# Algorithmes efficaces pour la détection de ruptures

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

Brief history of optimal change-point algorithms

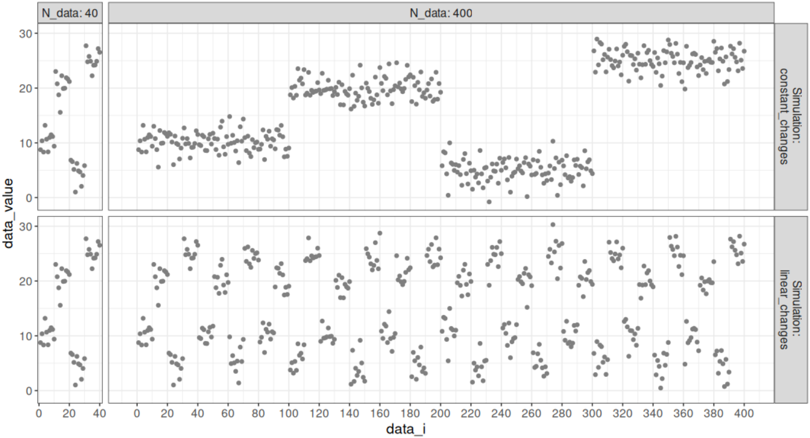
New algorithms with constraints between adjacent segments

Computing optimal changepoints subject to label constraints

Learning to predict the number of changepoints

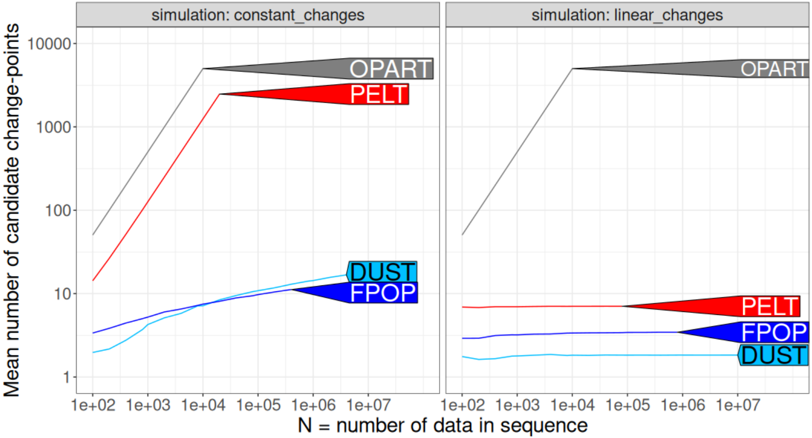
Summary and Discussion

# Simulated data with constant and linear changes



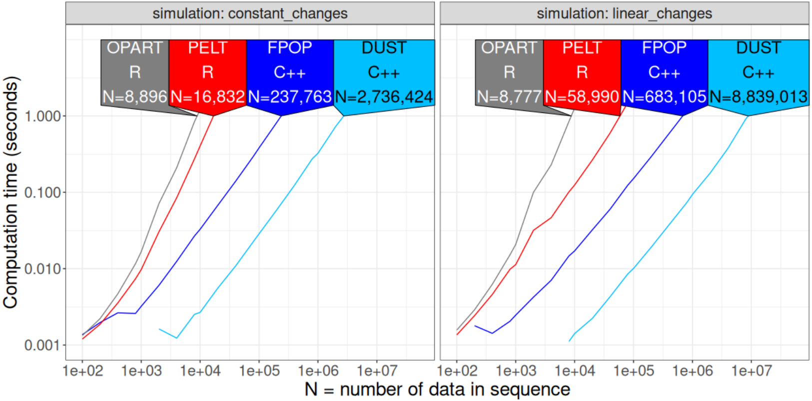

https://tdhock.github.io/blog/2025/PELT-vs-fpopw/

# Number of candidates considered




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

# Overall computation time



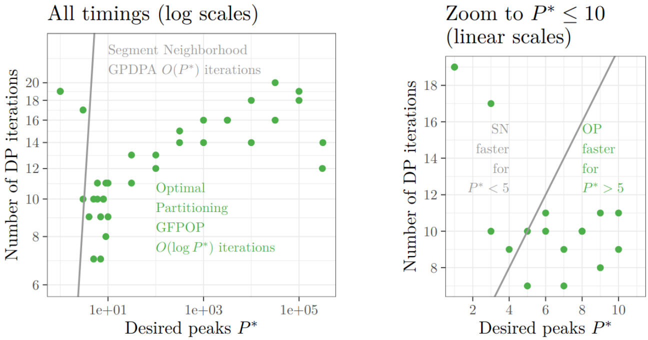

https://tdhock.github.io/blog/2025/PELT-vs-fpopw/

# 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 $\lambda$ 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 $\overline{P}$.

Brief history of optimal change-point algorithms

New algorithms with constraints between adjacent segments

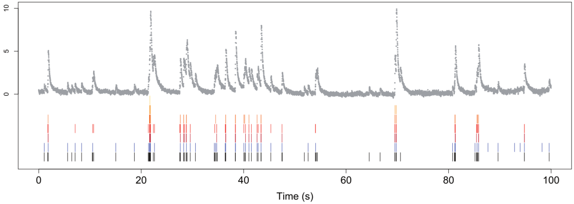
Computing optimal changepoints subject to label constraints

Learning to predict the number of changepoints

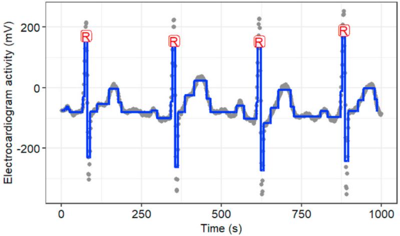
Summary and Discussion

# Changepoint detection algorithms for data over time

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

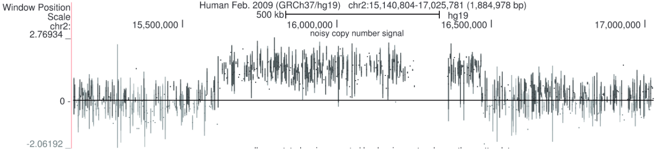



Electrocardiograms (heart monitoring), Fotoohinasab *et al.*, Asilomar conference 2020.

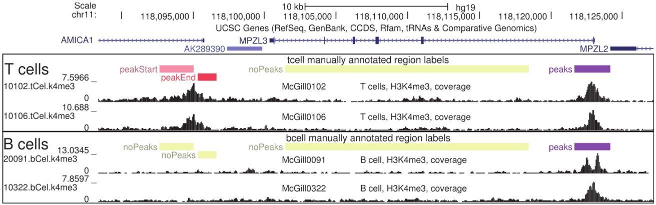


# Changepoint detection algorithms for data over space

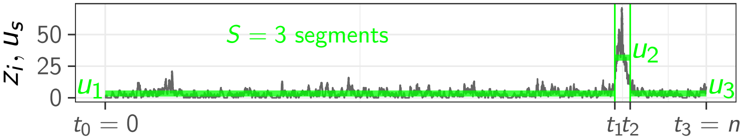
DNA copy number data for cancer diagnosis, Hocking *et al.*, Bioinformatics 2014.




ChIP-seq data for understanding the human genome, Hocking *et al.*, Bioinformatics 2017.



# Optimal changepoint detection problem and algorithms




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

- Algorithm inputs $n$ data $z_1, \dots, z_n$ and # of segments $S$.
- Goal is to compute best $S-1$ changepoints $t_1 < \cdots < t_{S-1}$ and $S$ segment parameters $u_1, \dots, u_S$.
- Hard non-convex optimization problem, naïvely $O(n^S)$ time.
- Auger and Lawrence 1989: $O(Sn^2)$ time algorithm.
- Rigaill 2015: $O(n \log n)$ time, change in any direction.
- Hocking *et al.*, 2020: $O(n \log n)$, directional constraints.

# Constrained optimization algorithm speed

Hocking *et al.*, Journal of Machine Learning Research 2020.

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

$$\text{subject to} \quad u_{s-1} \leq u_s \ \forall s \in \{2, 4, \dots\},$$
$$u_{s-1} \geq u_s \ \forall s \in \{3, 5, \dots\}.$$

Constraints used to force change up to peak state, then change down to background noise state.

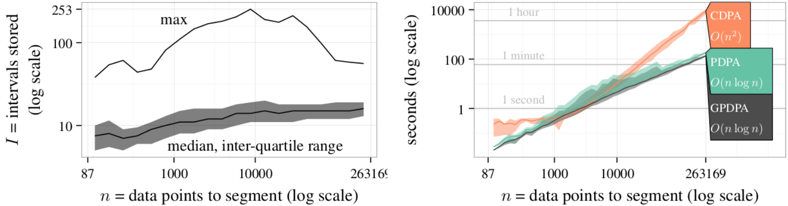
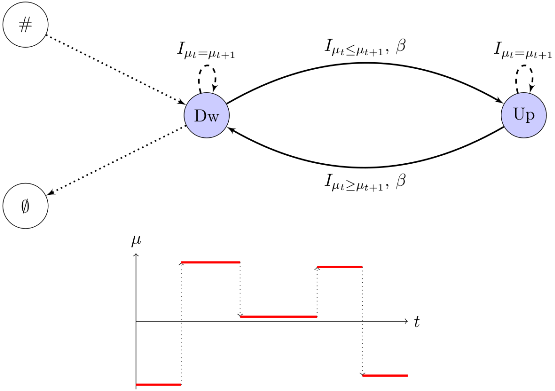



Figure 3: Empirical speed analysis on 2752 count data vectors from the histone mark ChIP-seq benchmark. For each vector we ran the GPDPA with the up-down constraint and a max of $K = 19$ segments. The expected time complexity is $O(KnI)$ where $I$ is the average number of intervals (function pieces; candidate changepoints) stored in the $C_{k,t}$ cost functions. **Left**: number of intervals stored is $I = O(\log n)$ (median, inter-quartile range, and maximum over all data points $t$ and segments $k$). **Right**: time complexity of the GPDPA is $O(n \log n)$ (median line and min/max band).

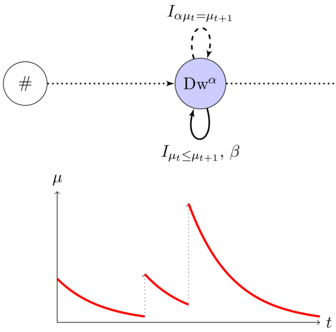
# Optimization constraints defined using a graph

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

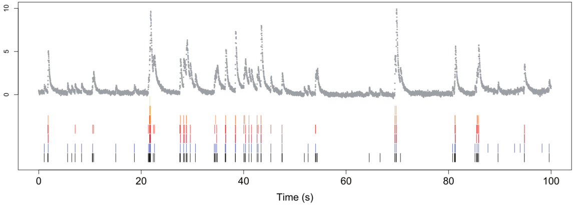



- Purple Dw/Up nodes represent hidden states.
- #/∅ nodes constrain start/end state.
- Edges represent possible state transitions.
- `gfpop` R package with C++ code computes optimal changepoints for user-defined constraint graphs.

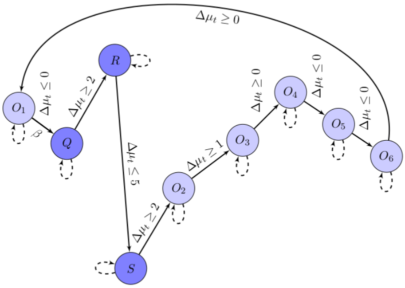
# All up changes, exponential decaying segments

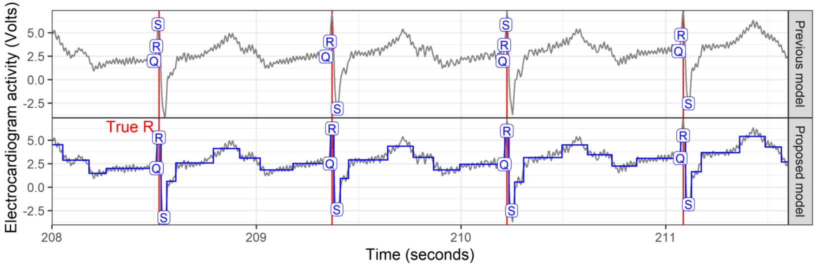



Jewell *et al.*, Biostatistics 2019.



# Complex graph for electrocardiogram data




Fotoohinasab *et al.*, Asilomar conference 2020.

# 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).

Brief history of optimal change-point algorithms

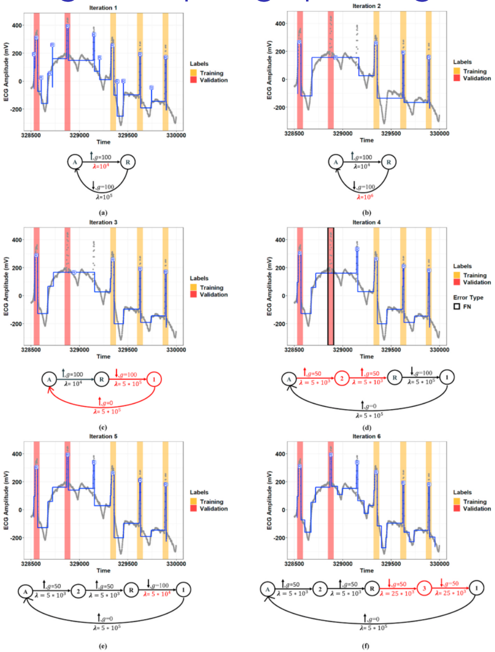
New algorithms with constraints between adjacent segments

Computing optimal changepoints subject to label constraints

Learning to predict the number of changepoints

Summary and Discussion

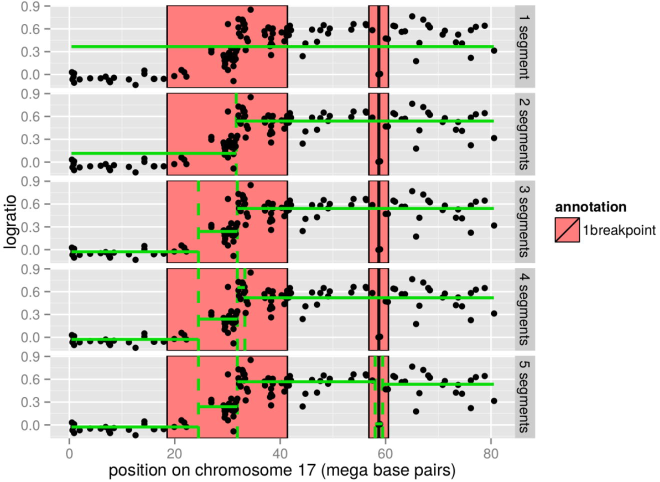
# Learning a complex graph using labels




- 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?

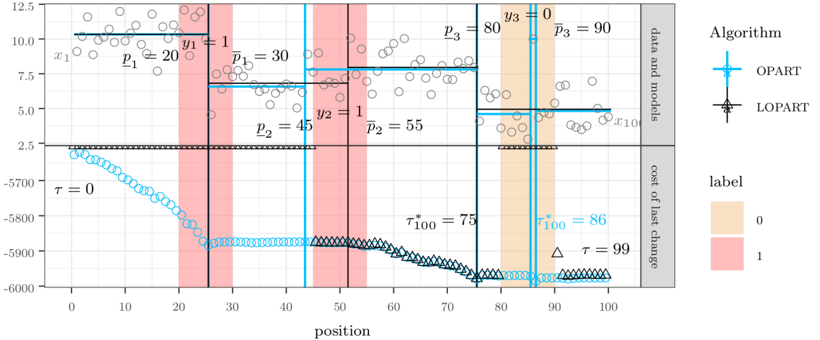
Hocking and Rigaill, Pre-print hal-00759129.




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

# Using expert labels as optimization constraints

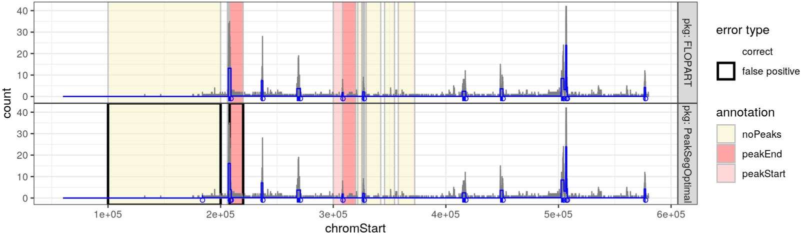
Hocking and Srivastava, Computational Statistics 38 (2023).




- 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

Stenberg and Hocking, in progress.




- 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.

# 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).

# How to predict the number of changes?

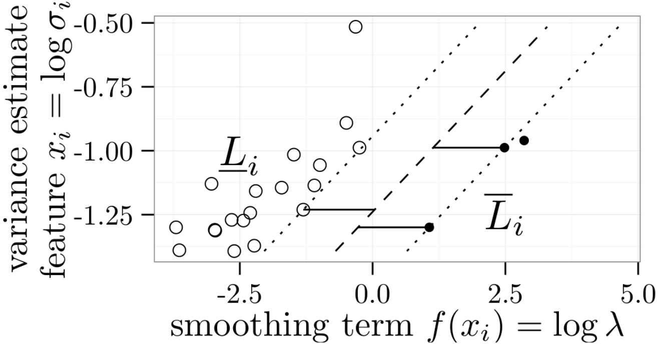
We assumed that the number of segments $S$ is provided as an input parameter to our optimization algorithm.

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

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

# Learning to predict number of changes similar to SVM

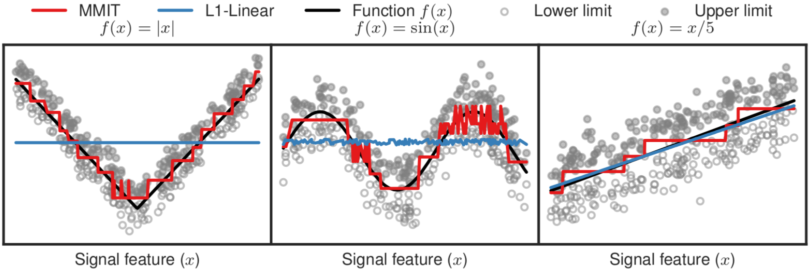
Hocking *et al.*, Int'l Conference on Machine Learning 2013.




- 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

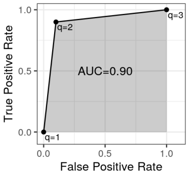
Drouin *et al.*, Neural Information Processing Systems 2017.



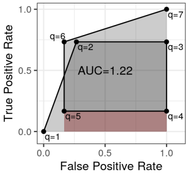

- Generalization of classical CART regression tree learning algorithm.
- Can learn non-linear functions of inputs.
- More recently we implemented a similar idea in xgboost, Barnwal *et al.*, Journal of Computational and Graphical Statistics 31(4) (2022).

# Is maximizing Area Under the ROC Curve desirable?

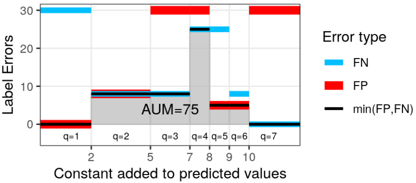
In binary classification the ROC curve is monotonic.




In changepoint detection it can have loops.

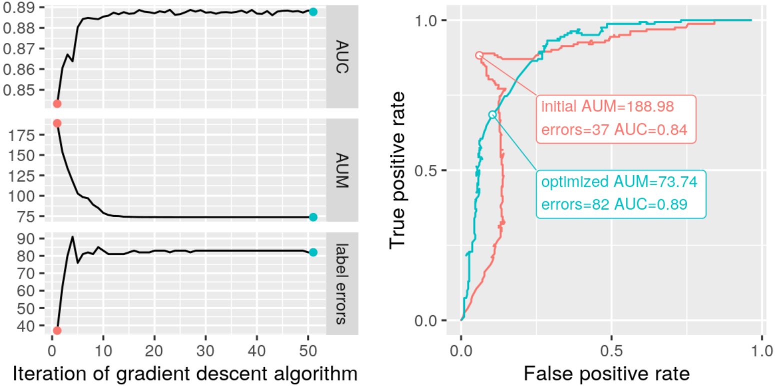



We propose instead to minimize the AUM = Area Under the Minimum of false positives and false negatives, as a function of prediction threshold.



# 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.

# 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).

Brief history of optimal change-point algorithms

New algorithms with constraints between adjacent segments

Computing optimal changepoints subject to label constraints

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