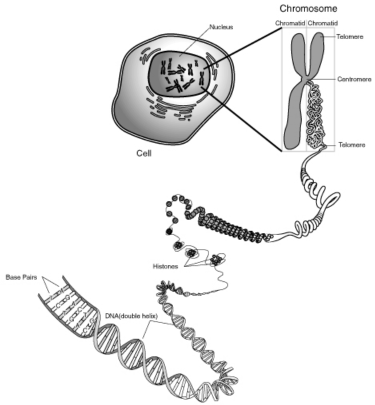
On segmentation of DNA copy number profiles

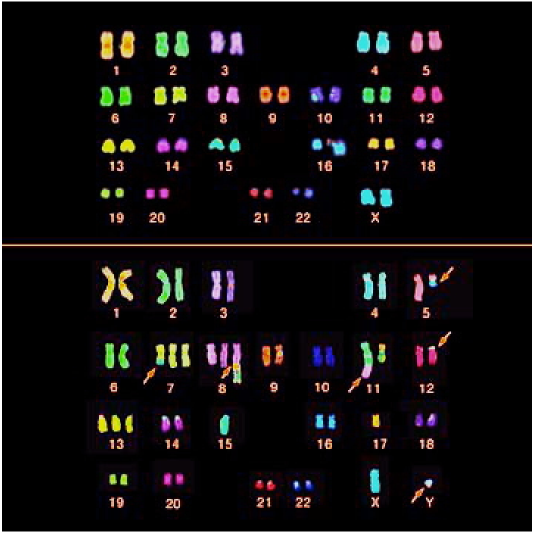
Jean-Philippe Vert

(joint work with Toby Hocking, Gudrun Schleiermacher, Isabelle Janoueix-Lerosey and Francis Bach)

13th International Workshop on Bioinformatics and Systems Biology (IBSB 2013), Kyoto University, August 1st, 2013

Chromosomic aberrations in cancer

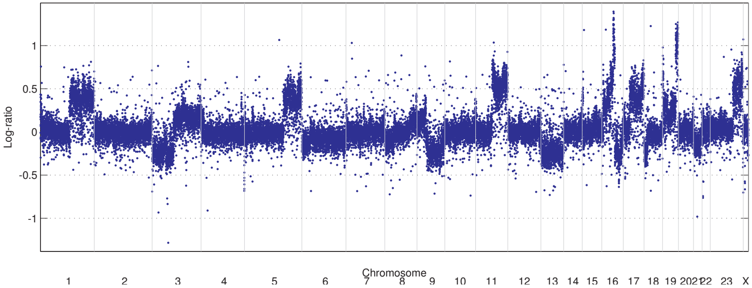




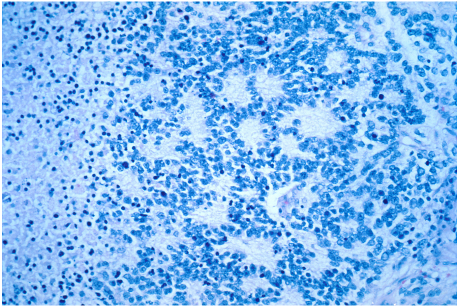
Array Comparative Genomic Hybridization (aCGH)

Motivation

- Comparative genomic hybridization (CGH) data measure the DNA copy number along the genome
- Very useful, in particular in cancer research to observe systematically variants in DNA content

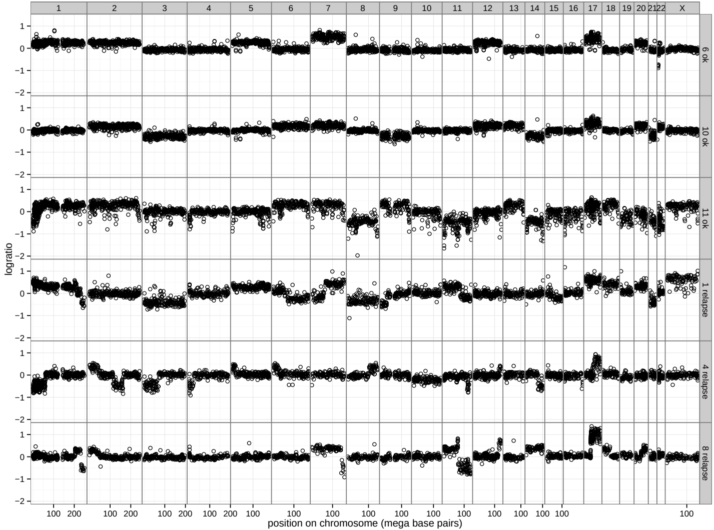


Neuroblastoma



- Rare, but most common cancer in infants
- Arises from nervous cells, frequent metastasis
- One of the few human malignancies known to demonstrate spontaneous regression

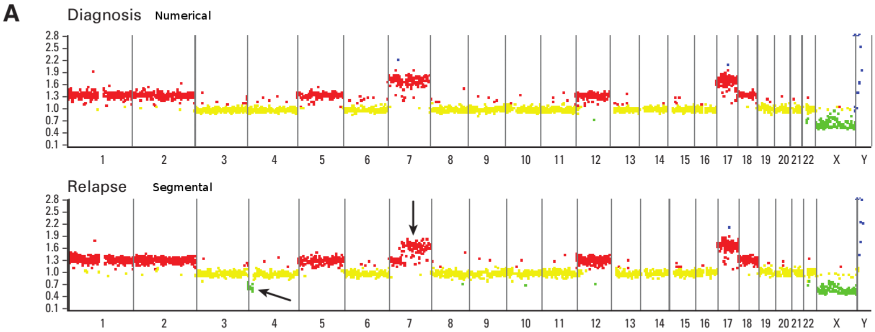
aCGH neuroblastoma copy number data



Copy number profiles are predictive of progression in neuroblastoma

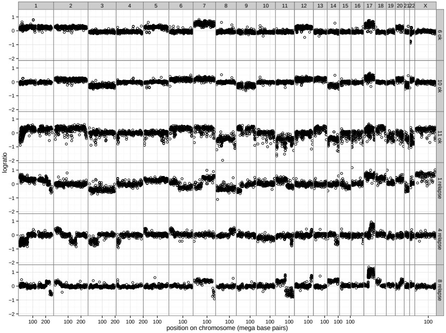
2 types of profiles:

- Numerical: entire chromosome amplification. Good outcome.
- Segmental: deletion 1p 3p 11q, gain 1q 2p 17q. Bad outcome. In this talk “breakpoints.”



(Schleiermacher et al., J Clinical Oncology, 2010)

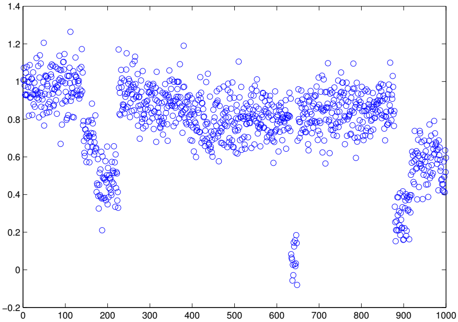
Questions



- Refine classification of neuroblastoma in terms of breakpoints
- Refine prognosis based on breakpoints
- Predict metastatic locations from breakpoints

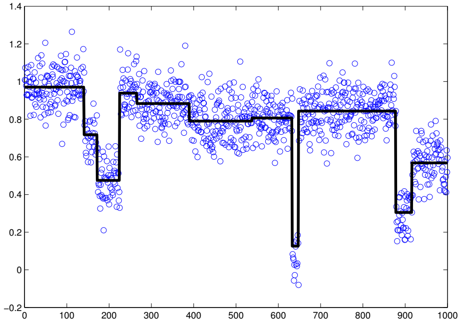
We need to automatically identify breakpoints

In this talk: How to automatically identify breakpoints?



1. Learning smoothing models using expert annotation
2. Optimizing multi-parameter models
3. Fast and scalable segmentation

In this talk: How to automatically identify breakpoints?



1. Learning smoothing models using expert annotation
2. Optimizing multi-parameter models
3. Fast and scalable segmentation

Outline

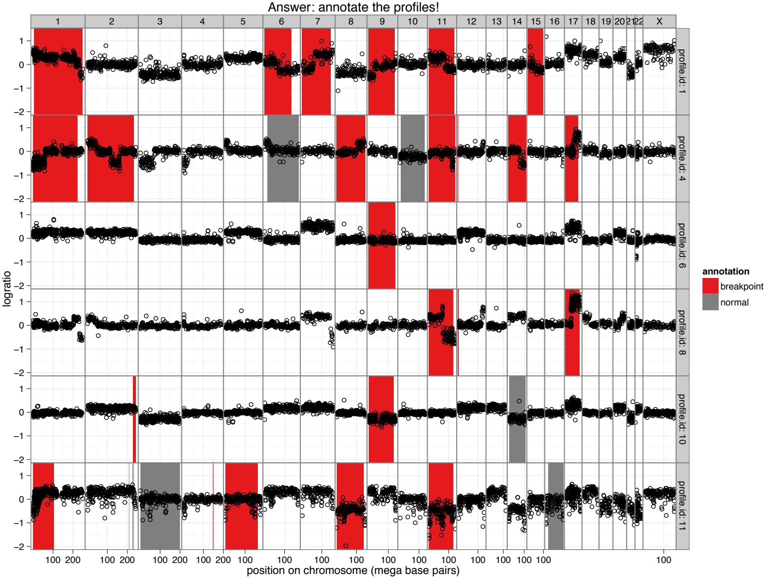
Many models proposed to detect breakpoints..

- GLAD: adaptive weights smoothing (Hupé *et al.*, 2004)
- DNAcopy: circular binary segmentation (Venkatraman and Olshen, 2007)
- cghFLasso: fused lasso signal approximator with heuristics (Tibshirani and Wang, 2007)
- HaarSeg: wavelet smoothing (Ben-Yaacov and Eldar, 2008)
- flsa: fused lasso signal approximator path algorithm (Hoefling 2009)
- cghseg (Rigaill 2010) and PELT (Kilick et al. 2012): pruned dynamic programming
- gada: Sparse representation and Bayesian detection (Pique-Regi et al, 2008)

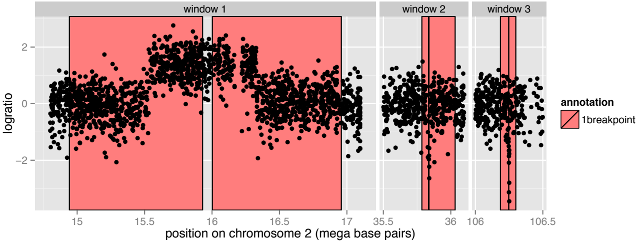
Each model has a parameter to tune the degree of smoothness, and often a default parameter.

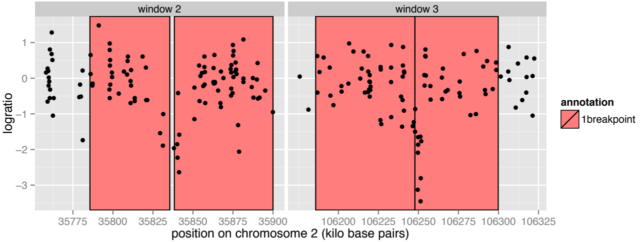
1. How to define which model is best?
2. And how to choose the degree of smoothness?

Our answer: Ask an expert!



SegAnnDB for easy and fast partial expert annotation





https://gforge.inria.fr/scm/viewvc.php/webapp/?root=breakpoints

2 experts annotated 575 neuroblastoma profiles

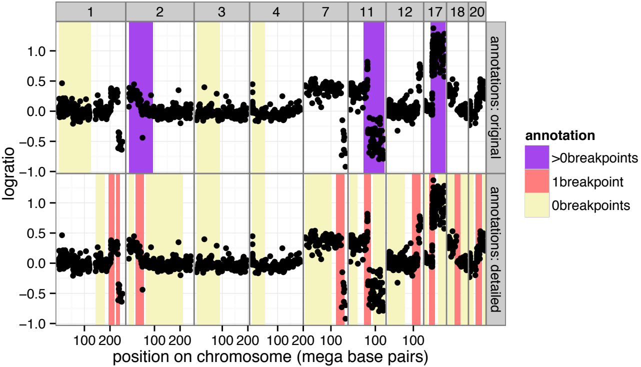
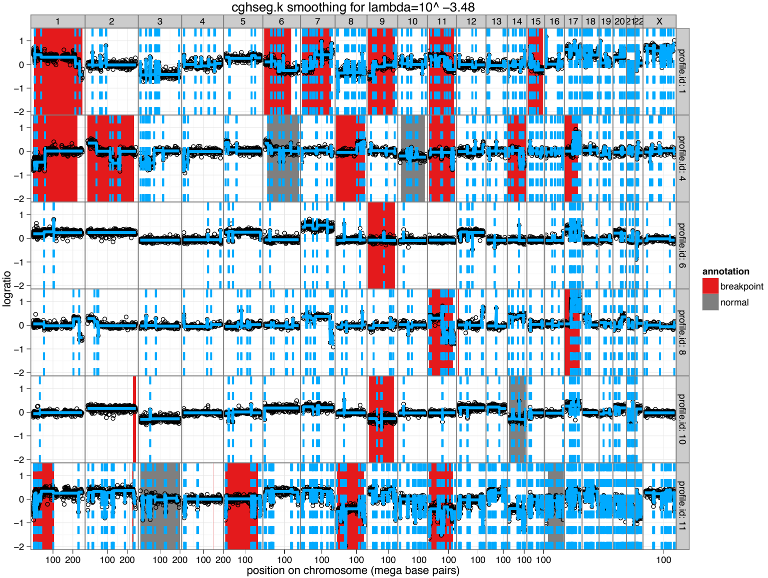


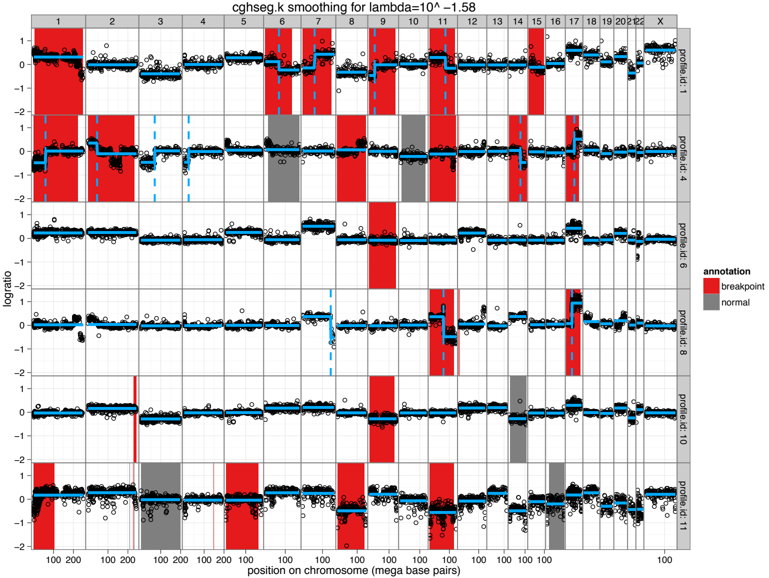
Table 1 Counts of annotations in two annotation data sets of the same copy number profiles

	Original	Detailed
protocol	Systematic	Any
annotated profiles	575	575
annotated chromosomes	3418	3730
annotations	3418	4359
0breakpoints	2845	3395
1breakpoint	0	521
>0breakpoints	573	443

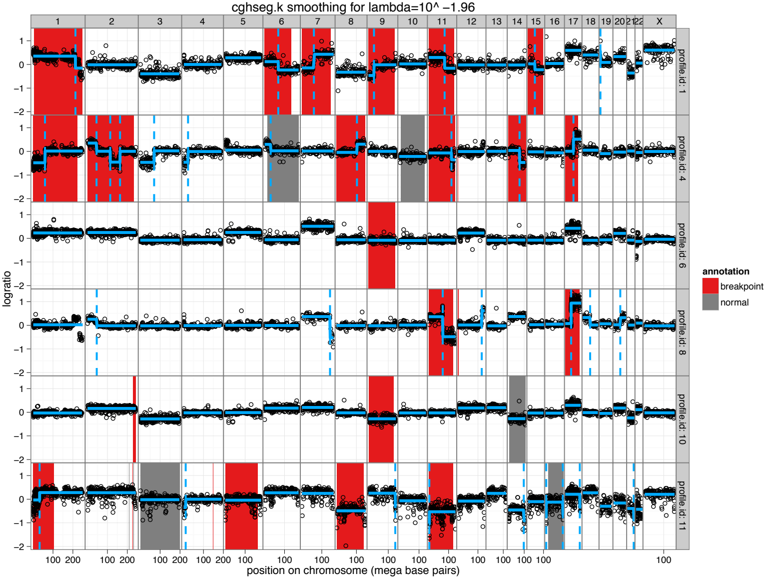
Testing a model: Mostly over-segmented



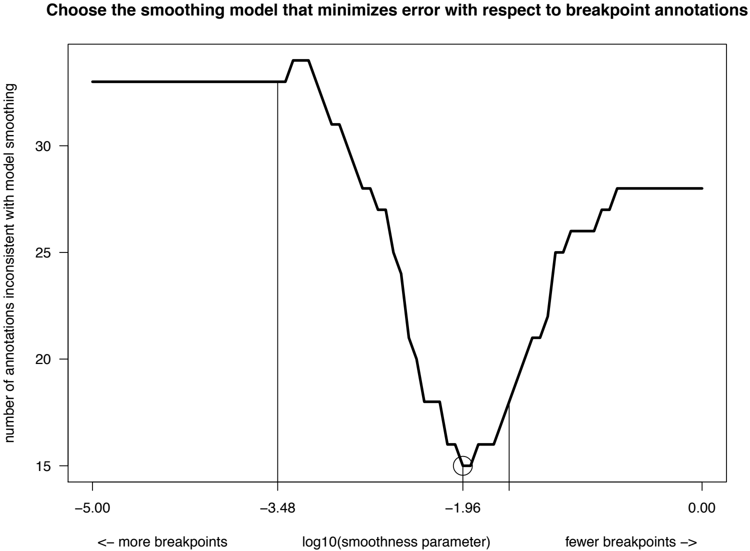
Testing a model: Mostly under-segmented



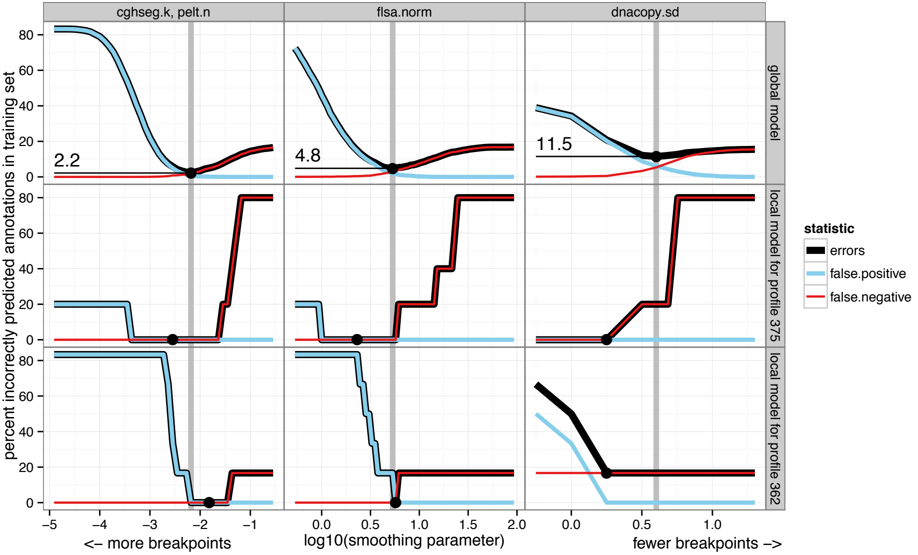
Testing a model: Not too bad



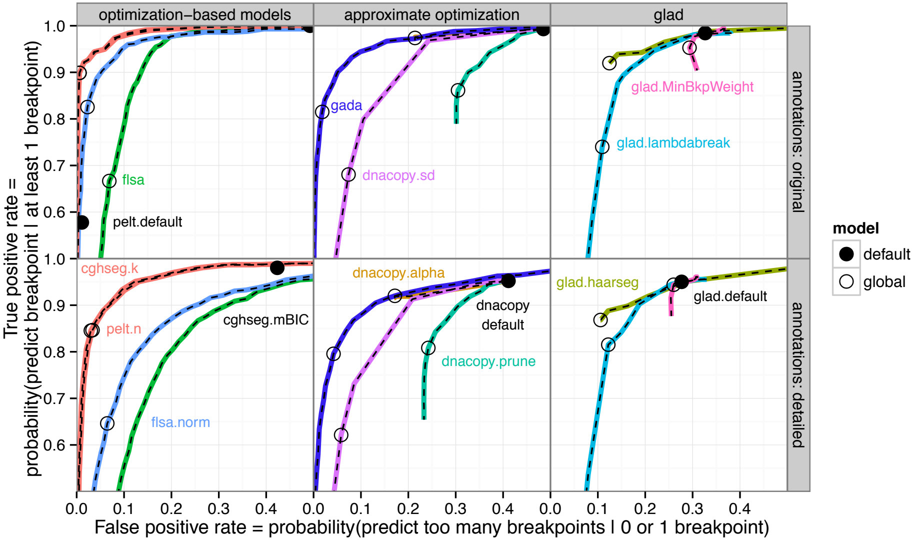
Error curve


Choose the smoothing model that minimizes error with respect to breakpoint annotations
number of annotations inconsistent with model smoothing
30
25
20
15
−5.00
−3.48
−1.96
0.00
<– more breakpoints
log10(smoothness parameter)
fewer breakpoints –>

Global error = same parameter for all profiles
Local error = parameter optimized for each profile



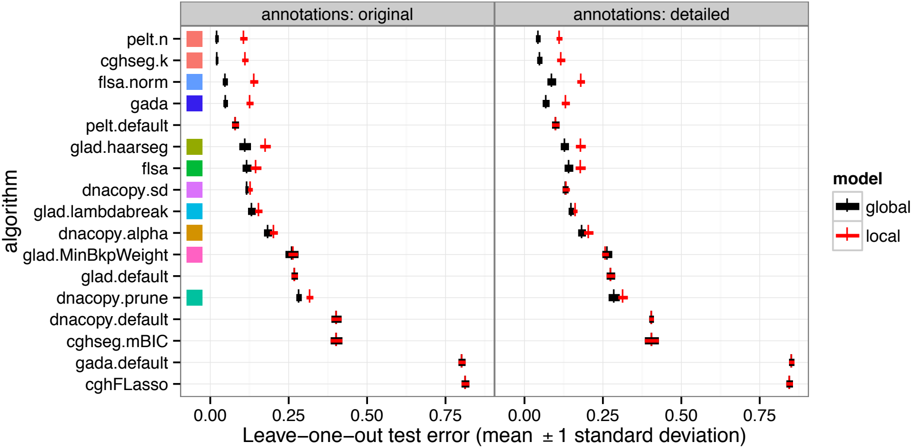
Global error of 17 segmentation methods



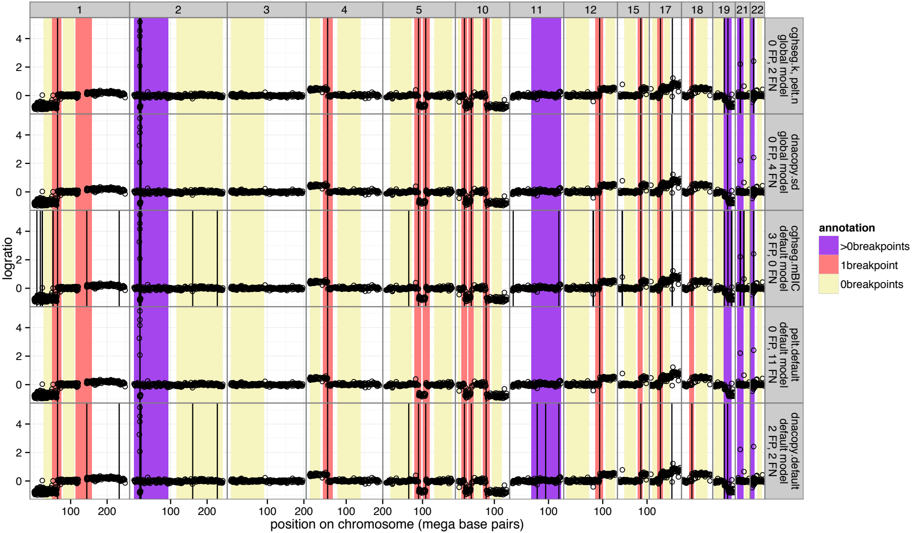
Best local errors

algorithm	original error	original FP	original FN	detailed error	detailed FP	detailed FN
pelt.n	0.3	0.7	0.2	2.1	5.3	1.0
cghseg.k	0.3	0.7	0.2	2.3	5.6	1.1
gada	0.6	1.6	0.5	2.5	6.3	1.2
dnacopy.sd	2.5	7.5	1.5	5.1	14.1	2.2
glad.lambdabreak	6.4	2.1	7.3	8.0	7.1	7.2
flsa.norm	1.2	3.0	0.8	8.7	19.5	4.9
flsa	1.3	1.4	1.3	8.9	20.6	4.9
glad.haarseg	9.0	1.6	10.5	9.5	6.0	9.0
pelt.default	8.0	42.2	1.1	13.9	59.0	1.0
dnacopy.alpha	17.9	1.4	21.2	16.8	7.2	16.9
glad.MinBkpWeight	19.7	0.7	23.6	18.4	4.6	19.4
dnacopy.prune	25.9	2.8	30.5	23.6	8.9	24.1
glad.default	27.4	1.6	32.7	26.0	5.0	27.7
dnacopy.default	40.5	0.7	48.5	38.0	4.8	41.1
cghseg.mBIC	41.0	0.0	49.2	38.5	2.0	42.3
gada.default	80.7	0.0	96.9	82.7	0.1	92.1
cghFLasso	80.9	0.0	97.2	83.8	0.8	93.1

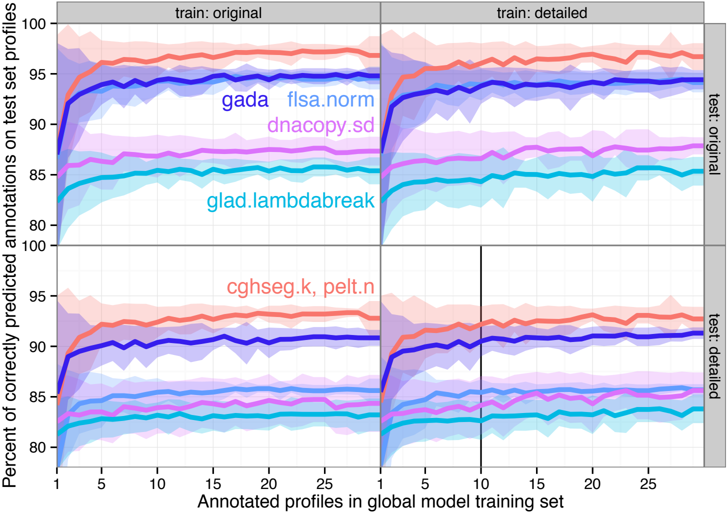
Leave-one-out error: Global models are more robust



Globally-optimized parameters are better than default parameters on new profiles



Only 10 annotated samples are sufficient to learn parameters, which are robust across annotators



Generalization error for models trained on 10 profiles

algorithm		error	sd	fn	sd	fp	sd	Timings
pelt.n	■	7.7	1.8	17.9	9.8	4.1	3.4	9.49
cghseg.k	■	7.8	1.8	17.8	9.7	4.3	3.2	2.79
gada	■	9.5	1.5	28.2	12.6	3.6	2.8	7.54
glad.haarseg	■	13.2	1.4	12.2	1.2	11.7	1.8	32.62
pelt.default		13.9	0.1	59.0	0.3	1.0	0.0	0.08
flsa.norm	■	14.6	1.3	39.3	13.1	6.5	3.6	0.12
dnacopy.sd	■	15.8	2.9	42.8	24.2	7.1	5.6	61.90
glad.lambdabreak	■	17.4	1.9	25.4	15.9	13.1	4.4	17.02
dnacopy.alpha	■	17.8	0.8	8.1	0.2	17.8	0.9	29.38
flsa	■	20.1	1.2	56.2	25.6	8.5	5.8	0.06
glad.MinBkpWeight	■	25.5	1.0	7.8	3.0	26.5	1.4	42.39
glad.default		26.0	0.1	5.0	0.2	27.7	0.1	1.34
dnacopy.prune	■	26.7	1.0	19.5	4.8	24.9	2.0	41.34
dnacopy.default		38.0	0.2	4.8	0.1	41.1	0.2	2.02
cghseg.mBIC		38.5	0.1	2.0	0.1	42.3	0.1	1.81
gada.default		82.7	0.1	0.1	0.0	92.1	0.1	0.20
cghFLasso		83.8	0.1	0.8	0.1	93.1	0.1	0.18

Summary: the winner is...

- Best model are cghseg.k and pelt.n: implement a Gaussian maximum-likelihood piecewise constant smoothing model:

$$\min_{k,\mu^k} \frac{1}{m} \sum_{i=1}^{m} (x_i - \mu_i)^2 + \lambda k$$

  where μ^k has at most k change-points
- λ is optimized on 10 expert-annotated profiles.
- Better than default parameters
- Robust across annotators
- More details: T. Hocking et al. (2013) Learning smoothing models of copy number profiles using breakpoint annotations. *BMC Bioinformatics* 14:164.

Outline

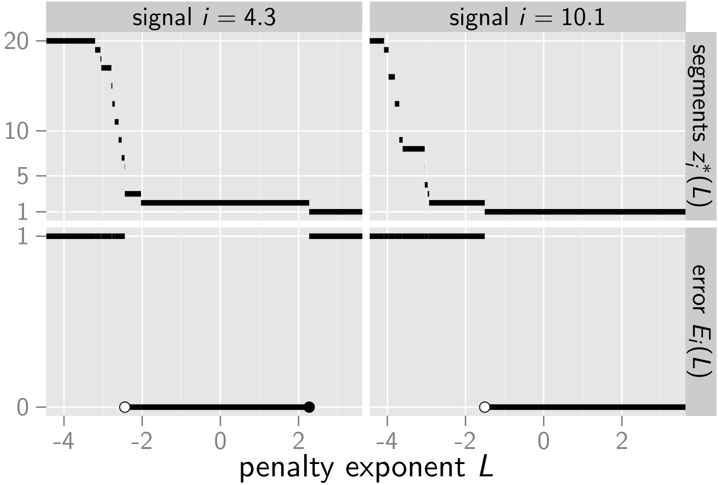
The cghseg.k / pelt.n least squares model

- The model isl:

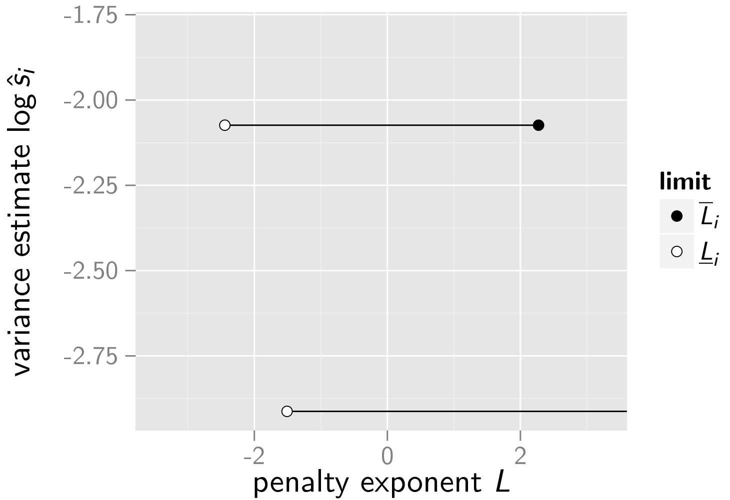
$$\min_{k,\mu^k} \frac{1}{m} \sum_{i=1}^{m} (x_i - \mu_i)^2 + \lambda k$$

  where μ^k has at most k change-points
- $\lambda = 1$ by default, but better to use λ optimized to maximize agreement with a database of breakpoint annotations
- Why this particular penalty? What about taking into account other properties of the signal, such as its length or variance?

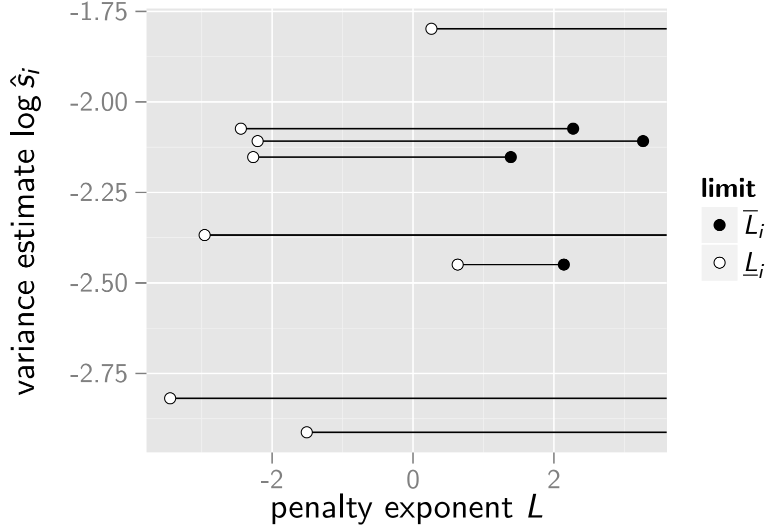
Error curve for 2 annotated signals, $L = \log(\lambda)$



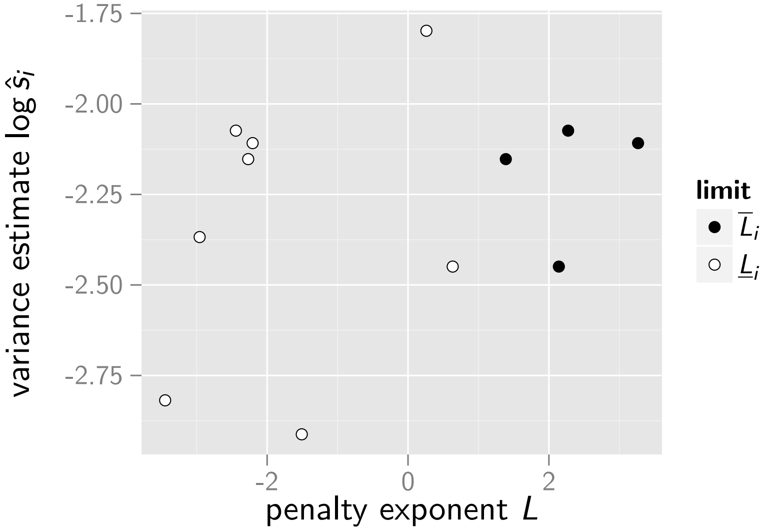
Target intervals $[\underline{L}_i, \overline{L}_i]$, as a function of estimated variance for 2 signals



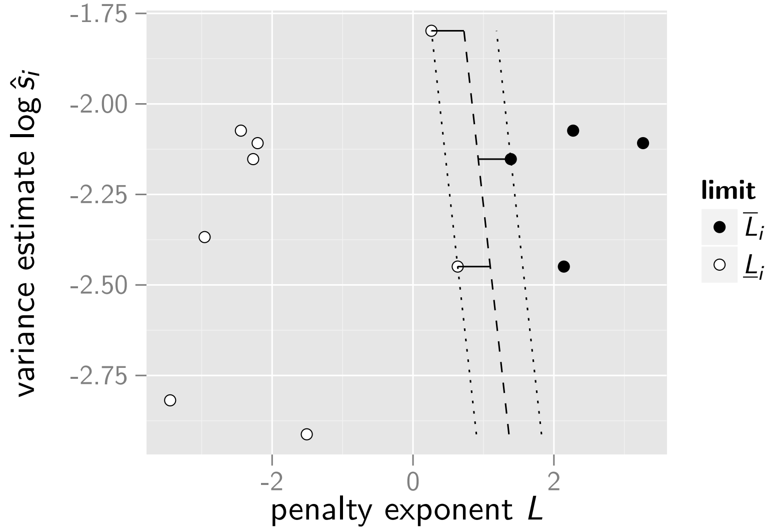
Target interval $[\underline{L}_i, \overline{L}_i]$ for all signals



Limit point representation



Max margin regression line



Learning the penalty function

$$\min_{k,\mu^k} \frac{1}{m} \sum_{i=1}^{m} (x_i - \mu_i)^2 + \lambda(\hat{\sigma}) k$$

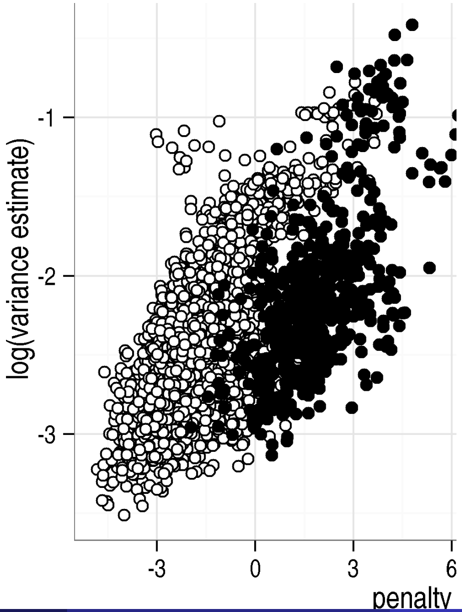
- For every signal i, estimate the variance $\hat{\sigma}_i$
- Parametrize the penalty as

$$\log \lambda_i = \beta + w \log \hat{\sigma}_i$$

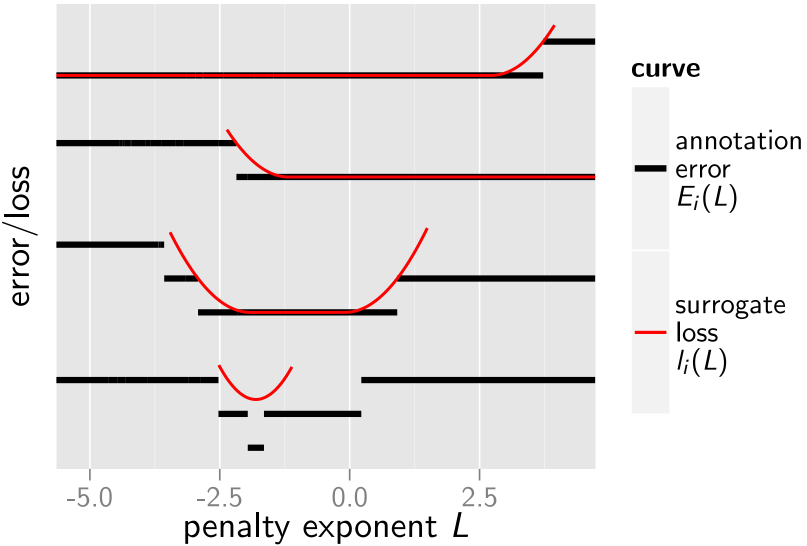
- Equivalent to learning a penalty function

$$\min_{k,\mu^k} \frac{1}{m} \sum_{i=1}^{m} (x_i - \mu_i)^2 + e^{\beta} \hat{\sigma}^{w} k$$

Real data can usually not be separated...

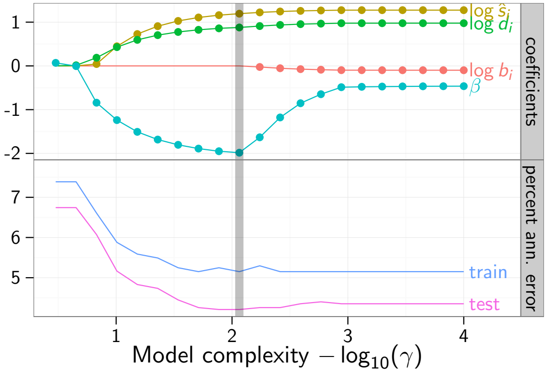


Convex surrogate loss for the annotation error

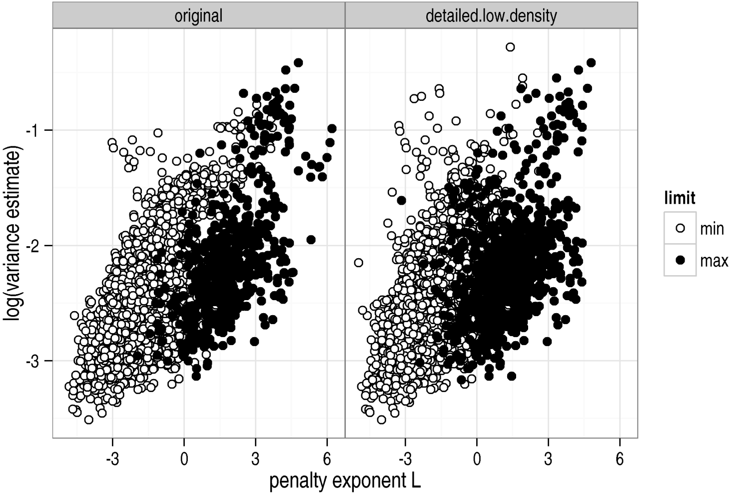


Learning the parameters

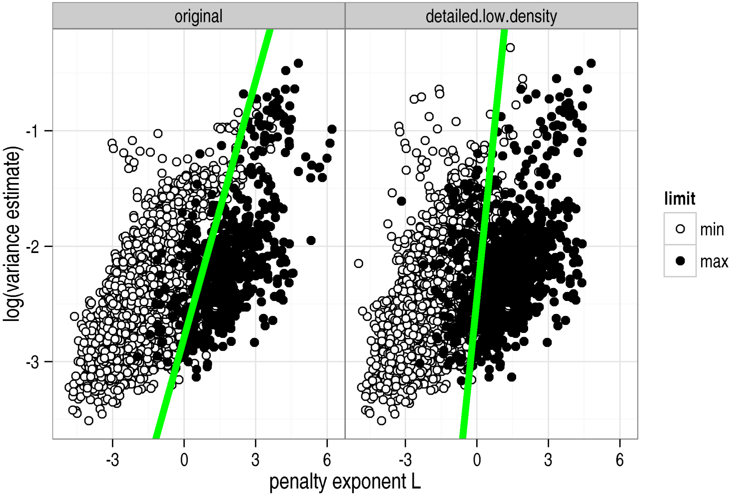
$$\underset{\beta,w}{\operatorname{argmin}} \frac{1}{N}\sum_{i=1}^{N}\ell_i(\beta + w^\top x_i) + \gamma\|w\|_1$$



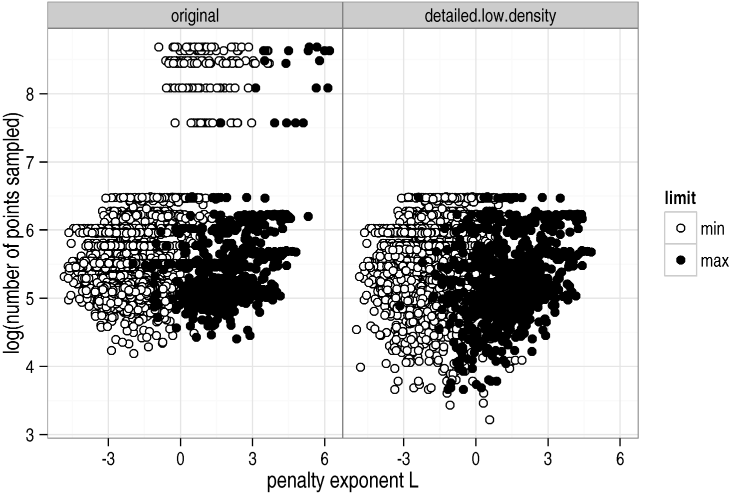
Optimal model complexity depends on variance



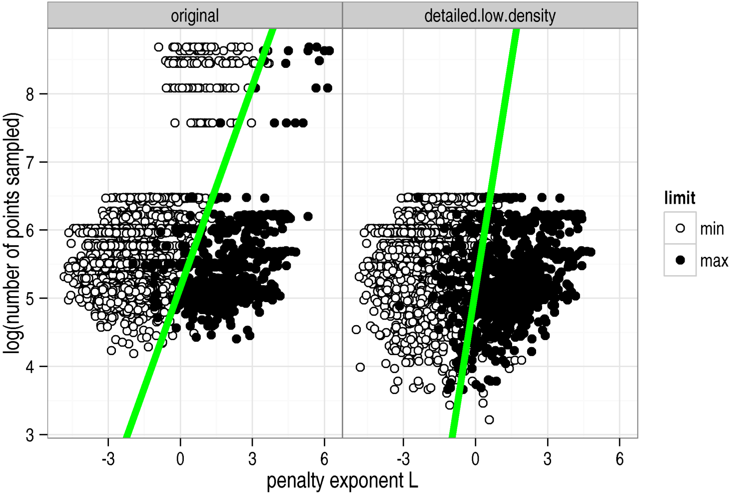
Optimal model complexity depends on variance



Optimal model complexity depends on number of points



Optimal model complexity depends on number of points



Learned coefficients

Model with two features:

- Variance estimate $\log \hat{s}_i$.
- Number of points sampled $\log d_i$.
- Learned model complexity

$$f(x_i) = w_1 \log \hat{s}_i + w_2 \log d_i + \beta.$$

- Learned penalty function

$$z_i^*[f(x_i)] = \arg\min_k ||y_i - \hat{y}_i^k||_2^2 + \hat{s}_i^{w_1} d_i^{w_2} e^{\beta} k.$$

annotation data set	variance w_1	points w_2	intercept β
original	1.01	0.96	−2.66
	±0.03	±0.02	±0.10
detailed.low.density	1.30	0.93	−2.00
	±0.02	±0.02	±0.13

Mean ± 1 standard deviation over 10 folds.

Error estimated using 10-fold cross-validation

$$f(x_i) = w_1 \log \hat{s}_i + w_2 \log d_i + \beta$$

- cghseg.k: $w_1 = 0$, $w_2 = 1$, learn β using grid search to minimize the annotation error E_i.
- log.s.log.d: learn β, w_1, w_2 by minimizing the un-regularized ($\gamma = 0$) surrogate loss l_i.
- L1-reg: variance estimate, signal size, model error, chromosome indicator features $x_i \in \mathbb{R}^{117}$, CV to choose the degree of ℓ_1 regularization γ.

model	variables	original	detailed.low.density
BIC	0	7.99 ± 0.00	13.64 ± 0.00
mBIC	0	40.99 ± 0.00	36.88 ± 0.00
cghseg.k	0	2.19 ± 0.82	6.49 ± 1.16
log.s.log.d	2	1.90 ± 0.77	4.72 ± 0.54
L1-reg	117	1.81 ± 0.58	4.70 ± 0.88

Summary

- Complex penalties with multiple parameters can be optimized
- Equivalent to interval regression problem, which we solve with convex optimization
- Optimized penalties are better than default or simple penalties
- More details: Rigaill et al. (2013) Learning sparse penalties for change-point detection using max margin interval regression. In Proceedings of ICML 2013.

Outline

How to scale to $p = 10^7 \sim 10^9$?

algorithm	error	sd	fn	sd	fp	sd	Timings
pelt.n	7.7	1.8	17.9	9.8	4.1	3.4	9.49
cghseg.k	7.8	1.8	17.8	9.7	4.3	3.2	2.79
gada	9.5	1.5	28.2	12.6	3.6	2.8	7.54
glad.haarseg	13.2	1.4	12.2	1.2	11.7	1.8	32.62
pelt.default	13.9	0.1	59.0	0.3	1.0	0.0	0.08
flsa.norm	14.6	1.3	39.3	13.1	6.5	3.6	0.12
dnacopy.sd	15.8	2.9	42.8	24.2	7.1	5.6	61.90
glad.lambdabreak	17.4	1.9	25.4	15.9	13.1	4.4	17.02
dnacopy.alpha	17.8	0.8	8.1	0.2	17.8	0.9	29.38
flsa	20.1	1.2	56.2	25.6	8.5	5.8	0.06
glad.MinBkpWeight	25.5	1.0	7.8	3.0	26.5	1.4	42.39
glad.default	26.0	0.1	5.0	0.2	27.7	0.1	1.34
dnacopy.prune	26.7	1.0	19.5	4.8	24.9	2.0	41.34
dnacopy.default	38.0	0.2	4.8	0.1	41.1	0.2	2.02
cghseg.mBIC	38.5	0.1	2.0	0.1	42.3	0.1	1.81
gada.default	82.7	0.1	0.1	0.0	92.1	0.1	0.20
cghFLasso	83.8	0.1	0.8	0.1	93.1	0.1	0.18

Promoting sparsity with the ℓ_1 penalty

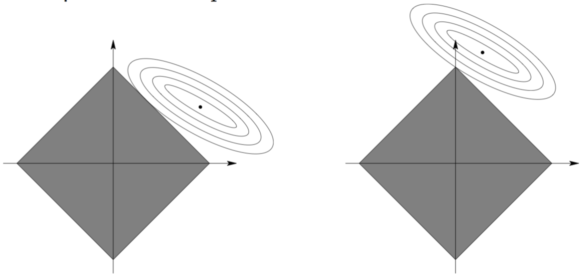
The ℓ_1 penalty (Tibshirani, 1996; Chen et al., 1998)

If $R(\beta)$ is convex and "smooth", the solution of

$$\min_{\beta \in \mathbb{R}^p} R(\beta) + \lambda \sum_{i=1}^{p} |\beta_i|$$

is usually sparse.

Geometric interpretation with $p = 2$



Promoting piecewise constant profiles penalty

The total variation / variable fusion penalty

If $R(\beta)$ is convex and "smooth", the solution of

$$\min_{\beta \in \mathbb{R}^p} R(\beta) + \lambda \sum_{i=1}^{p-1} |\beta_{i+1} - \beta_i|$$

is usually piecewise constant (Rudin et al., 1992; Land and Friedman, 1996).

Proof:

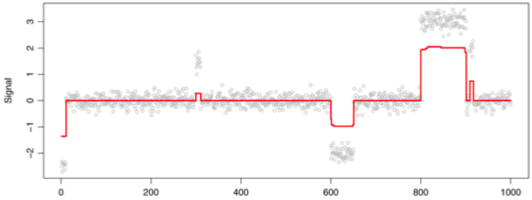
- Change of variable $u_i = \beta_{i+1} - \beta_i$, $u_0 = \beta_1$
- We obtain a Lasso problem in $u \in \mathbb{R}^{p-1}$
- u sparse means β piecewise constant

TV signal approximator (=FLSA)

$$\min_{\beta \in \mathbb{R}^p} \parallel Y - \beta \parallel^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} | \beta_{i+1} - \beta_i | \leq \mu$$

Adding additional constraints does not change the change-points:

- $\sum_{i=1}^{p} | \beta_i | \leq \nu$ (Tibshirani et al., 2005; Tibshirani and Wang, 2008)
- $\sum_{i=1}^{p} \beta_i^2 \leq \nu$ (Mairal et al. 2010)



Solving TV signal approximator

$$\min_{\beta \in \mathbb{R}^p} \| Y - \beta \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} | \beta_{i+1} - \beta_i | \leq \mu$$

- QP with sparse linear constraints in $O(p^2)$ -> 135 min for $p = 10^5$ (Tibshirani and Wang, 2008)
- Coordinate descent-like method $O(p)$? -> 3s s for $p = 10^5$ (Friedman et al., 2007)
- For all μ with the LARS in $O(pK)$ (Harchaoui and Levy-Leduc, 2008)
- For all μ in $O(p \ln p)$ (Hoefling, 2009)
- For the first K change-points in $O(p \ln K)$ (Bleakley and V., 2010)

TV signal approximator as dichotomic segmentation

Algorithm 1 Greedy dichotomic segmentation

Require: k number of intervals, $\gamma(I)$ gain function to split an interval I into $I_L(I), I_R(I)$

I_0 represents the interval $[1, n]$
$\mathcal{P} = \{I_0\}$
for $i = 1$ to k **do**
  $I^* \leftarrow \arg\max_{I \in \mathcal{P}} \gamma(I^*)$
  $\mathcal{P} \leftarrow \mathcal{P} \setminus \{I^*\}$
  $\mathcal{P} \leftarrow \mathcal{P} \cup \{I_L(I^*), I_R(I^*)\}$
end for
return $\mathcal{P}$

Theorem (V. and Bleakley, 2010; see also Hoefling, 2009)

TV signal approximator performs "greedy" dichotomic segmentation

Apparently greedy algorithm finds the global optimum!

TV signal approximator as dichotomic segmentation

Algorithm 1 Greedy dichotomic segmentation

Require: k number of intervals, $\gamma(I)$ gain function to split an interval I into $I_L(I), I_R(I)$

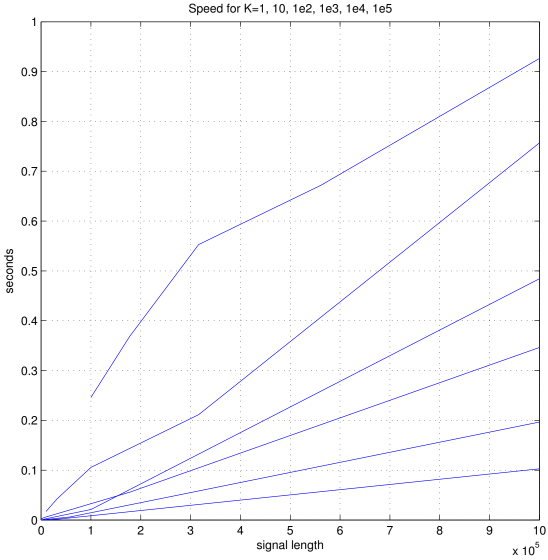
1: I_0 represents the interval $[1, n]$
2: $\mathcal{P} = \{I_0\}$
3: **for** $i = 1$ to k **do**
4: $\quad I^* \leftarrow \arg\max_{I \in \mathcal{P}} \gamma(I^*)$
5: $\quad \mathcal{P} \leftarrow \mathcal{P} \backslash \{I^*\}$
6: $\quad \mathcal{P} \leftarrow \mathcal{P} \cup \{I_L(I^*), I_R(I^*)\}$
7: **end for**
8: **return** $\mathcal{P}$

Theorem (V. and Bleakley, 2010; see also Hoefling, 2009)

TV signal approximator performs "greedy" dichotomic segmentation

Apparently greedy algorithm finds the global optimum!

Speed trial : 2 s. for $K = 100$, $p = 10^7$

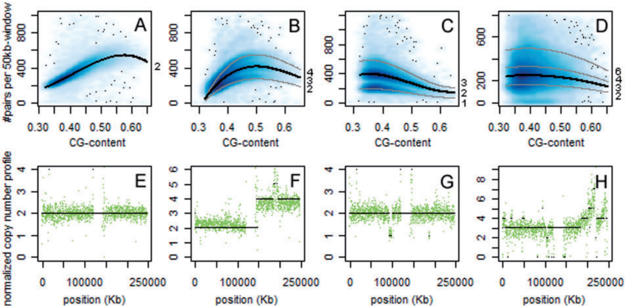


BIOINFORMATICS APPLICATIONS NOTE
Vol. 27 no. 2 2011, pages 268–269
doi:10.1093/bioinformatics/btq635

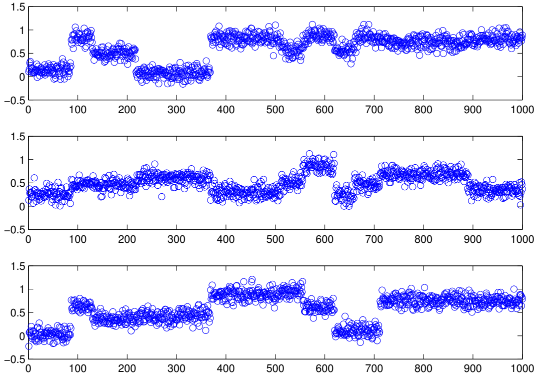
Genome analysis
Advance Access publication November 15, 2010

Control-free calling of copy number alterations in deep-sequencing data using GC-content normalization

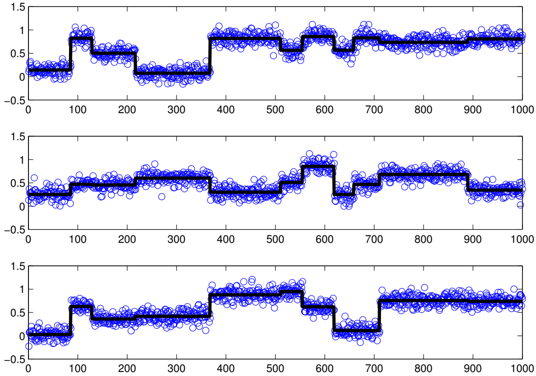
Valentina Boeva[1,2,3,4,*], Andrei Zinovyev[1,2,3], Kevin Bleakley[1,2,3], Jean-Philippe Vert[1,2,3], Isabelle Janoueix-Lerosey[1,4], Olivier Delattre[1,4] and Emmanuel Barillot[1,2,3]
[1]Institut Curie, [2]INSERM, U900, Paris, F-75248, [3]Mines ParisTech, Fontainebleau, F-77300 and [4]INSERM, U830, Paris, F-75248 France



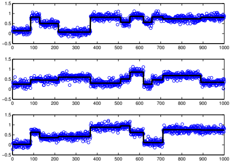
Extension: finding multiple change points shared by several profiles



Extension: finding multiple change points shared by several profiles



"Optimal" segmentation by dynamic programming



- Define the "optimal" piecewise constant approximation $\hat{U} \in \mathbb{R}^{p \times n}$ of Y as the solution of

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \mathbf{1}\left(U_{i+1,\bullet} \neq U_{i,\bullet}\right) \leq k$$

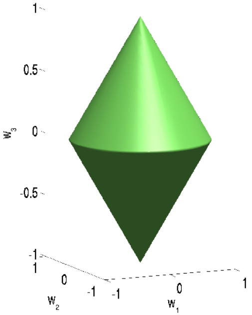
- DP finds the solution in $O(p^2 kn)$ in time and $O(p^2)$ in memory
- But: does not scale to $p = 10^6 \sim 10^9$...

Selecting pre-defined groups of variables

Group lasso (Yuan & Lin, 2006)

If groups of covariates are likely to be selected together, the ℓ_1/ℓ_2-norm induces sparse solutions *at the group level*:

$$\Omega_{group}(w) = \sum_g \|w_g\|_2$$



$$\Omega(w_1, w_2, w_3) = \|(w_1, w_2)\|_2 + \|w_3\|_2 = \sqrt{w_1^2 + w_2^2} + \sqrt{w_3^2}$$

GFLseg (Bleakley and V., 2011)

Replace

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \mathbf{1}\left(U_{i+1,\bullet} \neq U_{i,\bullet}\right) \leq k$$

by

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} w_i \| U_{i+1,\bullet} - U_{i,\bullet} \| \leq \mu$$

GFLseg = Group Fused Lasso segmentation

Questions

- Practice: can we solve it efficiently?
- Theory: does it recover the correct segmentation?

GFLseg (Bleakley and V., 2011)

Replace

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} \mathbf{1}\left(U_{i+1,\bullet} \neq U_{i,\bullet}\right) \leq k$$

by

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} w_i \| U_{i+1,\bullet} - U_{i,\bullet} \| \leq \mu$$

GFLseg = Group Fused Lasso segmentation

Questions

- Practice: can we solve it efficiently?
- Theory: does it recover the correct segmentation?

TV approximator implementation

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} w_i \| U_{i+1,\bullet} - U_{i,\bullet} \| \leq \mu$$

Theorem

The TV approximator can be solved efficiently:

- approximately with the group LARS in $O(npk)$ in time and $O(np)$ in memory
- exactly with a block coordinate descent + active set method in $O(np)$ in memory

Speed trial

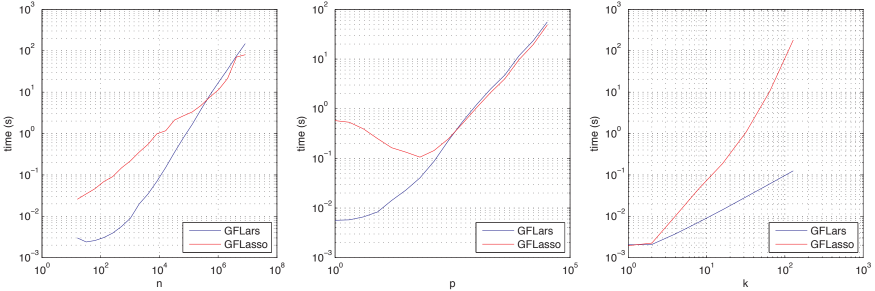
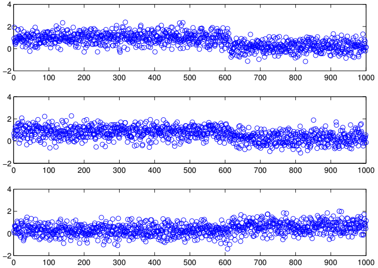


Figure 2: **Speed trials for group fused LARS (top row) and Lasso (bottom row).** *Left column:* varying n, with fixed $p = 10$ and $k = 10$; *center column:* varying p, with fixed $n = 1000$ and $k = 10$; *right column:* varying k, with fixed $n = 1000$ and $p = 10$. Figure axes are log-log. Results are averaged over 100 trials.

Consistency

Suppose a single change-point:

- at position $u = \alpha p$
- with increments $(\beta_i)_{i=1,\ldots,n}$ s.t. $\bar{\beta}^2 = \lim_{k\to\infty} \frac{1}{n}\sum_{i=1}^{n} \beta_i^2$
- corrupted by i.i.d. Gaussian noise of variance σ^2



Does the TV approximator correctly estimate the first change-point as p increases?

Consistency of the weighted TV approximator

$$\min_{U \in \mathbb{R}^{p \times n}} \| Y - U \|^2 \quad \text{such that} \quad \sum_{i=1}^{p-1} w_i \| U_{i+1,\bullet} - U_{i,\bullet} \| \leq \mu$$

Theorem

The weighted TV approximator with weights

$$\forall i \in [1, p-1] \,, \quad w_i = \sqrt{\frac{i(p-i)}{p}}$$

correctly finds the first change-point with probability tending to 1 *as* $n \to +\infty$.

- we see the benefit of increasing n
- we see the benefit of adding weights to the TV penalty

Consistency for a single change-point

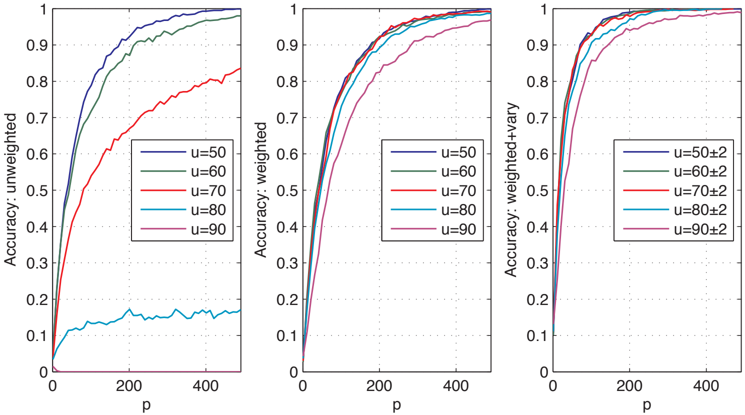


Figure 3: **Single change-point accuracy for the group fused Lasso.** Accuracy as a function of the number of profiles p when the change-point is placed in a variety of positions $u = 50$ to $u = 90$ (left and centre plots, resp. unweighted and weighted group fused Lasso), or: $u = 50 \pm 2$ to $u = 90 \pm 2$ (right plot, weighted with varying change-point location), for a signal of length 100.

Estimation of several change-points

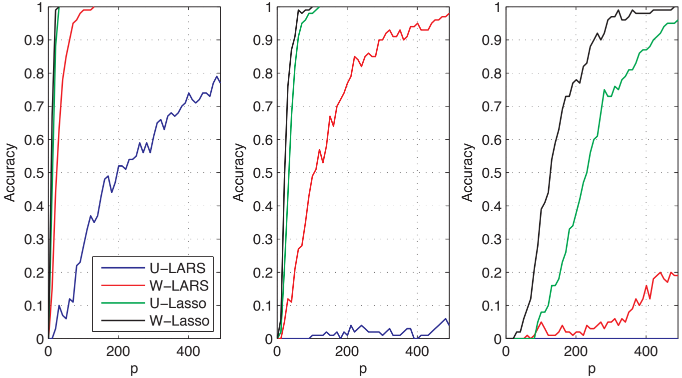
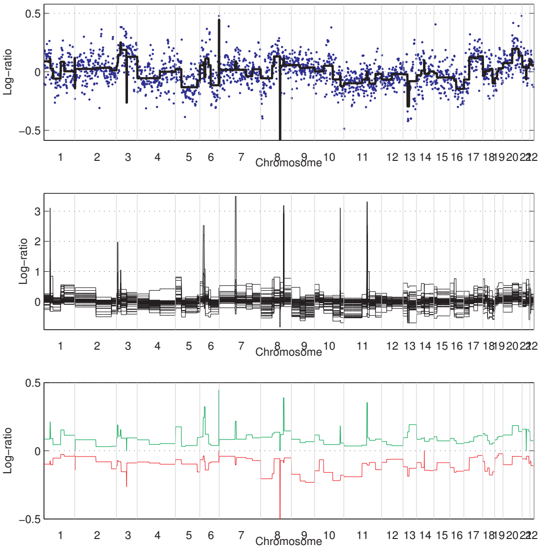


Figure 4: **Multiple change-point accuracy.** Accuracy as a function of the number of profiles p when change-points are placed at the nine positions $\{10, 20, \dots, 90\}$ and the variance σ^2 of the centered Gaussian noise is either 0.05 (left), 0.2 (center) and 1 (right). The profile length is 100.

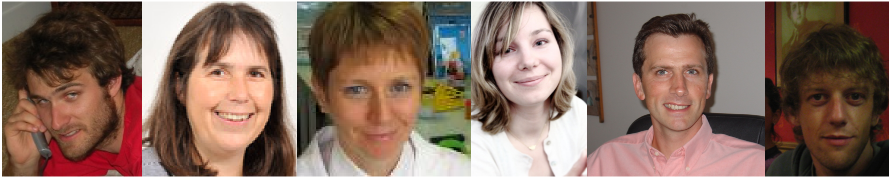
Application: detection of frequent abnormalities



Conclusion

- Partial expert annotation can be done efficiently to benchmark and optimize breakpoint detection methods
- Popular methods and default parameters are often not very good
- Multiparametric optimization can be formulated as interval regression
- Fast segmentation method for long, multiple signals

Acknowledgements



Toby Hocking (Tokodai), Gudrun Schleiermacher, Isabelle Janoueix and Valentina Boeva (Institut Curie), Francis Bach and Kevin Bleakley (INRIA)