Frontiers in optimal change-point detection algorithms

Toby Dylan Hocking — toby.dylan.hocking@usherbrooke.ca
Université de Sherbrooke, Département d'Informatique
Joint work with Charles Truong, Guillem Rigaill, Vincent Runge

1. Introduction, applications, recent algorithms

2. Constraints between adjacent segment means

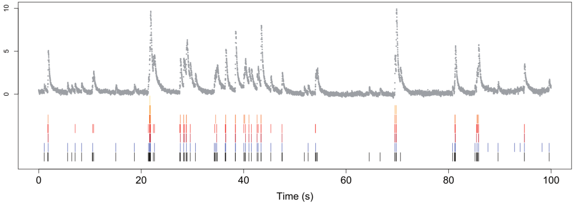
3. Optimal changepoints subject to label constraints

4. Learning to predict the number of changepoints

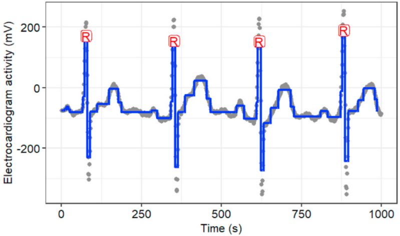
Summary and Discussion

Changepoint detection algorithms for data over time

Neuron spikes, Jewell *et al.*, Biostatistics 2019.

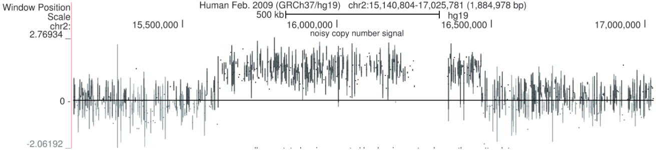


Electrocardiograms (heart monitoring), Fotoohinasab *et al.*, Asilomar conference 2020.

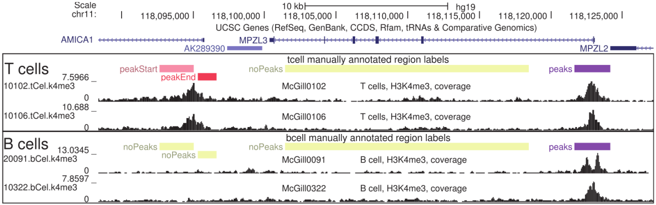


Changepoint detection algorithms for data over space

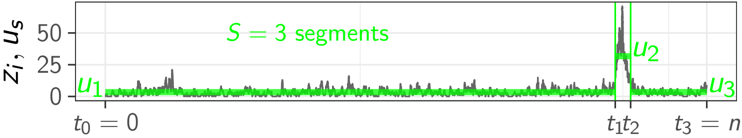
DNA copy number data for cancer diagnosis, Hocking *et al.*, Bioinformatics 2014.



ChIP-seq data for annotating active/inactive regions in different cell types, Hocking *et al.*, Bioinformatics 2017.



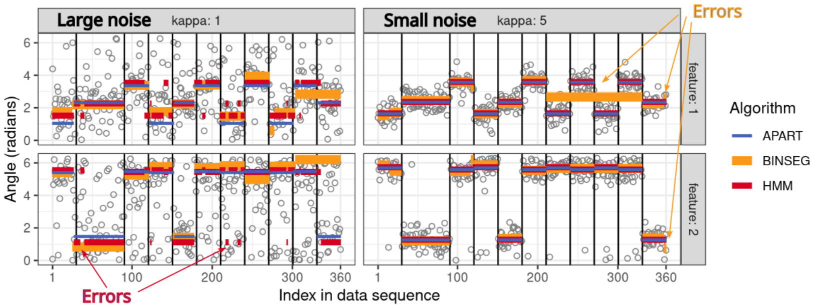
Optimal changepoint detection problem



$$\min_{\substack{\mathbf{u} \in \mathbb{R}^S \\ 0=t_0<t_1<\cdots<t_{S-1}<t_S=n}} \sum_{s=1}^{S} \sum_{i=t_{s-1}+1}^{t_s} \ell(u_s, z_i)$$

- Algorithm inputs n data $z_1, \dots, z_n$ and # of segments S.
- Loss function ℓ measures fit of means u_s to data z_i.
- Goal is to compute best $S - 1$ changepoints $t_1 < \cdots < t_{S-1}$ and S segment parameters $u_1, \dots, u_S$.
- Non-convex optimization problem, naïvely $O(n^S)$ time.
- Fast approximate heuristic algorithms?

Fast heuristics can yield inaccurate change-points



- Simulation: angular data, 2 dimensions/features.
- Heuristics are approximate optimization algorithms.
- Fast: linear $O(n)$ time for n data.
- Not guaranteed to compute change-points with best loss.
- BINSEG = Binary segmentation (Scott and Knott, 1974).
- HMM = Hidden Markov Model (Rabiner, 1989).

Constrained problem, Lasso heuristic

Constrained optimal change-point problem:

$$\min_{\mu\in\mathbb{R}^n} \underbrace{\sum_{i=1}^n \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} \quad \text{subject to} \quad \underbrace{\sum_{i=2}^n I[\mu_{i-1} \neq \mu_i] \leq S.}_{\text{Number of change-points (regularization)}}$$

- Hyper-parameter is number of segments S.
- Indicator function $I[\cdot] \in \{0, 1\}$ is non-convex.
- Gradient descent yields local min.
- Analog with best subset regression: hyper-parameter is number of variables (combinatorial search, relax to Lasso).
- Fused Lasso for change-points (Tibshirani *et al.*, JRSSB 2005).
- Lasso selects too many variables (false positive change-points).

Constrained problem, classic dynamic programming solution

Constrained optimal change-point problem:

$$\min_{\mu\in\mathbb{R}^n} \underbrace{\sum_{i=1}^n \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} \quad \text{subject to} \quad \underbrace{\sum_{i=2}^n I[\mu_{i-1} \neq \mu_i] \leq S.}_{\text{Number of change-points (regularization)}}$$

- Auger and Lawrence, Algorithms for the optimal identification of **segment neighborhoods**, Bull Math Biol (1989).
- Optimal recursive updates (dynamic programming algorithm).
- Let $C_{n,S}$ be best cost up to n data and S segments.
- Start by computing $C_{1,1}$ to $C_{1,n}$ (cum sum).
- Then compute $C_{2,2}$ to $C_{2,n}$, $C_{3,3}, \ldots, C_{3,n}$, etc.
- Output all optimal models from 1 to S segments.
- Time complexity $O(Sn^2)$ — faster than naïve $O(n^S)$.
- Still too slow for large data n and model sizes S.

Penalized problem, classic dynamic programming solution

Penalized optimal change-point problem:

$$\min_{\mu \in \mathbb{R}^n} \underbrace{\sum_{i=1}^n \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} + \underbrace{\lambda \sum_{i=2}^n I[\mu_{i-1} \neq \mu_i]}_{\text{Number of change-points (regularization)}}.$$

- Hyper-parameter is non-negative penalty $\lambda \geq 0$.
- Jackson, *et al.*, An algorithm for **optimal partitioning** of data on an interval, IEEE Sig Proc Lett (2005).
- Optimal recursive updates (dynamic programming algorithm).
- Let C_n be best cost up to n data.
- Recursively compute $C_1, C_2, \dots, C_n$.
- Time complexity $O(n^2)$ — faster than $O(Sn^2)$.
- User can not directly specify number of segments S.
- Output one optimal change-point model (not all in $1, \dots, S$).
- Quadratic time is still too slow for large data n.

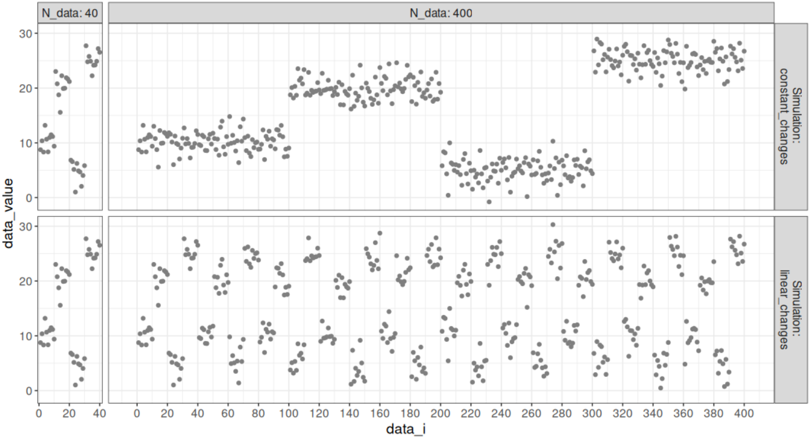
Penalized problem, PELT algorithm

Penalized optimal change-point problem:

$$\min_{\mu \in \mathbb{R}^n} \underbrace{\sum_{i=1}^{n} \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} + \underbrace{\lambda \sum_{i=2}^{n} I[\mu_{i-1} \neq \mu_i]}_{\text{Number of change-points (regularization)}}.$$

- Recursively compute $C_1, C_2, \dots, C_n$.
- Classic DP considers all previous change-points.
- At time n, consider $C_1, \dots, C_{n-1}$.
- Pruned Exact Linear Time algo: Killick, *et al.*, JASA (2012).
- At time n consider only a subset of $C_1, \dots, C_{n-1}$.
- Easy to implement, with only 1 additional line of code!
- Same output: one optimal change-point model.
- Time complexity: best $O(n)$, worst $O(n^2)$.

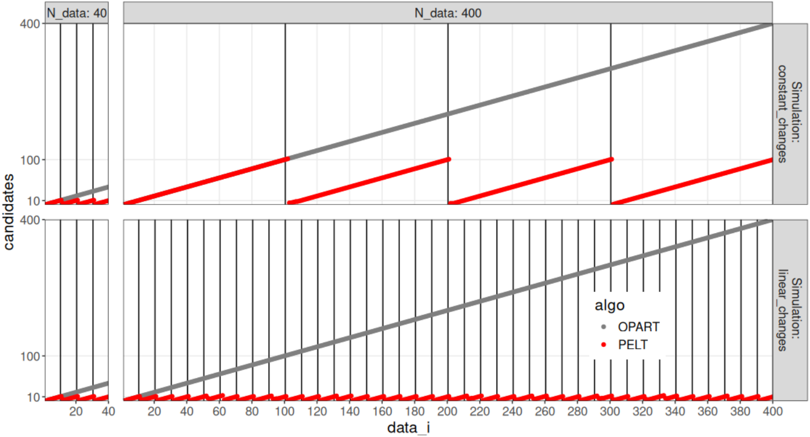
Simulated data with constant and linear changes



- Constant changes: always 4 segments (for any data size n).
- Linear: change every 10 data.

https://tdhock.github.io/blog/2025/PELT-vs-fpopw/

PELT algorithm demonstration



- After each change in data, PELT prunes prior candidates.
- More changes—more pruning—faster.

`https://tdhock.github.io/blog/2025/PELT-vs-fpopw/`

Penalized problem, PELT algorithm

Penalized optimal change-point problem:

$$\min_{\mu \in \mathbb{R}^n} \underbrace{\sum_{i=1}^{n} \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} + \underbrace{\lambda \sum_{i=2}^{n} I[\mu_{i-1} \neq \mu_i]}_{\text{Number of change-points (regularization)}}.$$

# changes in data	Algorithm	# candidates per data	Overall time
Constant $O(1)$	OPART	$O(n)$	$O(n^2)$
	PELT	$O(n)$	$O(n^2)$
Linear $O(n)$	OPART	$O(n)$	$O(n^2)$
	PELT	$O(1)$	$O(n)$

- PELT fast/linear for data with frequent changes.
- But slow/quadratic for long runs of data without changes.
- Can we go faster when there are no changes?

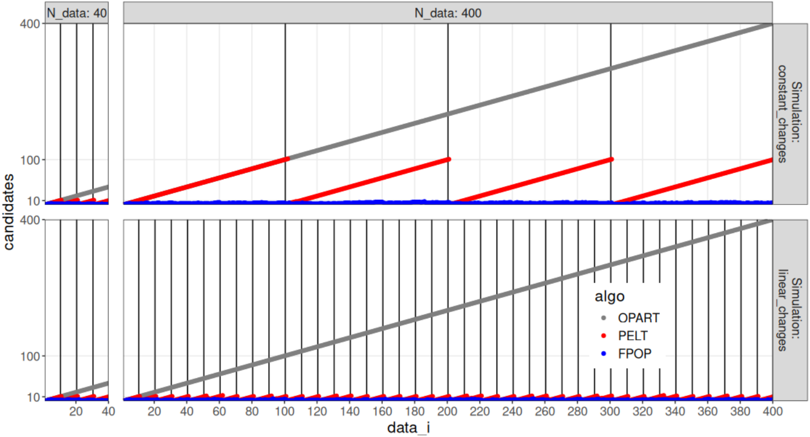
Yes, faster with FPOP algorithm!

Penalized optimal change-point problem:

$$\min_{\mu \in \mathbb{R}^n} \underbrace{\sum_{i=1}^n \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} + \underbrace{\lambda \sum_{i=2}^n I[\mu_{i-1} \neq \mu_i]}_{\text{Number of change-points (regularization)}}.$$

- FPOP algo: Maidstone, *et al.*, Stat. and Comp. (2017).
- Let $C_n(m)$ be the best cost with mean m at n data.
- Recursively compute functions $C_1(m), C_2(m), \dots, C_n(m)$.
- **Functional pruning** considers a small subset of candidates.
- Same output: one optimal change-point model.
- Same time as PELT in theory: best $O(n)$, worst $O(n^2)$.
- But much faster in practice!

FPOP algorithm demonstration



- FPOP always prunes, whether or not there are changes in data.

https://tdhock.github.io/blog/2025/PELT-vs-fpopw/

Penalized problem, FPOP algorithm

Penalized optimal change-point problem:

$$\min_{\mu\in\mathbb{R}^n} \underbrace{\sum_{i=1}^n \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} + \underbrace{\lambda \sum_{i=2}^n I[\mu_{i-1} \neq \mu_i]}_{\text{Number of change-points (regularization)}}.$$

# changes in data	Algorithm	# candidates per data	Overall time
Constant $O(1)$	OPART	$O(n)$	$O(n^2)$
	PELT	$O(n)$	$O(n^2)$
	FPOP	$O(\log n)$	$O(n \log n)$
Linear $O(n)$	OPART	$O(n)$	$O(n^2)$
	PELT	$O(1)$	$O(n)$
	FPOP	$O(1)$	$O(n)$

- FPOP always fast in 1d data, no matter how many changes.
- Worst case $O(n^2)$ only happens in pathological data.

Penalized problem, FPOP algorithm

Penalized optimal change-point problem:

$$\min_{\mu\in\mathbb{R}^n} \underbrace{\sum_{i=1}^n \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} + \underbrace{\lambda \sum_{i=2}^n I[\mu_{i-1} \neq \mu_i]}_{\text{Number of change-points (regularization)}}.$$

- FPOP algo: Maidstone, *et al.*, Stat. and Comp. (2017).
- Let $C_n(m)$ be the best cost with mean m at n data.
- Recursively compute functions $C_1(m), C_2(m), \ldots, C_n(m)$.
- **Functional pruning** considers a small subset of candidates.
- Same output: one optimal change-point model.
- Time complexity best $O(n)$, worst $O(n^2)$.

Penalized problem, FPOP extensions

Functional pruning can handle:

- Inequality constraints between segment means (next section).
- Robust loss functions: Fearnhead and Rigaill, JASA (2019).
- Auto-regressive models: Romano *et al.*, JASA (2021).
- Storage on disk: Hocking *et al.*, JSS (2022).
- Multi-variate time series: Pishchagina *et al.*, Computo (2024).
- Multi-scale penalties: Liehrmann and Rigaill, JCGS (2025).

But:

- Difficult to implement (100s of lines of C++ code).
- Not as fast in high dimensional data/models.

Is it possible to go faster?

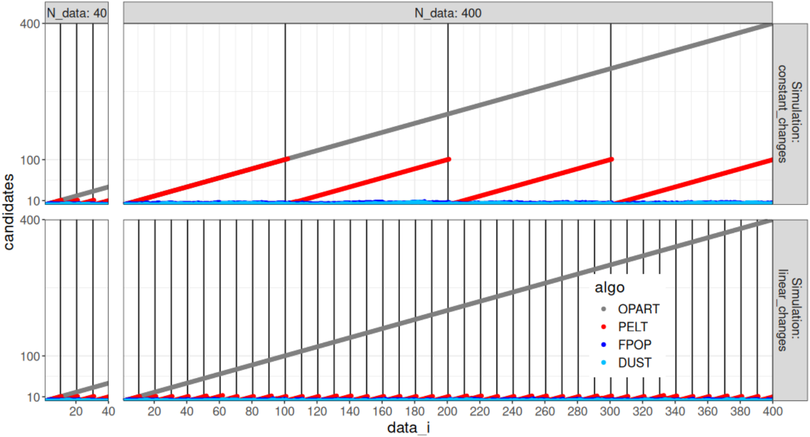
Yes, faster with DUST algorithm!

Penalized optimal change-point problem:

$$\min_{\mu \in \mathbb{R}^n} \underbrace{\sum_{i=1}^{n} \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} + \underbrace{\lambda \sum_{i=2}^{n} I[\mu_{i-1} \neq \mu_i]}_{\text{Number of change-points (regularization)}}.$$

- Truong and Runge, An Efficient Algorithm for Exact Segmentation of Large Compositional and Categorical Time Series, Stat (2024).
- DUST algo: DUality Sample Test.
- Combines ideas from PELT and FPOP algos.
- Solve a Langrange dual problem to prune change-points.
- Easier to code than FPOP.
- Same time as PELT/FPOP in theory: best $O(n)$, worst $O(n^2)$.
- But much faster in practice!

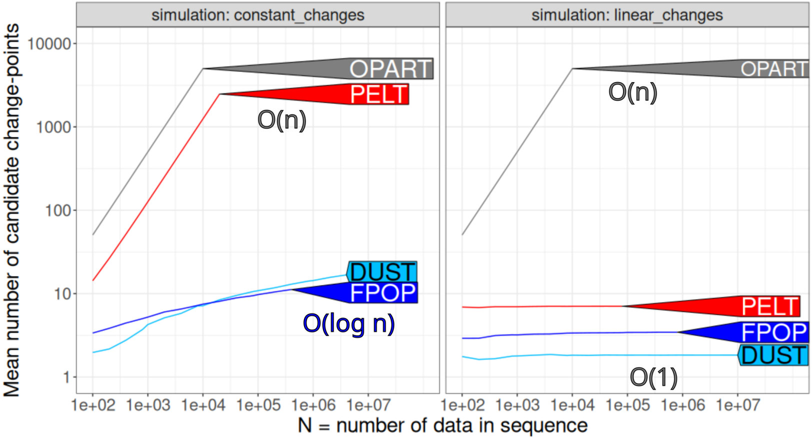
DUST algorithm demonstration



- DUST always prunes, whether or not there are changes in data.

https://tdhock.github.io/blog/2025/PELT-vs-fpopw/

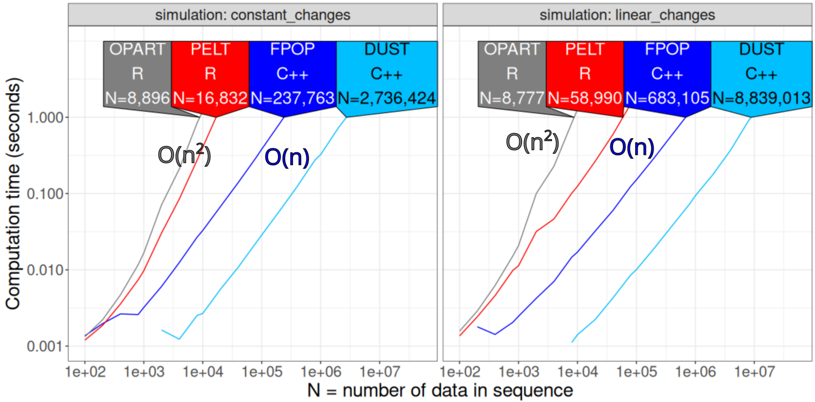
Number of candidates considered



- Data sizes varied from 100 to 10,000,000.
- Slope on log-log plot indicates asymptotic time complexity.

`https://tdhock.github.io/blog/2025/PELT-vs-fpopw/`

Overall computation time



- N= data size computable in time limit of 1 second.
- FPOP 10x faster than PELT.
- DUST 10x faster than FPOP.

https://tdhock.github.io/blog/2025/PELT-vs-fpopw/

Penalized problem, DUST algorithm

Penalized optimal change-point problem:

$$\min_{\mu \in \mathbb{R}^n} \underbrace{\sum_{i=1}^{n} \ell(\mu_i, z_i)}_{\text{Loss (data-fitting)}} + \underbrace{\lambda \sum_{i=2}^{n} I[\mu_{i-1} \neq \mu_i]}_{\text{Number of change-points (regularization)}}.$$

# changes in data	Algorithm	# candidates per data	Overall time
Constant $O(1)$	OPART	$O(n)$	$O(n^2)$
	PELT	$O(n)$	$O(n^2)$
	FPOP	$O(\log n)$	$O(n \log n)$
	DUST	$O(\log n)$	$O(n \log n)$
Linear $O(n)$	OPART	$O(n)$	$O(n^2)$
	PELT	$O(1)$	$O(n)$
	FPOP	$O(1)$	$O(n)$
	DUST	$O(1)$	$O(n)$

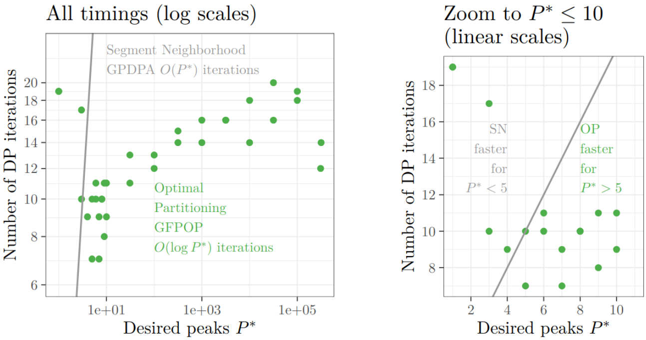
- DUST always fast in 1d data, no matter how many changes.
- Worst case $O(n^2)$ only happens in pathological data.

Computing various solutions using penalized solver

How to compute the best model with a given number P^* of change-points?

- Example: $P^* = 100$ change-points?
- Could run constrained solver to get all models from 0 to P^* change-points.
- Required number of DP iterations is linear in number of change-points, $O(P^*)$—slow if P^* large.
- Penalized solver returns best change-points for a given penalty $\lambda \geq 0$ (but λ that yields P^* is unknown).
- Sequential search: Hocking *et al.*, Journal of Statistical Software 101(10) (2022).
- DP iterations logarithmic in number of change-points, $O(\log P^*)$—fast for large P^*!

Penalized (OP) is faster than constrained (SN)



- Figure: genomic data, $N \approx 10^6$,
- Sequential search repeated runs penalized (OP) solver.
- Example: for $N = 10^7$, desired change-points $P^* = 3000$.
- Constrained (SN) solver: 100TB storage, 10 weeks.
- Penalized (OP) solver: 100GB storage, 10 hours.

Hocking *et al.*, Journal of Statistical Software 101(10) (2022).

Computing ranges of solutions using penalized solver

How to compute all models with penalties $\lambda \in [\underline{\lambda}, \overline{\lambda}]$?

- Example: $\lambda \in [0.1, \overline{1}0.5]$?
- CROPS: Change-points for a Range Of PenaltieS.
- Haynes, *et al.* Journal of Computational and Graphical Statistics, 26(1), 134-143 (2017).

How to compute all models with number of change-points $P \in [\underline{P}, \overline{P}]$?

- Example: all models from 50 to 60 change-points?
- CROCS: Change-points for a Range Of ComplexitieS.
- Liehrmann *et al.*, BMC Bioinformatics 22(323) (2021).

If M is the number of models to compute (ex: 11 models from 50 to 60 change-points), then $O(M + \log \overline{P})$ DP iterations—fast for large model sizes $\overline{P}$.

1. Introduction, applications, recent algorithms

2. Constraints between adjacent segment means

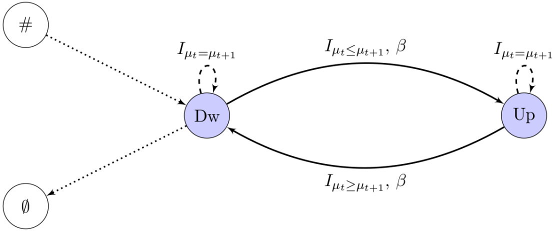
3. Optimal changepoints subject to label constraints

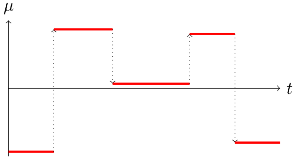
4. Learning to predict the number of changepoints

Summary and Discussion

Optimization constraints defined using a graph

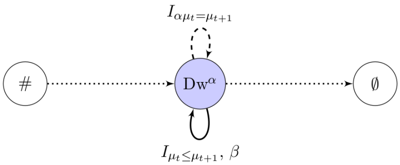
Runge *et al.*, Journal of Statistical Software 2023 (graph figures).

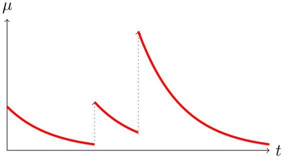




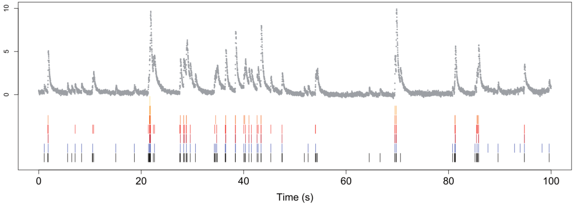
- Purple Dw/Up nodes represent hidden states.
- #/∅ nodes constrain start/end state.
- Edges represent possible state transitions.
- `gfpop` R package with C++ code computes optimal changepoints for user-defined constraint graphs.

All up changes, exponential decaying segments

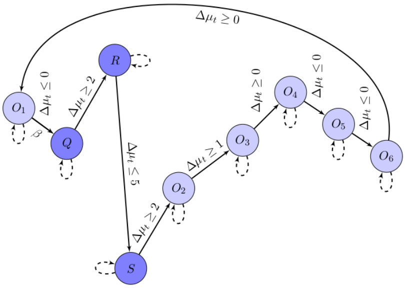


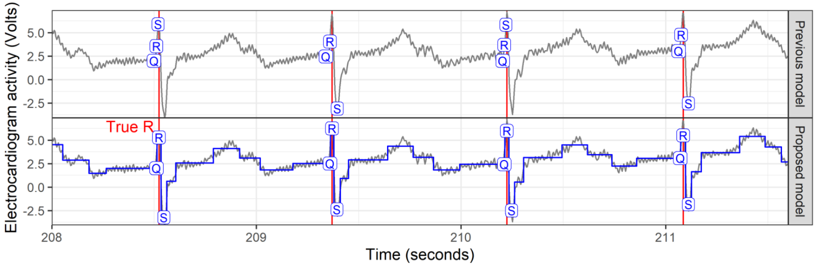


Jewell *et al.*, Biostatistics 2019.



Complex graph for electrocardiogram data





Fotoohinasab *et al.*, Asilomar conference 2020.

1. Introduction, applications, recent algorithms

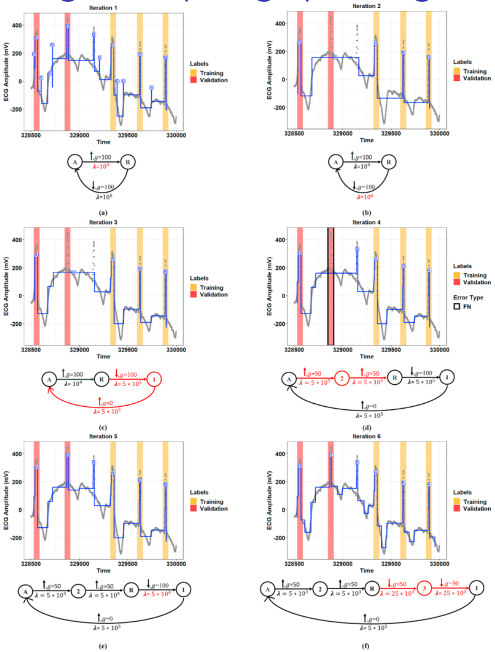
2. Constraints between adjacent segment means

3. Optimal changepoints subject to label constraints

4. Learning to predict the number of changepoints

Summary and Discussion

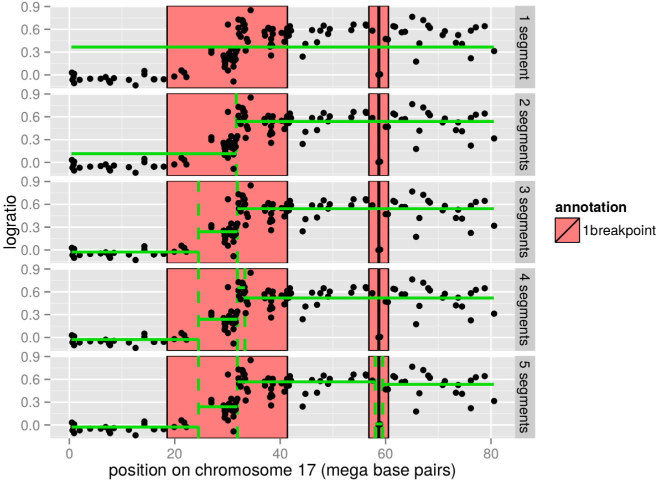
Learning a complex graph using labels



- Fotoohinasab *et al.*, 2021.
- Simple initial graph is iteratively edited (red) to agree with expert labeled regions (orange rectangles).
- Easier for expert to provide labels than graph.

What if no models agree with expert labels?

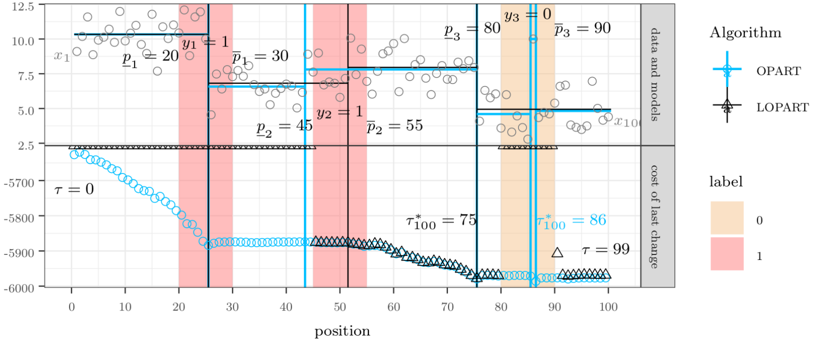
Hocking and Rigaill, Pre-print hal-00759129.



- Expert wants: one changepoint in each label (red rectangle).
- No model is consistent with all three labels.

Using expert labels as optimization constraints

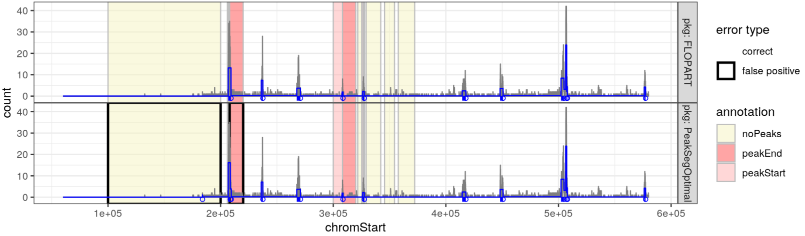
Hocking and Srivastava, Computational Statistics 38 (2023).



- Previous OPART model (blue) ignores labels (two errors).
- Main idea: add optimization constraints to ensure that there is the right number of changepoints predicted in each label.
- Proposed LOPART model (black) consistent with labels.

Label constraints and directional constraints

Kaufman *et al.*, Journal of Computational and Graphical Statistics 33(4) (2024).



- Previous PeakSegOptimal algorithm (bottom) ignores labels (two errors).
- Proposed FLOPART model (top) consistent with labels, and interpretable in terms of up changes to peaks and down changes to background noise.

1. Introduction, applications, recent algorithms

2. Constraints between adjacent segment means

3. Optimal changepoints subject to label constraints

4. Learning to predict the number of changepoints

Summary and Discussion

How to predict the number of changes?

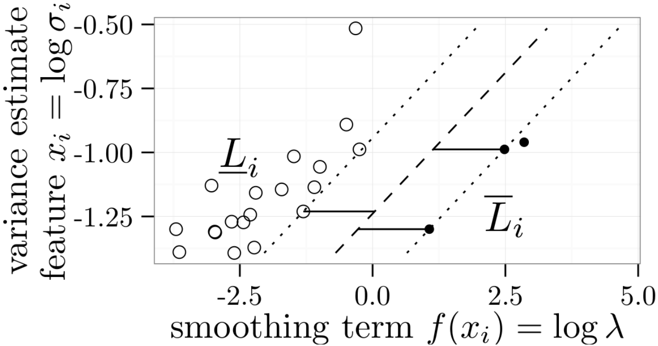
We assumed that the number of segments S is provided as an input parameter to our optimization algorithm.

$$\min_{\substack{\mathbf{u}\in\mathbb{R}^S \\ 0=t_0<t_1<\cdots<t_{S-1}<t_S=n}} \sum_{s=1}^{S} \sum_{i=t_{s-1}+1}^{t_s} \ell(u_s, z_i)$$

In practice S is often unknown — what value should we use?

Learning to predict number of changes similar to SVM

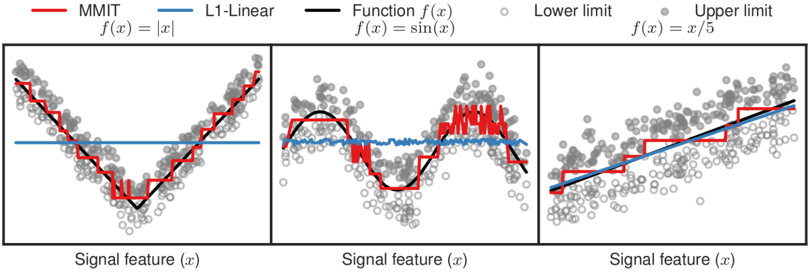
Hocking *et al.*, Int'l Conference on Machine Learning 2013.



- Train on several data sequences with labels (dots).
- Want to compute function between white and black dots.
- SVM margin is multi-dimensional (diagonal).
- Here margin to maximize is one-dimensional (horizontal).
- Learned function predicts number of changepoints/segments.

Decision tree learns non-linear function of inputs

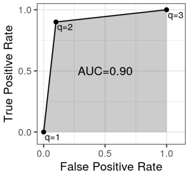
Drouin *et al.*, Neural Information Processing Systems 2017.



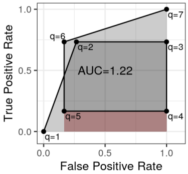
- Generalization of classical CART regression tree learning algorithm.
- Can learn non-linear functions of inputs.
- More recently we implemented a similar idea in xgboost, Barnwal *et al.*, Journal of Computational and Graphical Statistics 31(4) (2022).

Is maximizing Area Under the ROC Curve desirable?

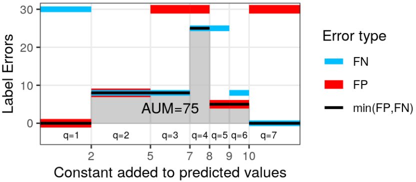
In binary classification the ROC curve is monotonic.



In changepoint detection it can have loops.

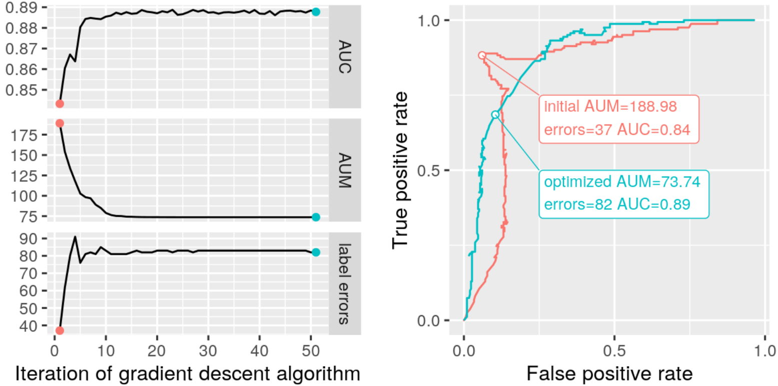


We propose instead to minimize the AUM = Area Under the Minimum of false positives and false negatives, as a function of prediction threshold.



AUM gradient descent algorithm optimizes AUC

Hillman and Hocking, in progress.



- Initial predictions: minimum label errors.
- ROC curves become more regular/monotonic after optimization, but label error increases.
- Trade-off between AUC and label error optimization that does not exist in binary classification.

1. Introduction, applications, recent algorithms
2. Constraints between adjacent segment means
3. Optimal changepoints subject to label constraints
4. Learning to predict the number of changepoints

Summary and Discussion

Summary and Discussion

- Optimal changepoint detection in n data is a non-convex problem, naïvely a $O(n^S)$ computation for S segments.
- Recent algorithms can compute a globally optimal changepoint model much faster, $O(n \log n)$.
- Directional constraint graphs specified using domain prior knowledge, or learned using expert labels.
- Expert labels can also be used as optimization constraints, to ensure that predicted changepoints are consistent.
- Number of changes can be predicted with new learning algorithms, including ROC curve optimization.
- Let's collaborate! toby.hocking@nau.edu

References

- Auger IE and Lawrence CE. Algorithms for the optimal identification of segment neighborhoods. Bull Math Biol 51:39–54 (1989).
- G Rigaill. A pruned dynamic programming algorithm to recover the best segmentations with 1 to kmax change-points. Journal de la Société Française de la Statistique, 156(4), 2015.
- **Hocking TD**, Boeva V, Rigaill G, Schleiermacher G, Janoueix-Lerosey I, Delattre O, Richer W, Bourdeaut F, Suguro M, Seto M, Bach F, Vert J-P. SegAnnDB: interactive Web-based genomic segmentation. Bioinformatics (2014) 30 (11): 1539-1546.
- **Hocking TD**, Goerner-Potvin P, Morin A, Shao X, Pastinen T, Bourque G. Optimizing ChIP-seq peak detectors using visual labels and supervised machine learning. Bioinformatics (2017) 33 (4): 491-499.
- Jewell S, **Hocking TD**, Fearnhead P, Witten D. Fast Nonconvex Deconvolution of Calcium Imaging Data. Biostatistics (2019).
- Fotoohinasab A, **Hocking TD**, Afghah F. A Graph-Constrained Changepoint Learning Approach for Automatic QRS-Complex Detection. Asilomar Conference on Signals, Systems, and Computers (2020).
- **Hocking TD**, Rigaill G, Fearnhead P, Bourque G. Constrained Dynamic Programming and Supervised Penalty Learning Algorithms for Peak Detection in Genomic Data. Journal of Machine Learning Research 21(87):1–40, 2020.
- Runge V, **Hocking TD**, Romano G, Afghah F, Fearnhead P, Rigaill G. gfpop: an R Package for Univariate Graph-Constrained Change-point Detection. Journal of Statistical Software 106(6) (2023).

References

- **Hocking TD**, Rigaill G, Bach F, Vert J-P. Learning sparse penalties for change-point detection using max-margin interval regression. International Conference on Machine Learning 2013.
- Drouin A, **Hocking TD**, Laviolette F. Maximum margin interval trees. Neural Information Processing Systems 2017.
- Barnwal A, Cho H, **Hocking TD**. Survival regression with accelerated failure time model in XGBoost. Journal of Computational and Graphical Statistics 31(4) (2022).
- Hillman J, **Hocking TD**. Optimizing ROC Curves with a Sort-Based Surrogate Loss Function for Binary Classification and Changepoint Detection. Journal of Machine Learning Research 24(70) (2023).

References

- **Hocking TD**, Rigaill G. SegAnnot: an R package for fast segmentation of annotated piecewise constant signals, Pre-print hal-00759129.
- **Hocking TD**, Srivastava A. Labeled Optimal Partitioning. Computational Statistics 38 (2023).
- Fotoohinasab A, **Hocking TD**, Afghah F. A Greedy Graph Search Algorithm Based on Changepoint Analysis for Automatic QRS-Complex Detection. Computers in Biology and Medicine 130 (2021).