Adaptive Sequential Bayesian Change Point Detection

UNIVERSITY OF CAMBRIDGE

Computational and Biological Learning

Ryan Turner, Yunus Saatci, and Carl Edward Rasmussen

Abstract

- Treat change points as latent variables handled in a coherently Bayesian fashion
- Closed form and online inference algorithm
- Learn hyper-parameters efficiently from data
- Highly modular framework
- MATLAB code made publicly available at [1]

1 Introduction

- Many Bayesian change approaches are retrospective, while many applications demand online behavior
- Bayesian online change point detection (BOCPD) introduced by [2]
- BOCPD sensitive to hyper-parameters, but we learn them from data
- Define run length r_t as time since last change point at time t
- Goal is calculate $p(r_t|x_{1:t})$ from observations $x_{1:t}$.
- Two components are underlying predictive model (UPM), $p(x_t|x_{(t-\tau):t}, \theta_m)$, and change point hazard function, $H(r|\theta_h)$.

2 The BOCPD algorithm

Consider,

$$p(x_{t+1}|x_{1:t}) = \sum_{r_t} p(x_{t+1}|x_{1:t}, r_t) p(r_t|x_{1:t}) = \sum_{r_t} p(x_{t+1}|x_t^{(r)}) p(r_t|x_{1:t}) \,, \quad (1)$$

$$\begin{aligned}\gamma_t := p(r_t, x_{1:t}) &= \sum_{r_{t-1}} p(r_t, r_{t-1}, x_{1:t}) = \sum_{r_{t-1}} p(r_t, x_t|r_{t-1}, x_{1:t-1}) p(r_{t-1}, x_{1:t-1}) \\ &= \sum_{r_{t-1}} \underbrace{p(r_t|r_{t-1})}_{\text{hazard}} \underbrace{p(x_t|r_{t-1}, x_t^{(r)})}_{\text{likelihood (UPM)}} \underbrace{p(r_{t-1}, x_{1:t-1})}_{\gamma_{t-1}} \,. \quad (2)\end{aligned}$$

Defines forward message passing scheme.
Learn the parameters by maximizing the marginal likelihood

$$\log p(x_{1:T}|\theta) = \sum_{t=1}^{T} \log p(x_t|x_{1:t-1}, \theta) \,. \quad (3)$$

Using the derivatives of the UPM, $\frac{\partial}{\partial \theta_m} p(x_t|r_{t-1}, x_t^{(r)}, \theta_m)$, and those of the hazard function, $\frac{\partial}{\partial \theta_h} p(r_t|r_{t-1}, \theta_h)$, the derivatives of the one-step ahead predictors can be propagated forward.

3 Improving BOCPD

Pruning: The total run time of a naive implementation is $\mathcal{O}(T^2)$. In practice the run length distribution will be highly peaked. We can prune out run lengths with low probability. The modified algorithm runs in $\mathcal{O}(T)$, where the constant factor depends on pruning threshold.

Modularity: Any hazard function $H(t) \in [0, 1]$ can be plugged in. Any model that provides a posterior predictive can be used. We have implemented BOCPD modules for changing Gaussian process regression, Bayesian linear regression, and Kernel Density Estimation.

Caching: Predictions under given run lengths are made repeatedly. Predictive modules for $p(x_t|r_{t-1}, x_t^{(r)})$ can usually be speed up using intelligent caching.

4 Results

Well Log Data We used the logistic hazard, $H(t) = h\sigma(at + b)$, and used an IID Gaussian UPM, with the aim of detecting changes in mean and variance. After learning the parameters our method has a better predictive likelihood than [2].

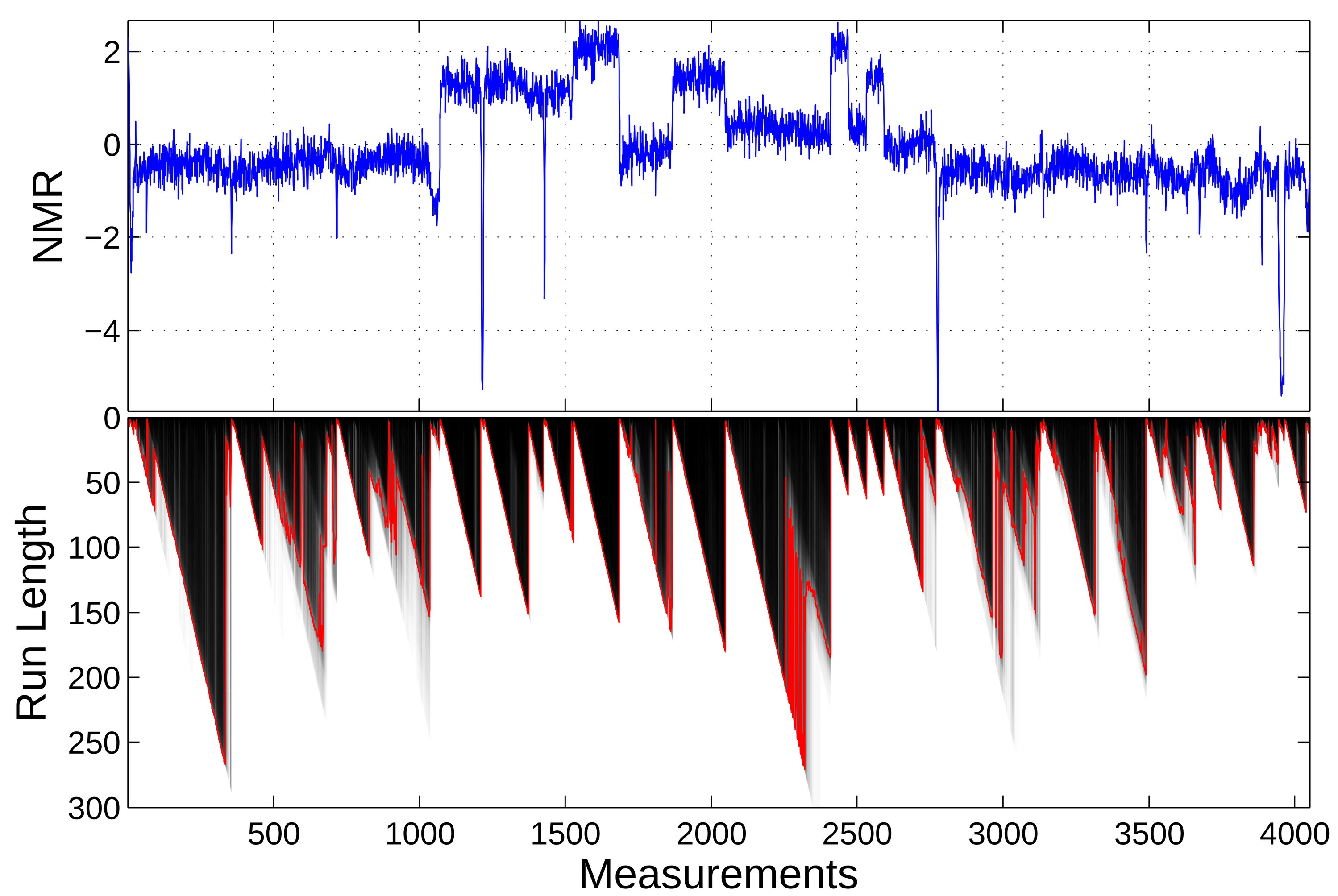

FIGURE 1: The BOCPD run length distribution on the well log data. The color represents the CDF of the run length distribution, while the red line represents the median of the distribution. Areas of a quick transition from black (CDF of zero) to white (CDF of one) indicate a sharply peaked run length distribution.

Industry Portfolio Data Tried the "30 industry portfolios" data set [3].
Change points found coincide with significant events: the climax of the Internet bubble, the burst of the Internet bubble, and the 2004 presidential election.

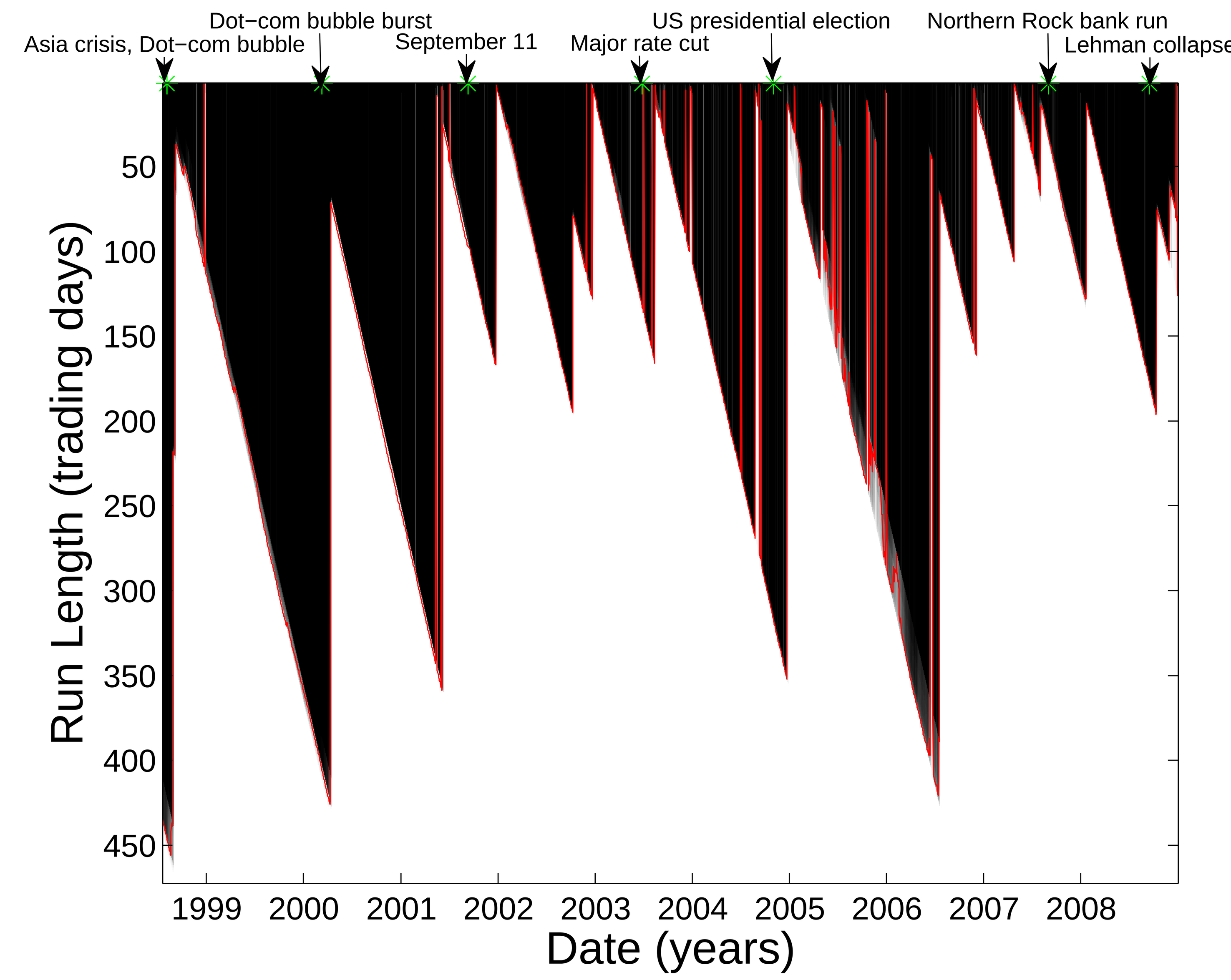

FIGURE 2: The BOCPD run length distribution between 1998 and 2008. Many events of market impact create change points. Some of the other change points correspond to minor rallies or rate changes but not to a historical event.

FIGURE 3: A summary of comparing the negative log predictive likelihoods (NLL) (nats/observation) on test data. We also include the 95% error bars on the NLL and the p-value that the joint model/learned hypers has a higher NLL using a one sided t-test. A reference method, the time independent model (TIM), treats the data as iid, normal for the well log and t for industry data. The TIM parameters are fit to the training set. **Well Log:** The learned hyper-parameter method was trained using the first 1000 points and tested on 3050 points. **Industry:** We test on the last 8455 points of the portfolio data, 3 July 1975–31 December 2008. The the methods were trained using the first 3000 points, 1 July 1963–2 July 1975. We compare running BOCPD independently on all 30 time series and using one joint BOCPD.

Well Log				Industry Portfolios			
Method	NLL	error bars	p-value	Method	NLL	error bars	p-value
TIM	1.53	0.0449	<1e-10	TIM	42.6	0.246	<1e-10
fixed hypers	0.313	0.0267	6e-4	indep.	39.64	0.217	0.271
learned hypers	**0.247**	0.0293	NA	joint	**39.54**	0.213	NA

References

[1] http://mlg.eng.cam.ac.uk/rdturner/bocpd/

[2] R. P. Adams and D. J. C. MacKay, "Bayesian online changepoint detection," Tecnical Report, University of Cambridge, Cambridge, UK, 2007. arXiv:0710.3742v1 [stat.ML].

[3] http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html.