mathbf{N}_k - \hat{w}_{k+1} + \hat{w}_k$; $\beta_{\max} \leftarrow \mathbf{N}_k + \hat{w}_{k+1} + \hat{w}_k$; $\theta_{\min} \leftarrow \hat{w}_{k+1}$; $\theta_{\max} \leftarrow -\hat{w}_{k+1}$;
4. **else if** $\mathbf{N}_{k+1} + \theta_{\max} > \beta_{\max} + \hat{w}_{k+2}$ **then** `/* positive jump */`
   $\hat{\beta}_{k_0} = \ldots = \hat{\beta}_{k_+} \leftarrow \beta_{\max}$; $k = k_0 = k_- = k_+ \leftarrow k_+ + 1$;
   $\beta_{\min} \leftarrow \mathbf{N}_k - \hat{w}_{k+1} - \hat{w}_k$; $\beta_{\max} \leftarrow \mathbf{N}_k + \hat{w}_{k+1} - \hat{w}_k$; $\theta_{\min} \leftarrow \hat{w}_{k+1}$; $\theta_{\max} \leftarrow -\hat{w}_{k+1}$;
5. **else** `/* no jump */`
   set $k \leftarrow k + 1$; $\theta_{\min} \leftarrow \mathbf{N}_k + \hat{w}_{k+1} - \beta_{\min}$; $\theta_{\max} \leftarrow \mathbf{N}_k - \hat{w}_{k+1} - \beta_{\max}$;
   if $\theta_{\min} \geq \hat{w}_{k+1}$ **then**
      $\beta_{\min} \leftarrow \beta_{\min} + \frac{\theta_{\min} - \hat{w}_{k+1}}{k - k_0 + 1}$; $\theta_{\min} \leftarrow \hat{w}_{k+1}$; $k_- \leftarrow k$;
   if $\theta_{\max} \leq -\hat{w}_{k+1}$ **then**
      $\beta_{\max} \leftarrow \beta_{\max} + \frac{\theta_{\max} + \hat{w}_{k+1}}{k - k_0 + 1}$; $\theta_{\max} \leftarrow -\hat{w}_{k+1}$; $k_+ \leftarrow k$;
6. **if** $k < m$ **then**
   go to 3.;
7. **if** $\theta_{\min} < 0$ **then**
   $\hat{\beta}_{k_0} = \cdots = \hat{\beta}_{k_-} \leftarrow \beta_{\min}$; $k = k_0 = k_- \leftarrow k_- + 1$; $\beta_{\min} \leftarrow \mathbf{N}_k - \hat{w}_{k+1} + \hat{w}_k$;
   $\theta_{\min} \leftarrow \hat{w}_{k+1}$; $\theta_{\max} \leftarrow \mathbf{N}_k + \hat{w}_k - v_{\max}$; **go to 2.**;
8. **else if** $\theta_{\max} > 0$ **then**
   $\hat{\beta}_{k_0} = \cdots = \hat{\beta}_{k_+} \leftarrow \beta_{\max}$; $k = k_0 = k_+ \leftarrow k_+ + 1$; $\beta_{\max} \leftarrow \mathbf{N}_k + \hat{w}_{k+1} - \hat{w}_k$;
   $\theta_{\max} \leftarrow -\hat{w}_{k+1}$; $\theta_{\min} \leftarrow \mathbf{N}_k - \hat{w}_k - \theta_{\min}$; **go to 2.**;
9. **else**
   $\hat{\beta}_{k_0} = \cdots = \hat{\beta}_m \leftarrow \beta_{\min} + \frac{\theta_{\min}}{k - k_0 + 1}$;

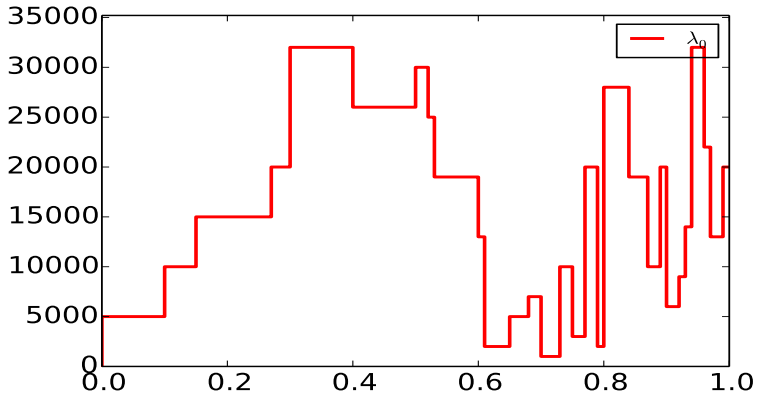
Simulated data: example with 5, 15 and 30 change-points

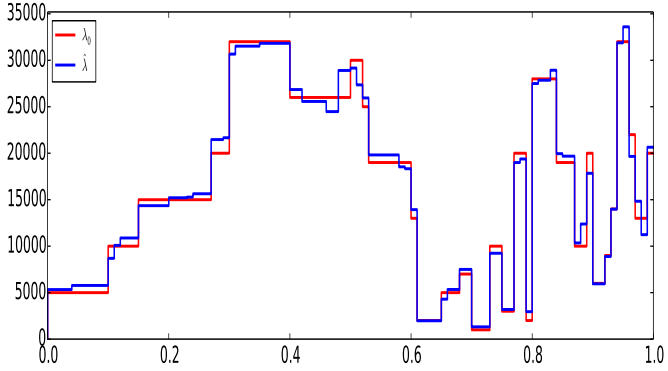










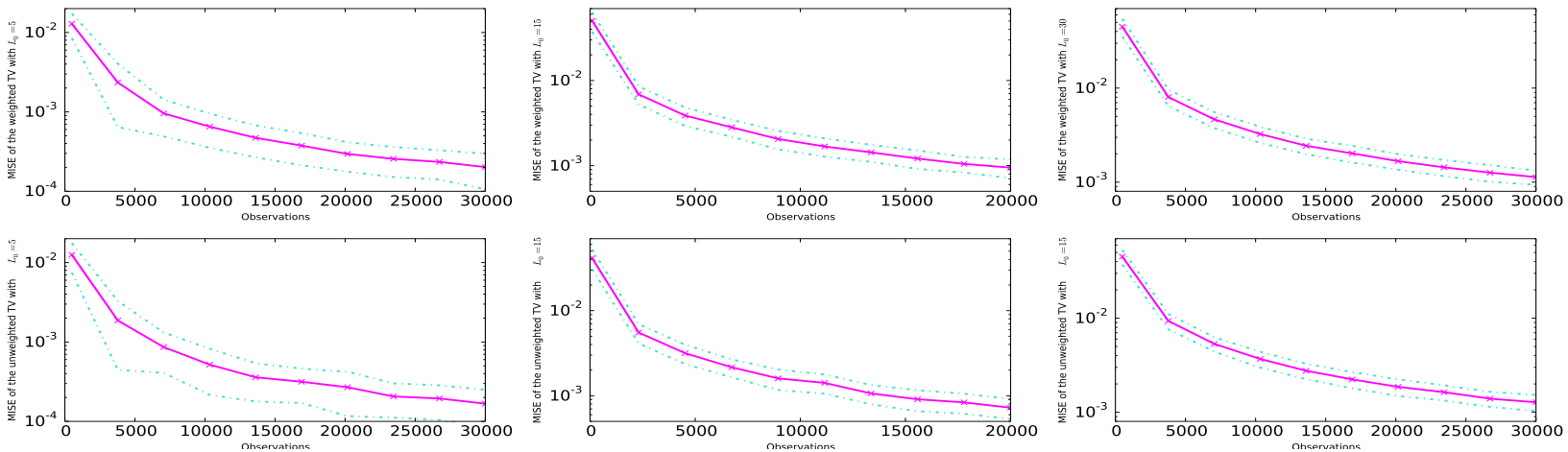


Simulated data

- To evaluate the performance of the TV procedure $\hat{\lambda}$, we use a Monte-Carlo averaged mean integrated squared error MISE.

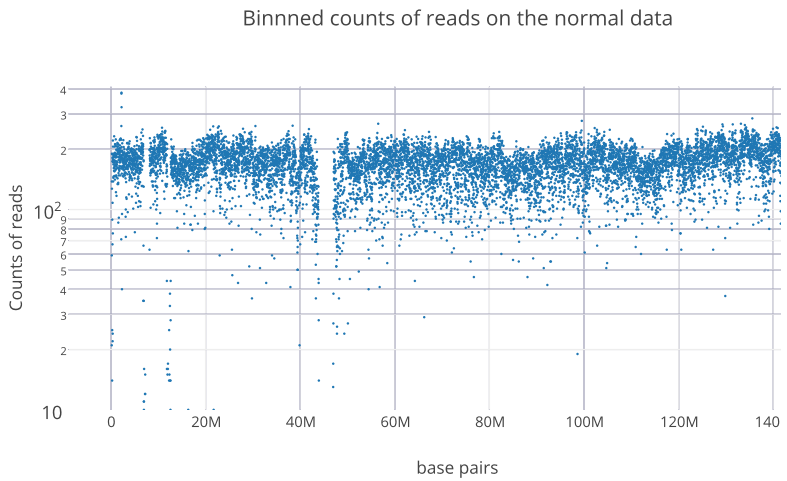
$$\mathrm{MISE}(\hat{\lambda}, \lambda_0) = \mathbb{E} \int_0^1 (\hat{\lambda}(t) - \lambda_0(t))^2 dt.$$

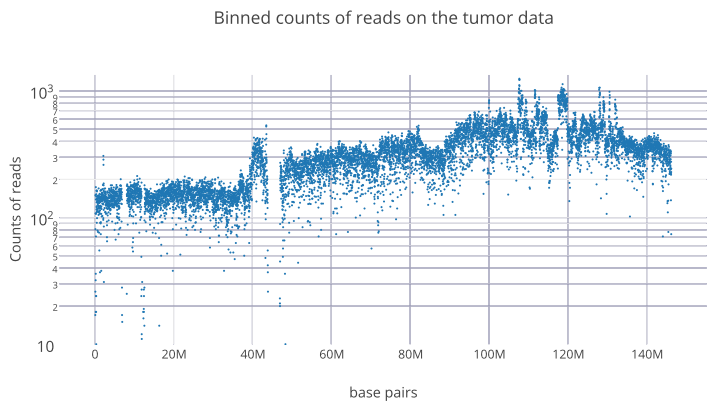
- We run 100 Monte-Carlo experiments, for an increasing sample size between $n = 500$ and $n = 30000$, for each 3 examples.

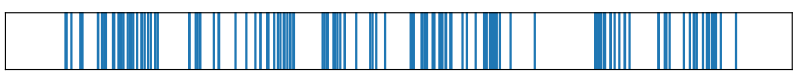


Real data

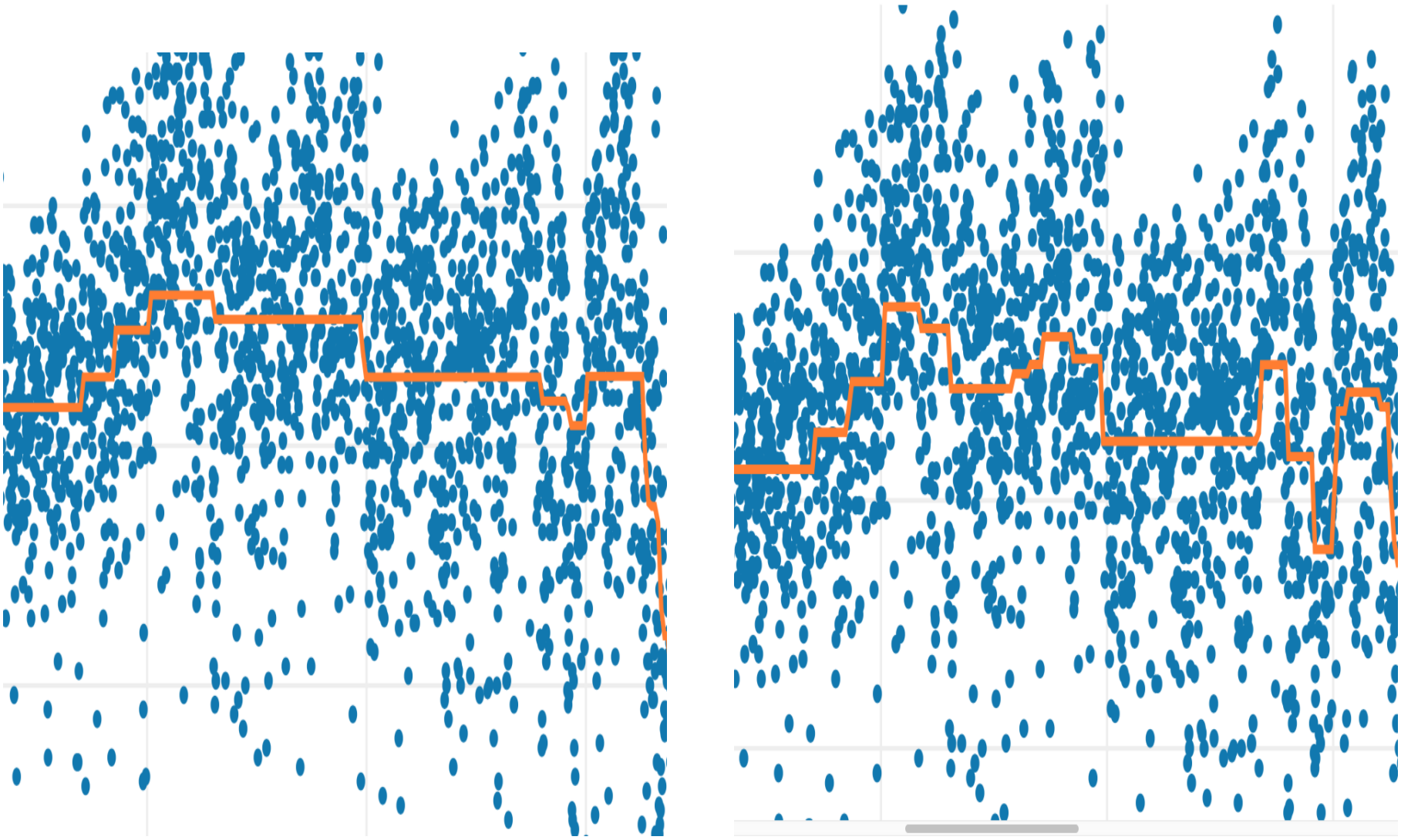
- RNA-seq can be modelled mathematically as replications of an inhomogeneous counting process with a piecewise constant intensity [Shen and Zhang (2012)].
- We applied our method to the sequencing data of the breast tumor cell line HCC1954 7.72 million reads) and its reference cell line BL1954 (6.65 million reads) [Chiang et al. (2009)].





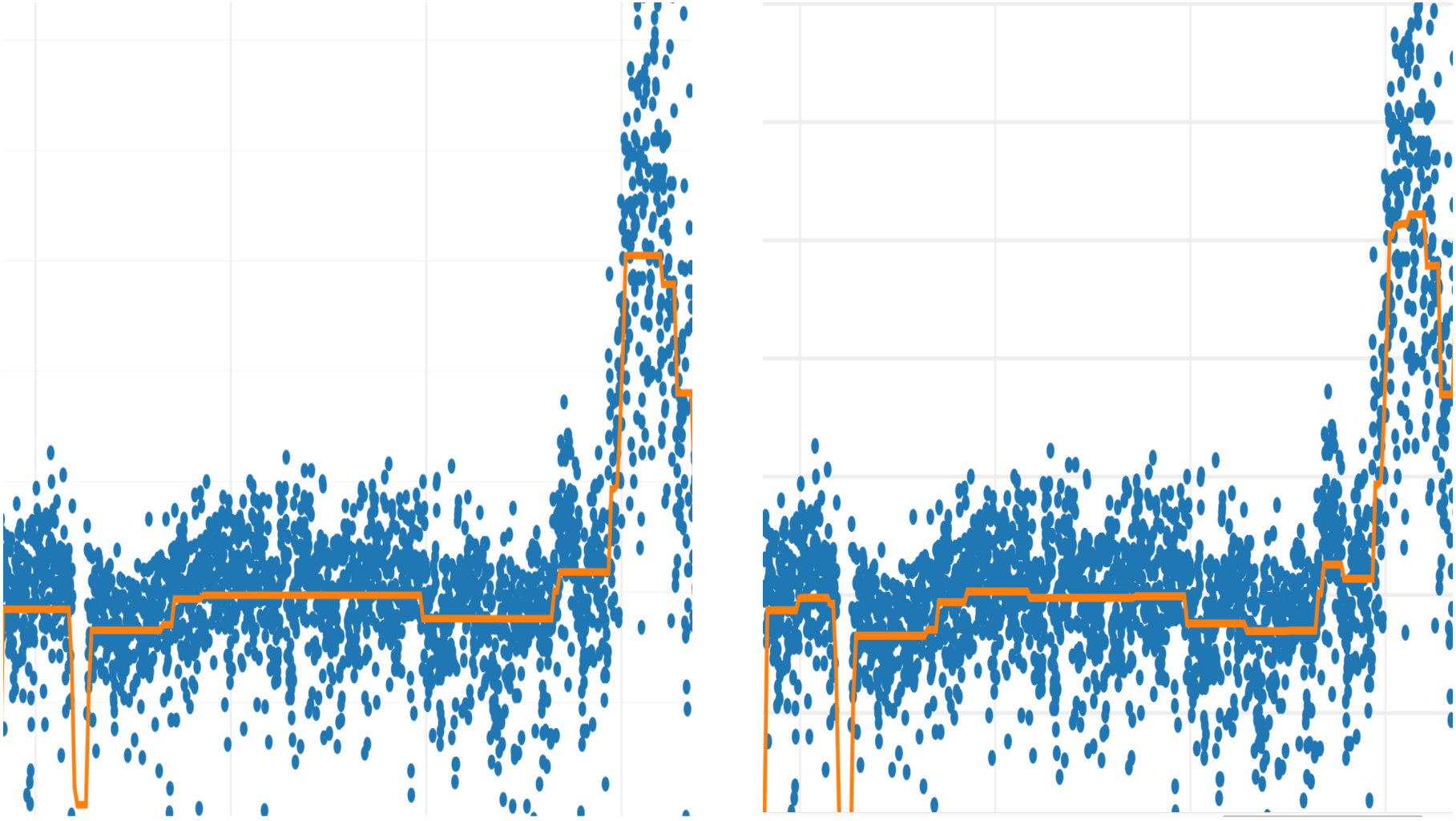


Real data



Zoom into the weighted (left) and unweighted (right) TV estimators applied to the normal data.

Real data



Zoom into the weighted (left) and unweighted (right) TV estimators applied to the tumor data.

Plan

Features binarization

- We have a raw design matrix $\boldsymbol{X} = [\boldsymbol{X}_{i,j}]_{1 \leq i \leq n; 1 \leq j \leq p}$ with n examples and p raw features.
- We denote by $\boldsymbol{X}_{\bullet,j}$ the j-th feature column and by $\boldsymbol{X}_{i,\bullet}$ the i-th data row of $\boldsymbol{X}$.
- The binarized matrix $\boldsymbol{X}^B$ is a matrix with an extended number $d > p$ of columns (only binary).
- The j-th column $\boldsymbol{X}_{\bullet,j}$ is replaced by a number d_j of columns $\boldsymbol{X}^B_{\bullet,j,1}, \ldots, \boldsymbol{X}^B_{\bullet,j,d_j}$ containing only zeros and ones.

Features binarization

- If $X_{\bullet,j}$ takes values (modalities) in the set $\{1, \ldots, M_j\}$ with cardinality M_j, we take $d_j = M_j$, and use a binary coding of each modality by defining

$$\boldsymbol{X}^B_{i,j,k} = \begin{cases} 1, & \text{if } \boldsymbol{X}_{i,j} = k, \\ 0, & \text{otherwise,} \end{cases}$$

- If $X_{\bullet,j}$ is quantitative, then d_j we consider a partition of intervals $I_{j,1}, \ldots, I_{j,d_j}$ for the range of values of $\boldsymbol{X}_{\bullet,j}$ and define

$$\boldsymbol{X}^B_{i,j,k} = \begin{cases} 1, & \text{if } \boldsymbol{X}_{i,j} \in I_{j,k}, \\ 0, & \text{otherwise,} \end{cases}$$

- A natural choice of intervals is given by the quantiles, namely we can typically choose $I_{j,k} = \left(q_j(\frac{k-1}{d_j}), q_j(\frac{k}{d_j})\right]$ for $k = 1, \ldots, d_j$.

Features binarization

- To each binarized feature $\boldsymbol{X}^B_{\bullet,j,k}$ corresponds a parameter $\theta_{j,k}$.
- The parameters associated to the binarization of the j-th feature is denoted $\theta_{j,\bullet} = (\theta_{j,1} \cdots \theta_{j,d_j})^\top$.
- The full parameters vector of size $d = \sum_{j=1}^{p} d_j$, is simply

$$\theta = (\theta_{1,\bullet}^\top \cdots \theta_{p,\bullet}^\top)^\top = \left(\theta_{1,1} \cdots \theta_{1,d_1}\, \theta_{2,1} \cdots \theta_{2,d_2} \cdots \theta_{p,1} \cdots \theta_{p,d_p}\right)^\top.$$

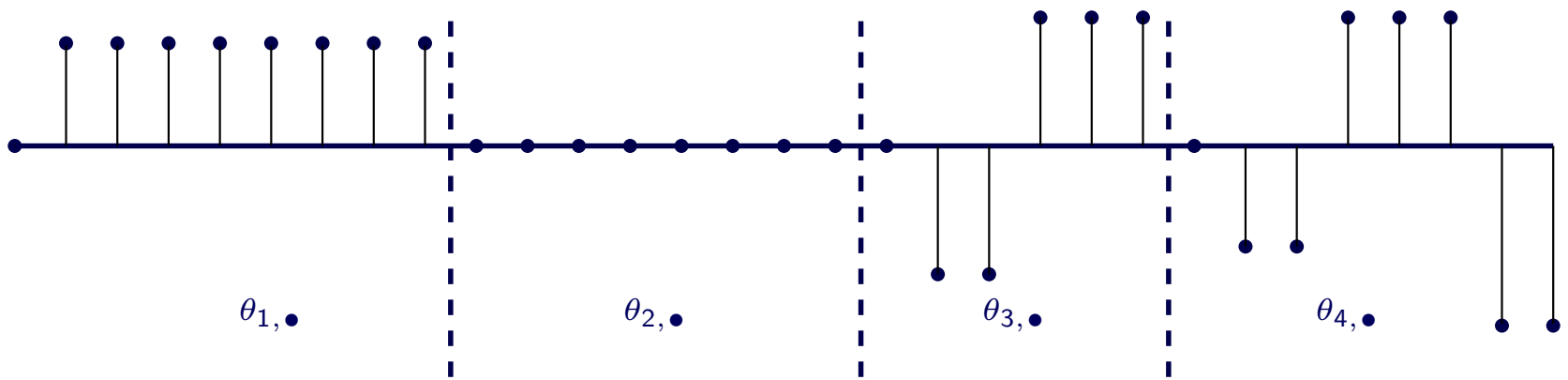


Illustration of $\theta = (\theta_{1,\bullet}^\top \cdots \theta_{p,\bullet}^\top)^\top$ with: $p = 4$, $d_1 = 9$, $d_2 = 8$, $d_3 = 6$, $d_4 = 8$.

Features binarization

- The binarized matrix $\boldsymbol{X}^B$ is not full rank, since in each block the sum of the columns $\boldsymbol{X}^B_{\bullet,j,1}, \ldots, \boldsymbol{X}^B_{i,j,d_j}$ is equal to $\mathbf{1}_n$ (intercept).
- To avoid this over-parametrization, we must add a constraint.
- We can either drop a parameter or add a linear constraint in each bloc $\theta_{j,\bullet}$.
- One sets $\theta_{j,k} = 0$, for one value k in $\{1, \ldots, d_j\}$. This is called a *k^{th}-baseline-constraint*.
- Another useful possibility is to impose $\sum_{k=1}^{d_j} \theta_{j,k} = 0$, called *sum-to-zero-constraint* (**the one we prefer**).

Binarsity

- We therefore introduce the following new penalization called *binarsity*

$$\text{bina}(\theta) = \sum_{j=1}^{p} \Big(\|\theta_{j,\bullet}\|_{\text{TV}} + \delta_{\mathcal{H}_j}(\theta_{j,\bullet}) \Big),$$

  where $\mathcal{H}_j = \{\beta_{j,\bullet} \in \mathbb{R}^{d_j} : \sum_{k=1}^{d_j} \beta_{j,k} = 0\}$, and the indicator function

$$\delta_{\mathcal{H}_j}(\beta_{j,\bullet}) = \begin{cases} 0, & \text{if } \beta_{j,\bullet} \in \mathcal{H}_j, \\ \infty, & \text{otherwise.} \end{cases}$$

- If a raw feature j is statistically not relevant for predicting the labels, then the full block $\theta_{j,\bullet}$ should be zero.
- If a raw feature j is relevant, then the number of different values for the coefficients of $\theta_{j,\bullet}$ should be kept as small as possible, in order to balance bias and variance.

Weighted Binarsity

We consider the following data-driven weighted version of Binarsity given by

$$\mathrm{bina}_{\hat{w}}(\theta) = \sum_{j=1}^{p} \Big(\|\theta_{j,\bullet}\|_{\mathrm{TV},\hat{w}_{j,\bullet}} + \delta_{\mathcal{H}_j}(\theta_{j,\bullet}) \Big)$$

$$\hat{w}_{j,k} \approx C \sqrt{\frac{\log p}{n}} \hat{n}_{j,k},$$

where

$$\hat{n}_{j,k} = \frac{\#\Big(\Big\{ i = 1, \ldots, n : \boldsymbol{X}_{i,j} \in \big(q_j\big(\tfrac{k}{d_j}\big), q_j(1) \big] \Big\} \Big)}{n}.$$

Generalized linear models

- Let a couple of input-output variables $(\boldsymbol{X}, \boldsymbol{Y})$ where the conditional distribution of $\boldsymbol{Y}$ given $\boldsymbol{X} = \mathbf{x}$ is assumed to be from one parameter exponential family

$$f(\mathbf{y}; m_0(\mathbf{x})) = \exp\big(\mathbf{y} m_0(\mathbf{x}) - b(m_0(\mathbf{x}))\big).$$

- The function $b(\cdot)$ is known, while the natural parameter function $m_0(x)$ is *unknown* and specifies how the response depends on the feature.
- We have

$$\mathbb{E}[\boldsymbol{Y}|\boldsymbol{X}] = b'(m_0(\boldsymbol{X})), \text{ and } m_0(\boldsymbol{X}) = g\big(\mathbb{E}[\boldsymbol{Y}|\boldsymbol{X}],\big)$$

  where the dot denotes differentiation and $b' = g^{-1}$ is the *link* function transformation.
- Logistic and probit regression for binary data or multinomial regression for categorical data, Poisson regression for count data, etc ...

Generalized linear models + binarsity

- We consider the empirical risk

$$R_n(m_\theta) = R_n(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell\big(\boldsymbol{Y}_i, m_\theta(\boldsymbol{X}_{i,\bullet})\big) = \frac{1}{n}\sum_{i=1}^{n} \ell\big(\boldsymbol{Y}_i, \langle \boldsymbol{X}^B_{i,\bullet}, \theta\rangle\big).$$

- ℓ is the generalized linear model loss function and is given by

$$\ell\big(\boldsymbol{Y}_i, m_\theta(\boldsymbol{X}_{i,\bullet})\big) = -\boldsymbol{Y}_i\, m_\theta(\boldsymbol{X}_{i,\bullet}) + b(m_\theta(\boldsymbol{X}_{i,\bullet})).$$

- Our estimator of m_0 is given by $\hat{m} = m_{\hat{\theta}}$, where $\hat{\theta}$ is the solution of the penalized log-likelihood problem

$$\hat{\theta} = \underset{\theta\in\mathbb{R}^d}{\operatorname{argmin}} \big\{ R_n(\theta) + \operatorname{bina}_{\hat{w}}(\theta) \big\}.$$

- To evaluate the quality of the estimation, we shall use the *excess risk* of $\hat{m}$,

$$R(\hat{m}(\boldsymbol{X})) - R(m_0(\boldsymbol{X})) = \mathbb{E}_{\mathscr{L}(\boldsymbol{Y}|\boldsymbol{X})}[R_n(\hat{m}(\boldsymbol{X})) - R_n(m_0(\boldsymbol{X}))].$$

- Define the empirical Kullback-Leibler divergence between m_0 and its estimator $\hat{m}$ as follows

$$KL_n(m_0(\boldsymbol{X}), \hat{m}(\boldsymbol{X})) = \frac{1}{n}\sum_{i=1}^{n} KL\big(f(\boldsymbol{Y}; m_0(\boldsymbol{X}_{i,\bullet})), f(\boldsymbol{Y}; \hat{m}(\boldsymbol{X}_{i,\bullet}))\big).$$

- One has

Lemma

$$R(\hat{m}(\boldsymbol{X})) - R(m_0(\boldsymbol{X})) = KL_n(m_0(\boldsymbol{X}), \hat{m}(\boldsymbol{X})).$$

Oracle inequality

Theorem 3

Assume that $\boldsymbol{Y}_i - m_0(\boldsymbol{X}_{i,\bullet})$ is a subgaussian random variable. Then, with a probability larger than $1 - p^{1-A}, (A > 1)$ the estimator $\hat{m}$ verifies

$$KL_n(m_0(\boldsymbol{X}), \hat{m}(\boldsymbol{X})) \leq \inf_{\theta \in \mathbb{R}^d} \left(KL_n(m_0(\boldsymbol{X}), m_\theta(\boldsymbol{X})) + 2\,\mathrm{bina}_{\hat{w}}(\theta) \right).$$

The variance term satifies

$$\mathrm{bina}_{\hat{w}}(\theta) \approx \mathrm{bina}(\theta) \max_{j=1,\dots,p} \max_{k=1,\dots,d_j} \sqrt{\frac{\log p}{n}}.$$

Proximal algorithm of weighted binarsity

- Since Binarsity is separable by blocks, we have

$$\left(\operatorname{prox}_{\text{bina}_{\hat{w}}}(\theta)\right)_{j,\bullet} = \operatorname{prox}_{(\|\cdot\|_{\text{TV},\hat{w}_{j,\bullet}} + \delta_{\mathcal{H}_j})}(\theta_{j,\bullet}),$$

  for all $j = 1, \ldots, p$.

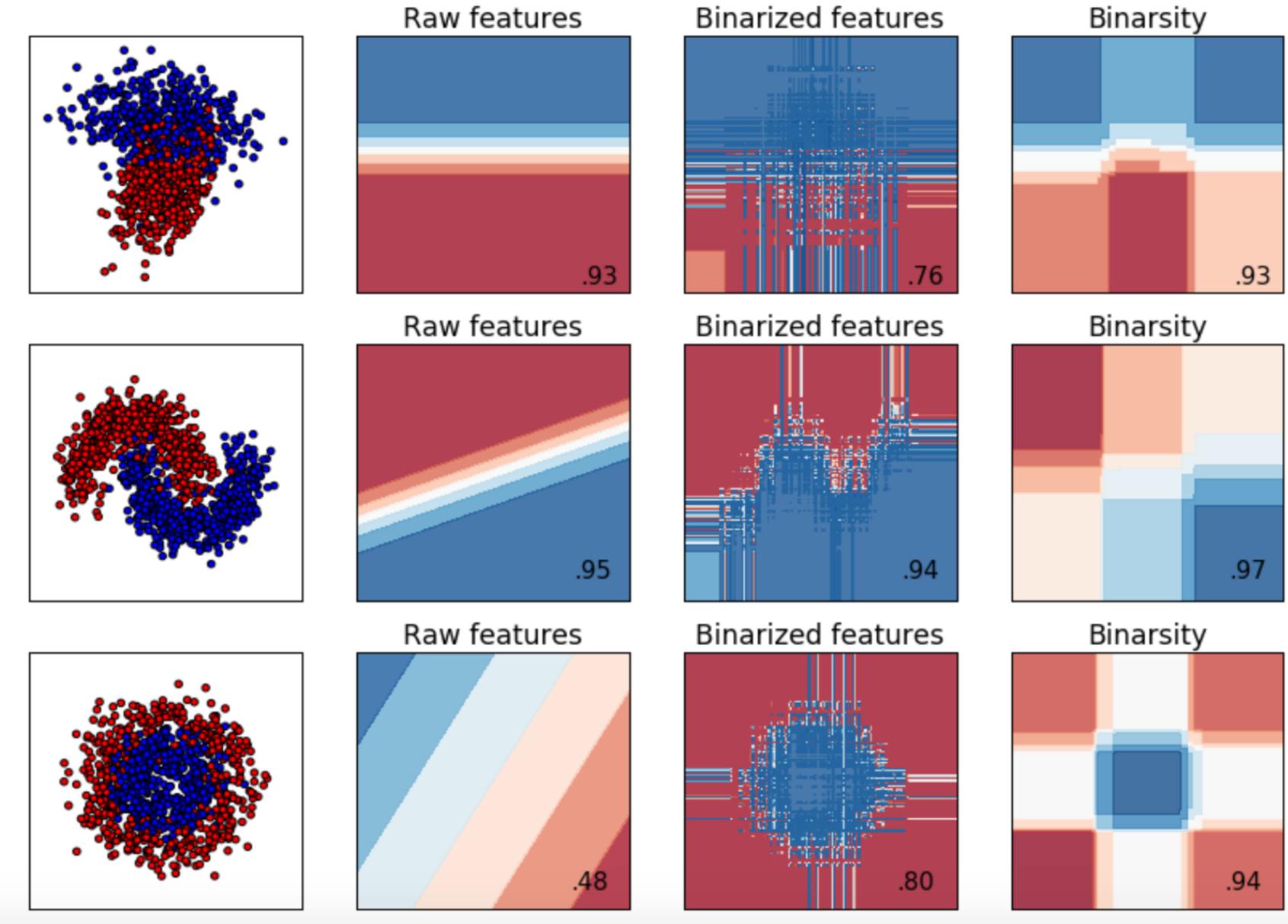
- Algorithm 2 expresses $\operatorname{prox}_{\text{bina}_{\hat{w}}}$ based on the proximal operator of the weighted TV penalization.

Algorithm 2: $\operatorname{prox}_{\text{bina}_{\hat{w}}}(\theta)$

for $j = 1, \ldots, p$ **do**
  $\beta_{j,\bullet} \leftarrow \operatorname{prox}_{\|\cdot\|_{\text{TV},\hat{w}_{j,\bullet}}}(\theta_{j,\bullet})$;
  $\eta_{j,\bullet} \leftarrow \beta_{j,\bullet} - \frac{1}{d_j}\sum_{k=1}^{d_j} \beta_{j,k}$;
  return $\eta_{j,\bullet}$;

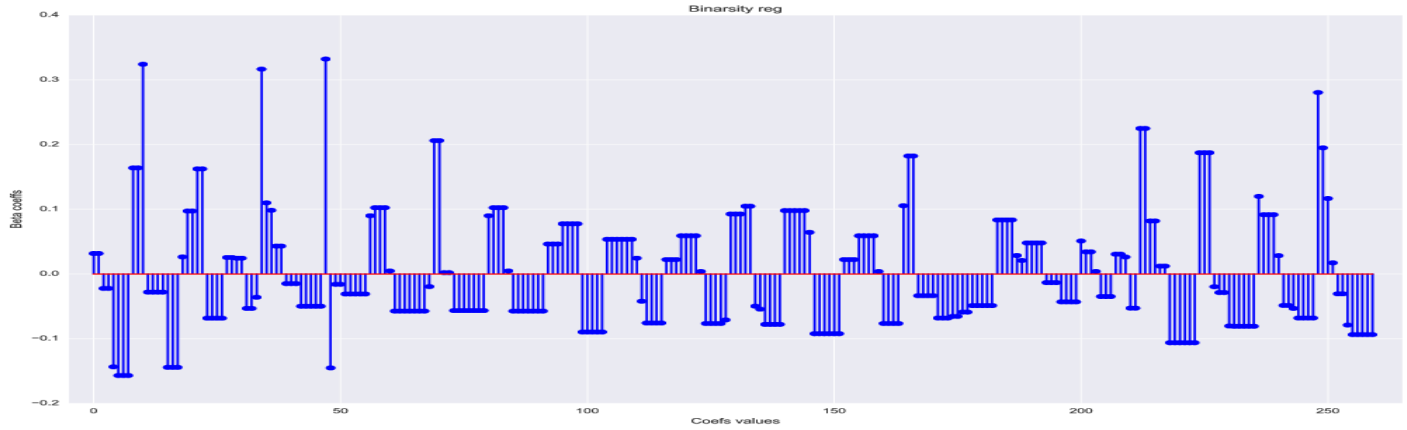
- TV regularization and mean removal in each block.

Toy example ($n = 1000$, $p = 2$, n_cuts =100)



Rela data: Parkinsons dataset ($n = 130, p = 22$)

[Source: https://archive.ics.uci.edu/ml/datasets/Parkinsons]



Algorithms	AUC	n_cuts
log reg on std features, no *pen*	0.851	-
log reg std features, ℓ_2-*pen*	0.839	-
log reg std features, ℓ_1-*pen*	0.878	-
log reg on bina features, bina-*pen*	**0.901**	12

Plan

Framework and models: marked counting process

For individual $i, i = 1, \ldots, n$:

- $N_i(t)$ is a marked counting process over a fixed time interval $[0, \tau]$, with marker $Y_i(t)$.
- N_i has intensity, namely

$$\mathbb{P}[N_i \text{ has a jump in } [t, t+dt) | \mathcal{F}_t] = Y_i(t) \lambda_\star(t, X_i(t)) dt,$$

  where $\mathcal{F}_t = \sigma(N_i(s), Y_i(s), X_i(s) : s \leq t)$.
- $X_i(t) = (X_i^1(t), \ldots, X_i^p(t))$ are temps-dependents covariables.
- High-dimensional setting: p is large.

Framework and models: dynamic Aalen and Cox models

- We consider two dynamic models for the function $\lambda_\star$:
  - a time-varying Aalen model

$$\lambda_\star^{\mathsf{A}}(t, X(t)) = X(t)\beta^\star(t),$$

  - a time-varying Cox model

$$\lambda_\star^{\mathsf{M}}(t, X(t)) = \exp\big(X(t)\beta^\star(t)\big),$$

- $\beta^\star$ is an unknown p-dimensional function from $[0, \tau]$ to $\mathbb{R}^p$.
- Aim to estimate the parameter $\beta^\star$ on the basis of data from n independent individuals:

$$\mathcal{D}_n = \big\{(X_i(t), Y_i(t), N_i(t)) : t \in [0, \tau], i = 1, \ldots, n\big\}.$$

Learning dictionary: piecewise constant functions

- We consider sieves (or histogram) based estimators of the p-dimensional unknown function $\beta^\star$ [Murphy and Sen (1991)].
- We hence consider a L-partition of the time interval $[0, \tau]$, where $L \in \mathbb{N}^*$

$$\varphi_l = \sqrt{\frac{L}{\tau}} \mathbb{1}(I_l) \text{ and } I_l = \Big(\frac{l-1}{L}\tau, \frac{l}{L}\tau \Big].$$

- Let the set of univariate piecewise constant functions

$$\mathcal{H}_L = \Big\{ \alpha(\cdot) = \sum_{l=1}^{L} \alpha_l \varphi_l(\cdot) : (\alpha_l)_{1 \leq l \leq L} \in \mathbb{R}_+^L \Big\}.$$

Learning dictionary: piecewise constant functions

- We define the sets of candidates for estimation as

$$\Lambda^{\mathsf{A}} = \{x, t \in [0, \tau] \mapsto \lambda^{\mathsf{A}}_{\beta}(t, x(t)) = x(t)\beta(t) \mid \forall j \ \beta_j \in \mathcal{H}_L\}.$$

  for the Aalen model and

$$\Lambda^{\mathsf{M}} = \{x, t \in [0, \tau] \mapsto \lambda^{\mathsf{M}}_{\beta}(t, x(t)) = \exp\left(x(t)\beta(t)\right) \mid \forall j \ \beta_j \in \mathcal{H}_L\}.$$

  for the Cox model.

- We consider

$$\beta = (\beta_{1,.}^{\top}, \ldots, \beta_{p,.}^{\top})^{\top} = (\beta_{1,1}, \ldots, \beta_{1,L}, \ldots, \beta_{p,1}, \ldots, \beta_{p,L})^{\top},$$

$$\forall j = 1 \ldots, p, \ \forall l = 1, \ldots, L. \text{ and } \forall t \in I_l, \beta_j(t) = \sqrt{\frac{L}{\tau}}\beta_{j,l}.$$

Estimation procedure: weighted $(\ell_1 + \ell_1)$-TV penalization

- Full likelihood functional: time-varying Cox model

$$\ell_n^{\mathrm{M}}(\beta) = -\frac{1}{n}\sum_{i=1}^{n}\Big\{\int_0^\tau \log\left(\lambda_\beta^{\mathrm{M}}(t, X_i(t))\right) dN_i(t) - \int_0^\tau Y_i(t)\lambda_\beta^{\mathrm{M}}(t, X_i(t))dt\Big\}.$$

  [Martinussen and Scheike (2007), Lemler (2013)].

- Our specific covariate weighted $(\ell_1 + \ell_1)$-TV penalty is given by

$$\|\beta\|_{\mathrm{gTV},\hat{w}} = \sum_{j=1}^{p}\left(\hat{w}_{j,1}|\beta_{j,1}| + \sum_{l=2}^{L}\|\beta_{j,\cdot}\|_{\mathrm{TV},\hat{w}_{j,\cdot}}\right), \text{ for } \beta \in \mathbb{R}^{pL}.$$

$$\hat{w}_{j,l} \approx C_\tau \sqrt{\frac{L\log(pL)}{n}\int_{(l-1)\tau/L}^{\tau}(X_i^j(t))^2 d\bar{N}_n(t)}.$$

Slow oracle inequality

- In the Cox model, our estimator is then respectively defined as $\hat{\lambda}^{\mathrm{M}} = \lambda^{\mathrm{M}}_{\hat{\beta}^{\mathrm{M}}}$, where

$$\hat{\beta}^{\mathrm{M}} = \underset{\beta \in \mathbb{R}^{p \times L}}{\operatorname{argmin}} \left\{ \ell_n^{\mathrm{M}}(\beta) + \|\beta\|_{\mathrm{gTV},\hat{w}} \right\}.$$

Theorem 4

For $x > 0$ fixed, the estimator $\hat{\lambda}^{\mathrm{M}}$ verifies with a probability larger than $1 - C_{\mathrm{M}} e^{-x}$,

$$K_n(\lambda^{\mathrm{M}}_{\star}, \hat{\lambda}^{\mathrm{M}}) \leq \inf_{\beta \in \mathbb{R}^{pL}} \left(K_n(\lambda^{\mathrm{M}}_{\star}, \lambda^{\mathrm{M}}_{\beta}) + 2||\beta||_{\mathrm{gTV},\hat{w}} \right).$$

- The variance term satisfies

$$\|\beta\|_{\mathrm{gTV},\hat{w}} \approx \|\beta\|_{\mathrm{gTV}} \max_{j=1,\dots,p} \max_{l=1,\dots,L} \sqrt{\frac{L \log pL}{n}}.$$

Proximal operator of the weighted $(\ell_1 + \ell_1)$-TV penalization

- $\theta = \text{prox}_{\|\cdot\|_{\text{gTV},\hat{w}}}(\beta)$

$$\theta = \underset{x \in \mathbb{R}^{pL}}{\text{argmin}} \left\{ \frac{1}{2}\|\beta - x\|_2^2 + \sum_{j=1}^{p} \left(\hat{w}_{j,1}|x_{j,1}| + \sum_{l=2}^{L} \|x_{j,.}\|_{\hat{w}_{j,.}} \right) \right\}.$$

Algorithm 3: $\theta = \text{prox}_{\|\cdot\|_{\text{gTV},\hat{w}}}(\beta)$

for $j = 1, \ldots, p$ **do**
  set $\mu \leftarrow \beta_{j,.}$; $\hat{\gamma} \leftarrow \hat{w}_{j,.} \setminus \{\hat{w}_{j,1}\}$;
  $\eta \leftarrow \text{prox}_{\|\cdot\|_{\text{TV},\hat{\gamma}}}(\mu)$;
  $\theta_{j,.} \leftarrow \eta - \left(\eta_1 - \text{sgn}(\eta_1) \max\left(0, |\eta_1| - \frac{\hat{\gamma}_1}{L}\right) \right) \mathbf{1}_L$;
  return $\theta_{j,.}$

- TV regularization and thresholding in each bloc.

.

Algorithm 4: SGD-timevar

1. Parameters: Integer $K > 0$;
2. Initialization: $(\hat{\beta})^{(1)} = 0 \in \mathbb{R}^{p \times L}$, and $r^{(1)} \in [0, 1]$;
3. **for** $k = 1, \ldots, K$ **do**

   Choose randomly $i_k \in \{1, \ldots, n\}$ and compute $\nabla_{i_k} = \nabla \ell_{i_k}((\hat{\beta})^{(k)})$;

   Update moving averages

   $a^{(k)} \leftarrow (1 - (r^{(k)})^{-1}) a^{(k)} + (r^{(k)})^{-1} \nabla_{i_k}$;

   $b_j^{(k)} \leftarrow (1 - (r^{(k)})^{-1}) b_j^{(k)} + (r^{(k)})^{-1} \| \nabla_{j,\cdot} \|^2$;

   $c^{(k)} \leftarrow (1 - (r^{(k)})^{-1}) c^{(k)} + (r^{(k)})^{-1} \operatorname{diag}(H_{i_k})$ where $H_{i_k} = \left(\frac{\partial^2 \left(\ell_{i_k}((\hat{\beta})^{(k)}) \right)}{\partial^2 \beta} \right)$;

   Estimate learning rate

   $\varepsilon_j^{(k)} \leftarrow \frac{1}{c_j^+} \frac{\sum_{l=1}^L (a_{j,l}^{(k)})^2}{b_j^{(k)}}$; where $c_j^+ = \max_{1 \le l \le L} c_{j,l}$

   $\eta_j \leftarrow \varepsilon_j^{(k)}$;

   $\varepsilon^{(k)} \leftarrow (\varepsilon_1^{(k)} \mathbf{1}_L, \ldots, \varepsilon_p^{(k)} \mathbf{1}_L)^\top$;

   Update memory size

   $r^{(k)} \leftarrow (1 - \frac{\sum_{l=1}^L (a_{j,l}^{(k)})^2}{b_j^{(k)}}) \odot r^{(k)} + 1$;

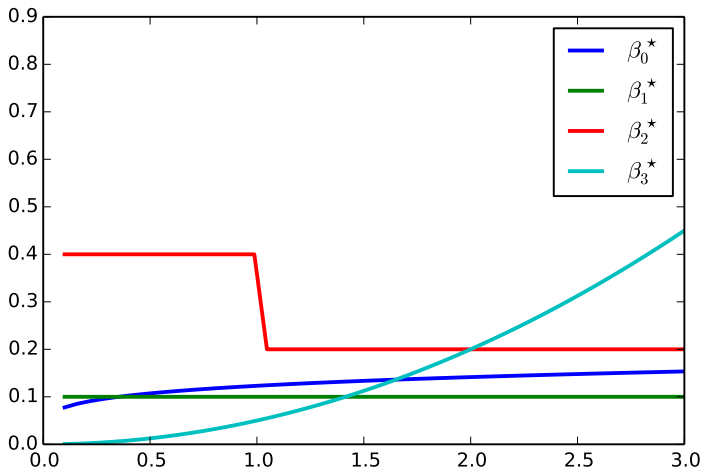
   $(\hat{\theta})^{(k)} \leftarrow (\hat{\beta})^{(k)} - \varepsilon^{(k)} \odot \nabla_{i_k}$;

   $(\hat{\beta})^{(k+1)} \leftarrow \left(\operatorname{prox}_{\eta_1 \| \cdot \|_{\mathrm{gTV}, \hat{w}_{1,\cdot}}} ((\hat{\theta})_{1,\cdot}^{(k)}), \ldots, \operatorname{prox}_{\eta_p \| \cdot \|_{\mathrm{gTV}, \hat{w}_{p,\cdot}}} ((\hat{\theta})_{p,\cdot}^{(k)}) \right)^\top$;
4. **return** $(\hat{\beta})^{(K)}$

[Schaul et al. (2012)].

Simulated data in the time-varying Cox model

- Right censoring: $n = 1000$, and T has hazard rate $\lambda_\star(t, X) = \beta_0^\star(t) \exp(X(t)\beta^\star(t))$.
- $p = 10$ covariates processes $X_i(t)_{i=1,\dots,n}$ which are $\mathcal{N}(0, 0.5)$ $i.i.d$ piecewise constant over a 50-partition of the time interval $[0, 3]$.
- The baseline $\beta_0^\star$ is defined through a Weibull $\mathcal{W}(1.2, 0.15)$.
- We draw the true regression functions $\beta_1^\star$, $\beta_2^\star$, and $\beta_3^\star$. We set $\beta_j^\star \equiv 0$, for $j = 4, \dots, 10$.



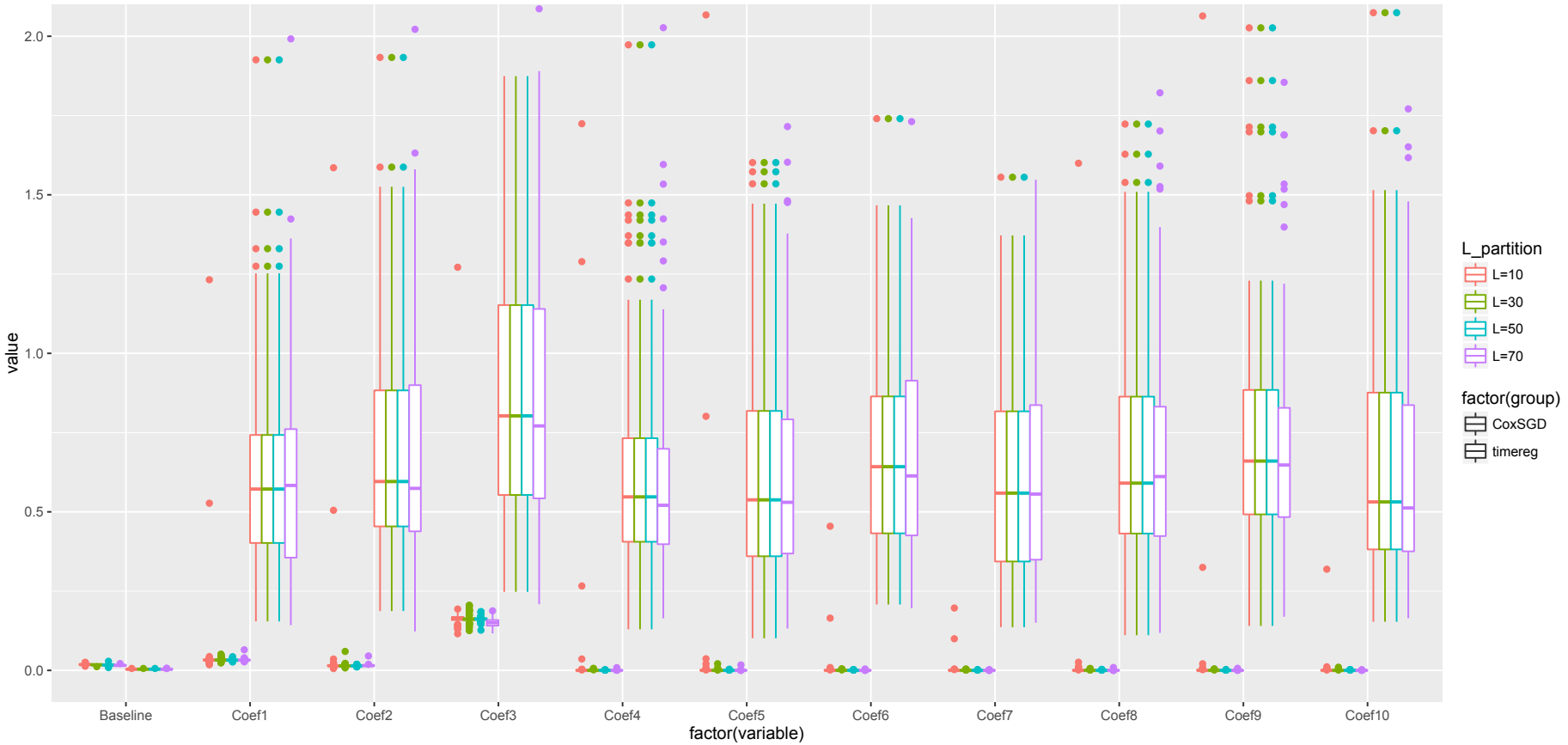
Simulated data in the time-varying Cox model

- We run 100 Monte-Carlo experiments of training data as described above.
- The estimation accuracy is investigated via a mean squared error defined as

$$\mathrm{MISE}_j = \frac{1}{100} \sum_{m=1}^{100} \int_0^{\tau} \left((\hat{\beta}_j^{\mathsf{M}}(t))_m - \beta_j^{\star}(t) \right)^2 dt,$$

where $(\hat{\beta}_j^{\mathsf{M}}(t))_m$ is the estimation of $\beta_j^{\star}(t)$ in the sample m, for all $j = 1, \ldots, p$.

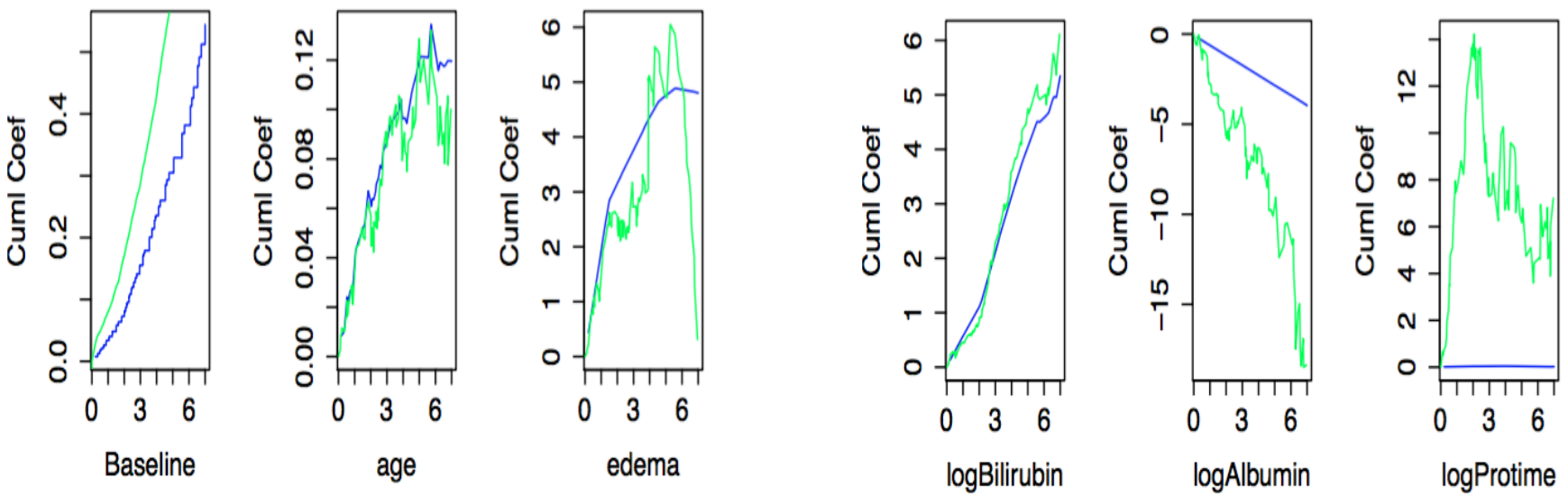
Simulated data in the time-varying Cox model



Boxplots of the MISE_j of estimated regression coefficients over L-partition ($L \in \{10, 30, 50, 70\}$) with `SGD-timevar` (left) and `timereg` R-package (right) [Martinussen and Scheike (2007)].

PBC data: time-varying Cox model

- Primary Biliary Cirrhosis (PBC) of the liver and was conducted between 1974 and 1984 [Feleming (1991)].
- A total of 418 patients are included in the dataset and were followed until death or censoring.
- We consider the covariates: age, edema, log(bilirubin), log(albumin) and log(protime).



Estimated cumulative regression coefficients $\hat{B}_j^{\mathsf{M}}(t) = \int_0^t \hat{\beta}_j(s)ds$ on PBC data with: `SGD-timevar` (blue) and `timereg` R-package (green).

Plan

Conclusion + Perspectives

- We introduce a data-driven weighted total-variation penalizations for three problems: change-points detection, generalized linear models with binarized features and learning high-dimensional time-varying Aalen and Cox models.
- For each procedure, we give: theoretical guaranties by proving oracles inequalities for the prediction error and algorithms that efficiently solve the studied convex problems.

With S. Bussy and A. Guilloux, we study the estimation problem of high-dimensional Cox model, with covariables having multiple cut-points, using binarsity penalization.

Comparing numerically the prediction performance of binarsity with others procedures (random forests).

References

Beck, A. and M. Teboulle (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences 2*(1), 183–202.

Chambolle, A., V. Caselles, D. Cremers, M. Novaga, and T. Pock (2010). An introduction to total variation for image analysis. *Theoretical foundations and numerical methods for sparse recovery 9*, 263–340.

Chiang, D. Y., G. Getz, D. B. Jaffe, M. J. T. O'Kelly, X. Zhao, S. L. Carter, C. Russ, C. Nusbaum, M. Meyerson, and E. S. Lander (2009). High-resolution mapping of copy-number alterations with massively parallel sequencing. *Nature methods 6*(1), 99–103.

Condat, L. (2013). A Direct Algorithm for 1D Total Variation Denoising. *IEEE Signal Processing Letters 20*(11), 1054–1057.

Daubechies, I., M. Defrise, and C. De Mol (2004). An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics 57*(11), 1413–1457.

Gaïffas, S. and A. Guilloux (2012). High-dimensional additive hazards models and the lasso. *Electron. J. Statist. 6*, 522–546.

Harchaoui, Z. and C. Lévy-Leduc (2010). Multiple change-point estimation with a total variation penalty. *J. Amer. Statist. Assoc. 105*(492), 1480–1493.

Lemler, S. (2013). Oracle inequalities for the lasso in the high-dimensional multiplicative aalen intensity model. *Les Annales de l'Institut Henri Poincaré, arXiv preprint*.

Martinussen, T. and T. H. Scheike (2007). *Dynamic regression models for survival data*. Springer Science & Business Media.

Murphy, S. A. and P. K. Sen (1991). Time-dependent coefficients in a cox-type regression model. *Stochastic Processes and their Applications 39*(1), 153–180.

Reynaud-Bouret, P. (2003). Adaptive estimation of the intensity of inhomogeneous Poisson processes via concentration inequalities. *Probab. Theory Related Fields 126*(1), 103–153.

Reynaud-Bouret, P. (2006). Penalized projection estimators of the Aalen multiplicative intensity. *Bernoulli 12*(4), 633–661.

Schaul, T., S. Zhang, and Y. LeCun (2012). No more pesky learning rates. *arXiv preprint arXiv:1206.1106*.

Shen, J. J. and N. R. Zhang (2012). Change-point model on nonhomogeneous Poisson processes with application in copy number profiling by next-generation DNA sequencing. *Ann. Appl. Stat. 6*(2), 476–496.

Tibshirani, R., M. Saunders, S. Rosset, J. Zhu, and K. Knight (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67*(1), 91–108.