Fused LASSO and change point detection

Paul Seidel

29/10/2018

Outline

The idea

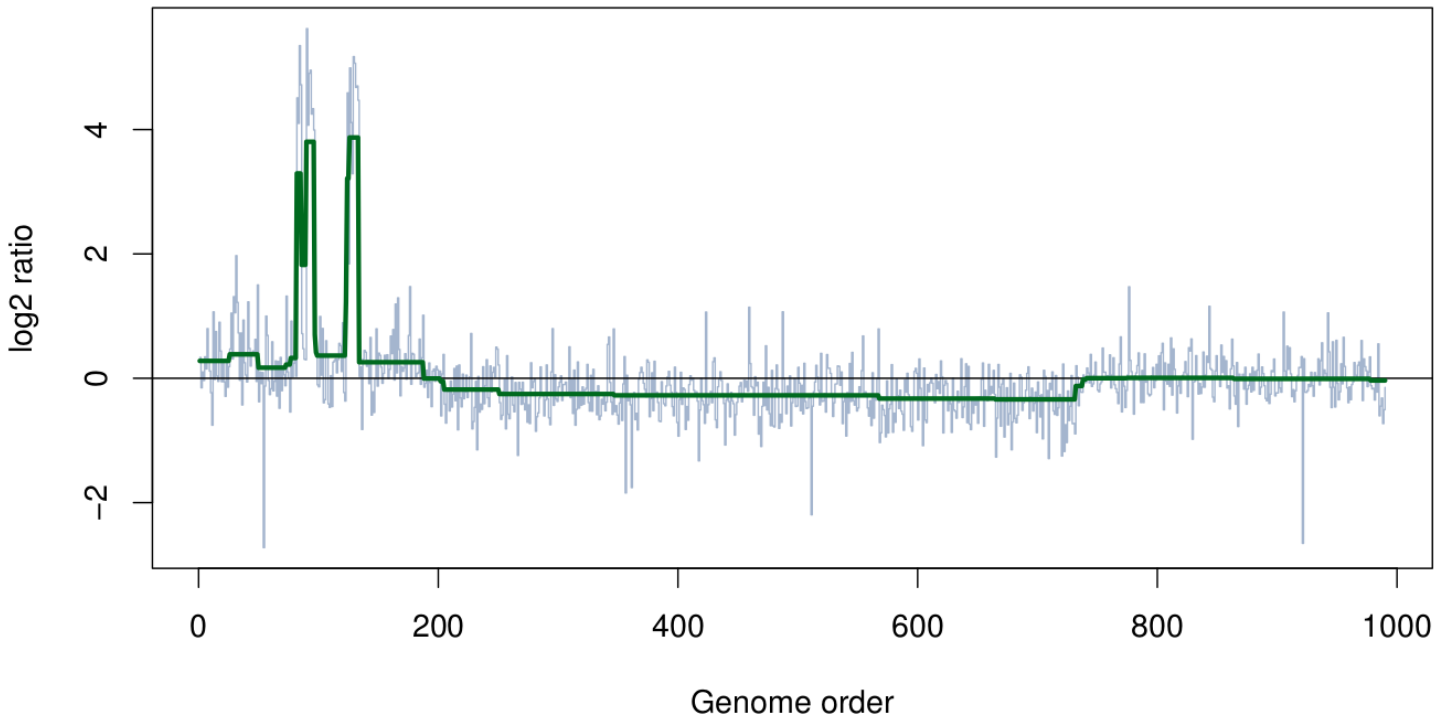


Figure 4.8 *Fused lasso applied to CGH data. Each spike represents the copy number of a gene in a tumor sample, relative to that of a control (on the log base-2 scale). The piecewise-constant green curve is the fused lasso estimate.*

Hastie, Tibshirani and Wainwright. Statistical learning with sparsity: The Lasso and generalizations, CRC Press, 2015, page 76

The fused LASSO

The fused LASSO solves the problem

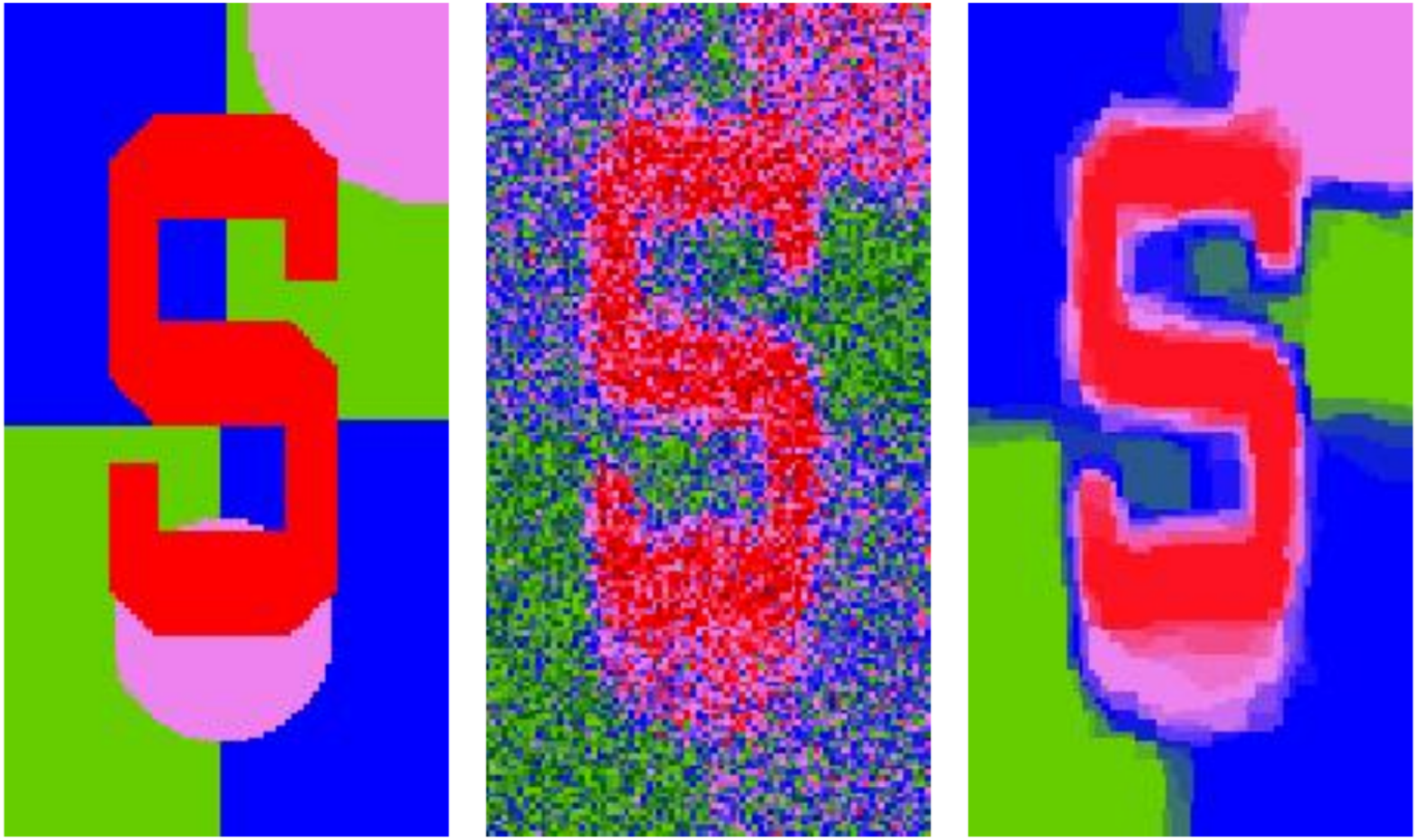
$$\underset{\theta\in\mathbb{R}^n}{\text{minimize}}\left\{ \sum_{i=1}^{n}(Y_i-\theta_i)^2+\lambda_1\sum_{i=1}^{n}|\theta_i|+\lambda_2\sum_{i=2}^{n}|\theta_i-\theta_{i-1}| \right\}.$$

More generally one can use the penalty

$$\lambda_2\sum_{i\sim j}|\theta_i-\theta_j|,$$

where $\sim$ is a relation depending on the problem at hand. For example:

A 2-dimensional example



(a) Original (b) Noisy (c) Denoised

Tibshirani, Ryan J.; Taylor, Jonathan. The solution path of the generalized lasso. Ann. Statist. 39 (2011), no. 3, 1335–1371

Simplifying the form of the fused LASSO

Both of these problems can be simplified due to the following result:

Lemma
Let $\hat{\theta}(\lambda_1, \lambda_2)$ be the fitted values for the fused LASSO. For any $\lambda_1' > \lambda_1$ and each $i = 1, \ldots, n$:

$$\hat{\theta}_i(\lambda_1', \lambda_2) = \mathcal{S}_{\lambda_1' - \lambda_1}(\hat{\theta}_i(\lambda_1, \lambda_2)),$$

where $\mathcal{S}_\lambda(z) = \text{sign}(z)(|z| - \lambda)_+$.

In particular

$$\hat{\theta}_i(\lambda_1, \lambda_2) = \mathcal{S}_{\lambda_1}(\hat{\theta}_i(0, \lambda_2)).$$

Simplifying the form of the fused LASSO

$$\hat{\theta}_i(\lambda_1, \lambda_2) = \mathcal{S}_{\lambda_1}(\hat{\theta}_i(0, \lambda_2)).$$

Thus only

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \left\{ \sum_{i=1}^{n} (Y_i - \theta_i)^2 + \lambda \sum_{i=2}^{n} |\theta_i - \theta_{i-1}| \right\}$$

remains.

This is a convex optimization problem. However, coordinate descent does not work because the penalty term is not separable!

(Recall that separability requires the form $f(\theta) = g(\theta) + \sum_{i=1}^{n} h_i(\theta_i)$, with g convex and differentiable and h_i convex.)

Reparametrization

Let

$$M = \begin{pmatrix} 1 & & & & \\ -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{pmatrix} \in \mathbb{R}^{n \times n}.$$

Reparametrization

Then we rewrite our problem

$$\underset{\theta\in\mathbb{R}^n}{\text{minimize}}\left\{\sum_{i=1}^{n}(Y_i-\theta_i)^2+\lambda\sum_{i=2}^{n}|\theta_i-\theta_{i-1}|\right\}$$

in an equivalent way as

$$\underset{\theta\in\mathbb{R}^n}{\text{minimize}}\left\{\|Y-\theta\|_2^2+\lambda\|M\theta\|_1\right\}=\underset{\theta'\in\mathbb{R}^n}{\text{minimize}}\left\{\|Y-M^{-1}\theta'\|_2^2+\lambda\|\theta'\|_1\right\}.$$

Note that this actually produces an excess $|\theta_1'|=|\theta_1|$ term, which we do not want to penalize. This is rectified by writing it as a LASSO problem with distinct regularization parameters for the coefficients, set the first one to 0 and all others to λ. We will not explicitly indicate this in the notation.

Reparametrization

In this form one could use methods like LARS or coordinate descent to solve the LASSO and obtain a solution for the fused LASSO.

However,

$$M^{-1} = \begin{pmatrix} 1 & & & & \\ 1 & 1 & & & \\ \vdots & \vdots & \ddots & & \\ 1 & & \dots & 1 & \\ 1 & 1 & \dots & 1 & 1 \end{pmatrix},$$

and this design matrix has high correlation amongst the predictors. Under this condition coordinate descent and LARS do not perform as well as other methods.

The monotone fusion property

The solutions to the fused LASSO as a function of the regularization parameter have a nice property, which can also be exploited for algorithmic purposes.

Lemma
Let $\hat{\theta}(\lambda)$ denote the solution to the convex optimization problem for the fused LASSO. If some λ and $1 \leq i < n$ have the property that $\hat{\theta}_i(\lambda) = \hat{\theta}_{i+1}(\lambda)$ then for any $\lambda' > \lambda$ we have $\hat{\theta}_i(\lambda') = \hat{\theta}_{i+1}(\lambda')$ as well.

This does not hold for generalizations like the 2-dimensional example.

Dynamic Programming

The minimization problem can be solved using dynamic programming. In the case of the fused lasso Johnson[3] proposed such an algorithm with a runtime of $O(n)$.

All terms which depend on θ_1 are split off:

$$(Y_1 - \theta_1)^2 + \lambda|\theta_2 - \theta_1| + \left\{ \sum_{i=2}^{n}(Y_i - \theta_i)^2 + \lambda \sum_{i=3}^{n}|\theta_i - \theta_{i-1}| \right\}.$$

Then one can write $\hat{\theta}_1 = \hat{\theta}_1(\theta_2)$ as a function of θ_2. This can be iterated to obtain $\hat{\theta}_i(\theta_{i+1})$ and finally $\hat{\theta}_n$. The other $\hat{\theta}_i$ are computed using the recursive functions.

[3] Nicholas A. Johnson (2013) A Dynamic Programming Algorithm for the Fused Lasso and L_0-Segmentation, Journal of Computational and Graphical Statistics

Trend filtering

The more general penalty

$$\lambda \sum_{i \sim j} |\theta_i - \theta_j|$$

can be rewritten (and generalized) in the form $\|D\theta\|_1$ for an appropriate $m \times n$ matrix D, where m is the number of relations in the penalty. As a special case, we consider the so-called k-th differences.

Trend filtering

Define

$$D_0^{(k)} = \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{pmatrix} \in \mathbb{R}^{(n-k-1)\times(n-k)}$$

and inductively

$$D_k = D_0^{(k)} D_{k-1} \in \mathbb{R}^{(n-k-1)\times n} \text{ for } k \geq 1.$$

(Where $D_0 = D_0^{(0)}$.) Note that $\lambda \| D_0^{(0)} \theta \|_1$ is the usual fused LASSO penalty.

Then

$$D_1 = \begin{pmatrix} 1 & -2 & 1 & & \\ & 1 & -2 & 1 & \\ & & \ddots & \ddots & \ddots \\ & & & 1 & -2 & 1 \end{pmatrix}.$$

Trend filtering

D_1 gives the penalty

$$\lambda \sum_{i=3}^{n} |\theta_{i-2} - 2\theta_{i-1} + \theta_i|.$$

This encourages three consecutive points to be on a line because

$$|\theta_{i-2} - 2\theta_{i-1} + \theta_i| = 0 \iff \theta_{i-1} = \frac{\theta_{i-2} + \theta_i}{2}.$$

Similarly to the reparametrization from before we can transform

$$\underset{\theta \in \mathbb{R}^n}{\text{minimize}} \left\{ \| Y - \theta \|_2^2 + \lambda \| D_1 \theta \|_1 \right\}$$

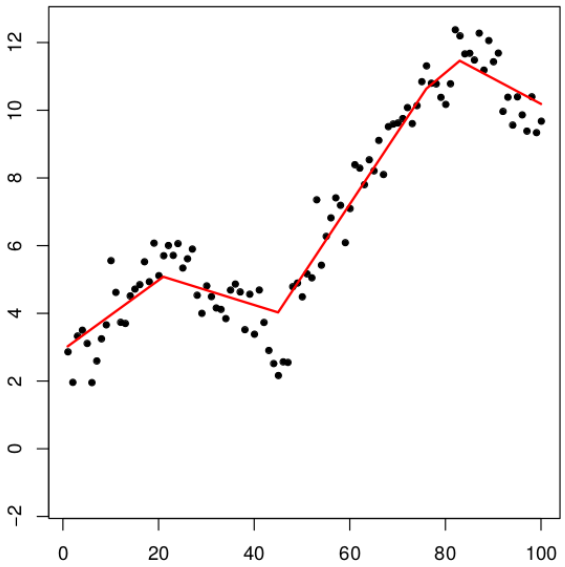
into LASSO form. Then sparsity kicks in and some of the penalty summands are set to 0, yielding segments where the solution is linear.

Trend filtering

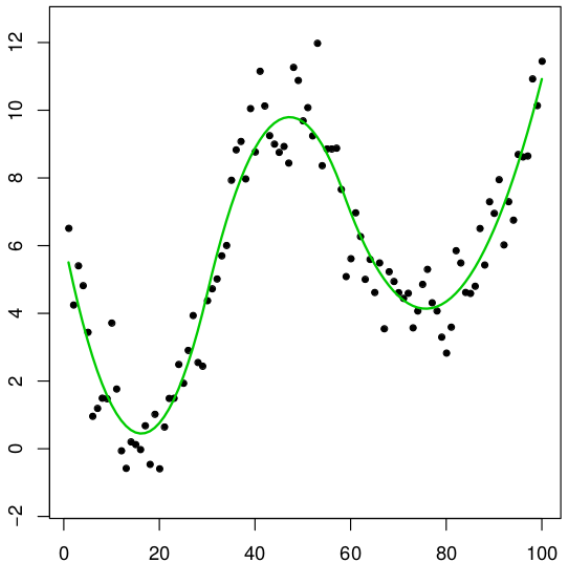
Note that this transformation is not immediate, because D_1 is not square (θ and $D_1\theta$ do not have the same dimension). For trend filtering there is a trick to save this, see [4] for the details.

[4] Tibshirani, Ryan J.; Taylor, Jonathan. The solution path of the generalized lasso. Ann. Statist. 39 (2011), no. 3, 1335–1371, section 3

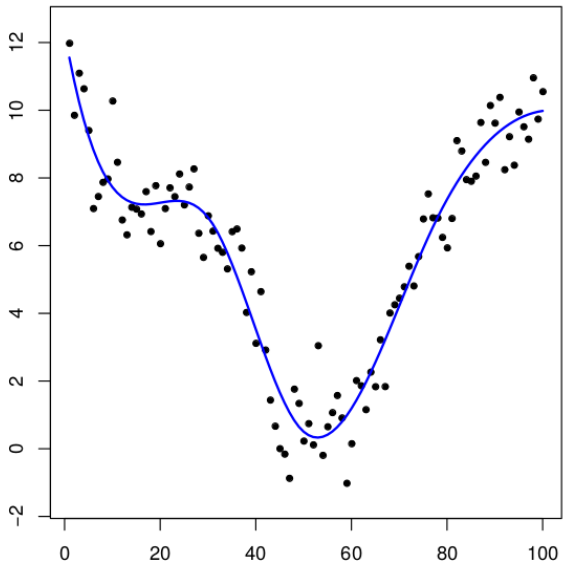
Trend filtering of orders 1, 2 and 3



(a) Linear



(b) Quadratic



(c) Cubic

Tibshirani, Ryan J.; Taylor, Jonathan. The solution path of the generalized lasso. Ann. Statist. 39 (2011), no. 3, 1335–1371

Examples for change point detection

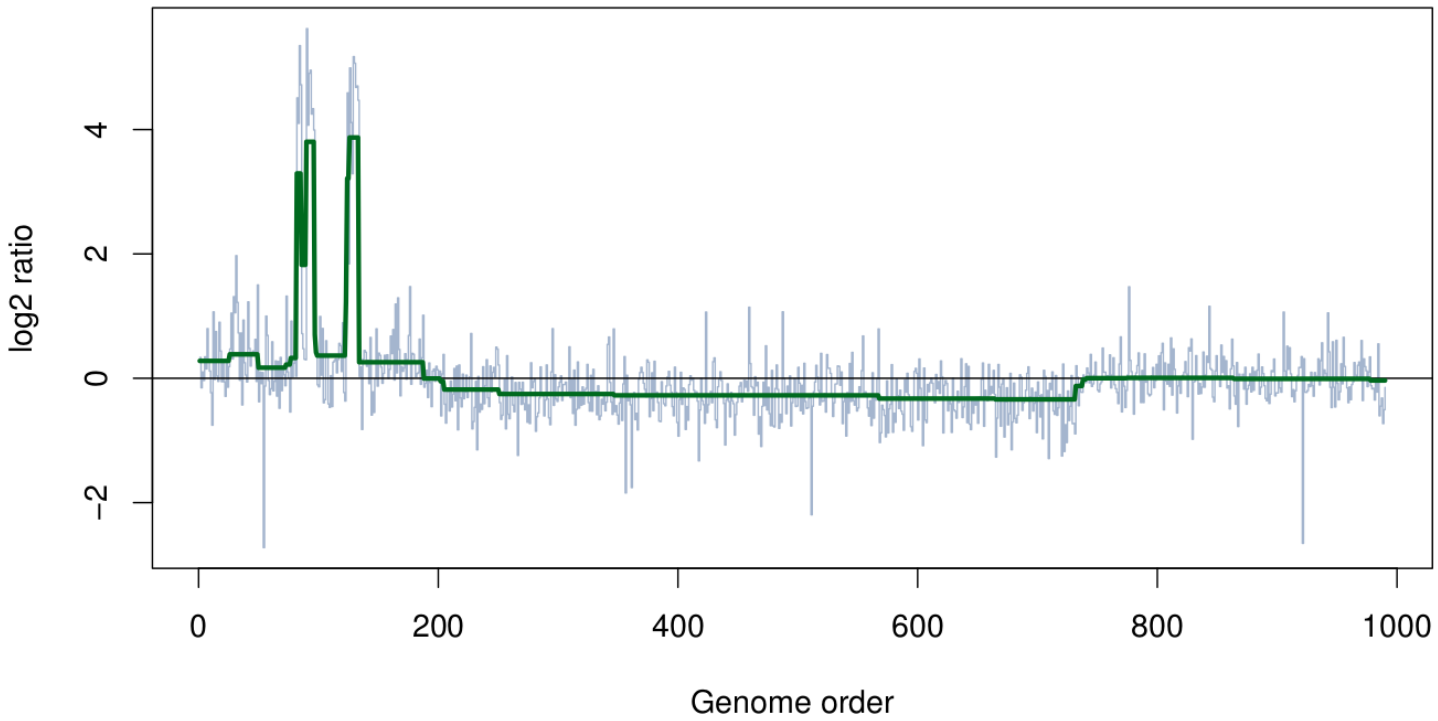


Figure 4.8 *Fused lasso applied to CGH data. Each spike represents the copy number of a gene in a tumor sample, relative to that of a control (on the log base-2 scale). The piecewise-constant green curve is the fused lasso estimate.*

Hastie, Tibshirani and Wainwright. Statistical learning with sparsity: The Lasso and generalizations, CRC Press, 2015, page 76

Examples for change point detection

Offline

- **Speech detection:** Given some audio find points where silence, words or sentences begin or end
- **Environment:** Water pollution, for example to detect how long an oil spill had a noticable effect

Online

- **Surveillance in security:** Detect changes in video images as quickly as possible for alerts
- **Finance:** If there are changes at the stock market, acting as quickly as possible is important

The model

- Times $t = 1, \ldots, n$ with observations $Y_1, \ldots, Y_n$.
- We assume that the true number K^* of change points is known.
- $Y_t = \theta_t^* + \epsilon_t$ and the θ_t^* are constant on the intervals $t_{k-1}^* \leq t < t_k^*$ for $k = 1, \ldots, K^* + 1$, where the t_k^* are the change points and the ϵ_t are noise.

Estimating the locations of the change points

$$\underset{\theta\in\mathbb{R}^n}{\text{minimize}} \sum_{i=1}^{n}(Y_i-\theta_i)^2 \text{ under the condition } \sum_{i=2}^{n} 1_{\{\theta_{i-1}\neq\theta_i\}}(\theta) = K^*$$

The Dynamic Programming approach for this problem has time complexity $O(n^2)$, which is too slow in general.

By changing the ℓ_0 condition to an ℓ_1 condition, the problem opens up to different approaches.

$$\underset{\theta\in\mathbb{R}^n}{\text{minimize}} \sum_{i=1}^{n}(Y_i-\theta_i)^2 \text{ under the condition } \sum_{i=2}^{n} |\theta_{i-1}-\theta_i| \leq K^* J^*_{\text{max}},$$

where $J^*_{\text{max}} = \underset{1<i\leq n}{max}|\theta^*_i - \theta^*_{i-1}|$ denotes the true maximum jump size.

Estimating the locations of the change points

With the substitution $\beta = M\theta$ from earlier this becomes

$$\underset{\beta \in \mathbb{R}^n}{\text{minimize}} \; \|Y - M^{-1}\beta\|_2^2 \text{ under the condition } \|\beta\|_1 \leq K^* J^*_{\max},$$

where again the excess $|\beta_1|$ term is not penalized. This is the LASSO, which will automatically produce a sparse vector β from which the change points can be recovered:

$$\{\hat{t}_1, \ldots, \hat{t}_K\} = \{1 \leq i \leq n \mid \hat{\beta}_i \neq 0\},$$

where K is the number of change points selected by the LASSO and the $\hat{t}_i$ are ordered ascendingly.

Asymptotic correctness of the locations of change points

One could ask how far away the estimated change points are from the true locations, that is how can we bound the expression $\max_{1 \leq k \leq K^*} |\hat{t}_k - t_k^*|$ (assuming that $K = K^*$).

We show a result by Harchaoui and Lévy-Leduc[7] in this direction.

For this we use the Lagrangian form

$$\hat{\beta}(\lambda_n) = \underset{\beta \in \mathbb{R}^n}{\text{minimize}} \left\{ \| Y - M^{-1}\beta \|_2^2 + \lambda_n \|\beta\|_1 \right\}.$$

[7] Z. Harchaoui and C. Lévy-Leduc, Multiple Change-Point Estimation With a Total Variation Penalty, Journal of the American Statistical Association, 2012.

Asymptotic correctness of the locations of change points

Some assumptions:

(1) The errors ϵ_i are subgaussian and iid with zero mean.

(2) The sequence $(\delta_n)_{n\in\mathbb{N}}$ is nonincreasing, positive and converges to 0 such that $n\delta_n(J^*_{\min})^2/\log(n) \to \infty$, where $J^*_{\min}$ is the minimum jump size.

(3) The true change points satisfy $I^*_{\min} = \min_{1\le k\le K^*} |t^*_{k+1} - t^*_k| \ge n\delta_n$, where $I^*_{\min}$ is the length of the shortest segment. (So $I^*_{\min}/n \ge \delta_n$).

(4) The parameters $(\lambda_n)_{n\in\mathbb{N}}$ satisfy $\lambda_n/(n\delta_n J^*_{\min}) \to 0$.

Asymptotic correctness of the locations of change points

Proposition

Let K^* be the true number of change points and K the number of estimated change points. If $K = K^*$ and the assumptions from the previous slide hold, then

$$\mathbb{P}\left(\max_{1\leq k\leq K^*} \frac{|\hat{t}_k - t_k^*|}{n} \leq \delta_n\right) \to 1 \text{ as } n \to \infty.$$

Consider the case where we want to find a constant number of segments. Let $I^*_{\text{min}} = n/10$, which means we have at most 10 segments. Assume that J^*_{min} is (at least) constant and take $\delta_n = n^{-\alpha}/10$ with $\alpha < 1$. Let $\lambda_n = \log(n)$. Then

$$\mathbb{P}\left(\max_{1\leq k\leq K^*} \left|\frac{\hat{t}_k - t_k^*}{n}\right| \leq \frac{1}{10n^{\alpha}}\right) \to 1 \text{ as } n \to \infty.$$

Another case

On the other hand one can look at the case where the number of segments increases with n, for example $I^*_{\text{min}} = \sqrt{n}$. If we use $\delta_n = 1/\sqrt{n}$ and $\lambda_n = \log(n)$, then the minimum jump size can be allowed to decrease, for example $J^*_{\text{min}} \geq n^{-\alpha}$ with $\alpha < \frac{1}{4}$ works. Then

$$\mathbb{P}\left(\max_{1 \leq k \leq K^*} \frac{|\hat{t}_k - t^*_k|}{n} \leq \frac{1}{\sqrt{n}} \right) \to 1 \text{ as } n \to \infty.$$