A New Efficient Algorithm for Exact Multiple Change Point Detection in Time Series

Simon Querné[1] and Vincent Runge[1]

Laboratoire de Mathématiques et Modélisation d'Evry, Université Paris-Saclay, CNRS, Univ Evry. simon.querne@ens.uvsq.fr

Abstract. We consider the problem of detecting multiple change points within time series by exactly optimizing a penalized likelihood. We introduce a novel and efficient algorithm called DUST, which combines dynamic programming with an inequality-based pruning step for discarding relevant indices. The talk will be focused on the presentation on the duality method for DUST along with simulation results highlighting its time efficiency in a large class of data models. We developed an Rcpp package utilizing polymorphic classes, which is available on GitHub `https://github.com/vrunge/dust`.

Keywords: Change Point Detection · Dynamic Programming · Duality Principle in Optimization · Pruning

1 The Pruning Efficiency Problem

Change point detection in time series is essential in fields like finance, genomics or climate change among many others. The goal is to identify points where the underlying distribution of the series changes. Although the OP (Optimal Partitioning) algorithm provides an exact solution for both univariate and multivariate series, its time complexity of $\mathcal{O}\left(n^2\right)$ makes it computationally expensive, especially for long series. For instance, optimizing a series of length 10^4 might take 10 seconds, while a series of 10^5 could take 17 minutes.

Pruning can significantly reduce the computational load of the algorithm. While the PELT algorithm [1] is a strong alternative to OP, it struggles with sparse change points. FPOP [2] excels with one-dimensional Gaussian data but falters in higher dimensions or non-Gaussian settings. Our proposed DUST algorithm builds on FPOP without these limitations.

We present a brief introduction to the change point problem for univariate data and the duality structure used by DUST. Preliminary simulations highlight its efficiency and versatility across exponential family cost models. The talk will extend these results to the multivariate case.

2 Introducing the DUST algorithm

2.1 The OP Algorithm

Consider some data set $y_{1:n} = (y_1, \dots, y_n) \in \mathcal{Y}^n$. Suppose there exist K change points in $y_{1:n}$, i.e. there exists an increasing vector of change points $\boldsymbol{\tau} = (0 = \tau_0, \tau_1, \dots, \tau_K, \tau_{K+1} = n)$ that defines $K+1$ segments made of iid data. We introduce the penalization function $m_K : \beta \mapsto K\beta$, and the cost function $\gamma : \mathcal{Y} \times \Theta \to \mathbb{R}$ where $\Theta \subset \mathbb{R}$ is some set of parameters, often the mean of the distribution. For Gaussian model and data point y, we have $\gamma(y, \theta) = (y - \theta)^2$. We find the number and location of the change points by computing:

$$Q_n := \min_{K, \boldsymbol{\tau} \in \mathcal{T}, \boldsymbol{\theta}} \left\{ \sum_{j=0}^{K} \left[\sum_{i=\tau_j+1}^{\tau_{j+1}} \gamma\left(y_i, \theta_j\right) + \beta \right] - \beta \right\} . \quad (1)$$

The OP algorithm divides this problem into n embedded sub-problems so that Q_n may be computed through a recursion over $Q_{n-1}, \dots, Q_1$. Let $Q_0 = -\beta$ and $\mathcal{T}_t = \{0, \dots, t-1\}$. We get the following recursion $\forall\, t = 1, \dots, n$:

$$Q_t = \min_{\tau \in \mathcal{T}_t} \left\{ Q_\tau + \min_\theta \sum_{i=\tau+1}^{t} \gamma(y_i, \theta) + \beta \right\} = \min_{\tau \in \mathcal{T}_t} \min_\theta q_t^\tau(\theta) \, . \quad (2)$$

Reducing the size of $\mathcal{T}_t$ with the addition of a pruning step is instrumental in reducing the computational cost of the OP algorithm. Denote $q_t^t(\theta) := Q_t + \beta$ and consider $\tau_t(\theta) := \arg\min_{i \in \mathcal{T}_t \cup \{t\}} q_t^i(\theta)$. This function gives us the optimal last change point at time t for each value of parameter θ. We know that $S_t^s(\theta) := \{\theta : \tau_t(\theta) = s\}$ shrinks as t increases and we can therefore deduce that if no value of θ maps to s at time t then s will never be the optimal last change point for any time $T > t$ and it can be pruned.

2.2 The Duality Test

The previous pruning rule requires storing each $S_t^s(\theta)$ at each time t for each $s \in \mathcal{T}_t$. This incurs a high computational cost (see [4]), especially at higher dimensions, which is why we developed a test with no storing required. We simply compare s to a single other index r based on the fact that if $\exists\, r \neq s,\ r < t$ such that $\forall\, \theta,\ q_t^s(\theta) \geq \min\{q_t^r(\theta),\, Q_t + \beta\}$, then $S_t^s = \emptyset$. This allows us to define the minimization problem:

$$q_{r,s,t}^* := \min_\theta \{q_t^s(\theta) : q_t^s(\theta) - q_t^r(\theta) \leq 0\} \, . \quad (3)$$

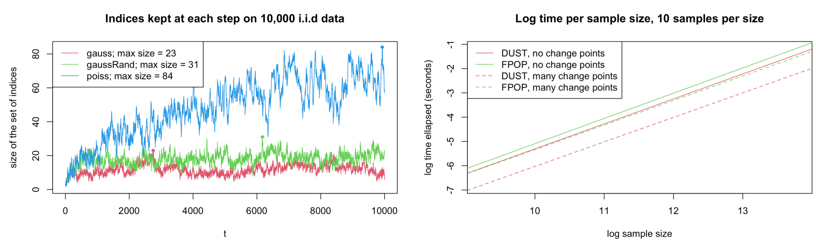
If the condition $q_{r,s,t}^* > Q_t + \beta$ is true, s may be pruned. However, the constraint is not convex, which may hinder the optimization process under certain models. To circumvent this issue, we consider the duality problem, with the lagrangian function $\mathcal{L}_{r,s,t}(\theta, \mu) := q_t^s(\theta) + \mu\left(q_t^s(\theta) - q_t^r(\theta)\right)$ and $\mu_{r,s,t}^{\max} := (t-s)/(s-r)$:

$$d_{r,s,t}^* := \max_{\mu \in \left[0, \mu_{r,s,t}^{\max}\right[} \min_\theta \mathcal{L}(\theta, \mu) \, . \quad (4)$$

By property of the duality problem, we have $d^*_{r,s,t} \leq q^*_{r,s,t}$. Therefore $d^*_{r,s,t} > Q_t + \beta \implies q^*_{r,s,t} > Q_t + \beta$ and we may prune s based on the value of $d^*_{r,s,t}$. This outlines the generic principle of the DUST method where upon computing Q_t at each time t, we loop through each $s \in \mathcal{T}_t$, and compute $d^*_{r,s,t}$ to perform the pruning test.

3 Simulation Results in 1D

Simulations show strong pruning capacity (Fig. 1a), as the size of $\mathcal{T}_t$ increases very slowly in the random Poisson version reaching $\#\mathcal{T}_n \approx n/100$, and stabilizing around $n/350$ in the Gaussian variants, with $\beta = 2\log(n)$. Figure 1b shows the lines defined by linear regressions of formula $\log(time) = a + b\log(n)$ for the DUST and FPOP algorithms, in Gaussian 1D data. All regressions are strongly significant and each R^2 value approaches 1. It is clear that the efficient pruning in DUST's Gaussian version allows the algorithm to reach lower run times than FPOP, especially on data with many change points.



(a) Size of $\mathcal{T}_t$ under variants of the DUST algorithm

(b) Time cost linear regression in log/log scale for DUST and FPOP

References

1. R. Killick, P. Fearnhead and I. A. Eckley: Optimal Detection of Changepoints With a Linear Computational Cost. Journal of the American Statistical Association (2012)https://doi.org/10.1080/01621459.2012.737745
2. Maidstone R, Hocking T, Rigaill G, Fearnhead P. On optimal multiple change point algorithms for large data. Stat Comput. (2017) https://doi.org/10.1007/s11222-016-9636-3
3. Runge V et al. gfpop: An R Package for Univariate Graph-Constrained Change-Point Detection. Journal of Statistical Software (2023)https://doi.org/10.18637/jss.v106.i06
4. Pishchagina L, Rigaill G, and Runge V. Geometric-Based Pruning Rules For Change Point Detection in Multiple Independent Time Series. Computo (2024) https://openreview.net/forum?id=YIes47zsCE