CH5115
Parameter and State Estimation
Final Project

G Prashant (BS17B011)

February 16, 2021

Question 1

Part (a)

Critical Review - Bayesian model for online detection of changepoints in time-series data

Introduction and Background

Real-time analyses of time-series data have a plethora of applications in several engineering, financial and econometric domains. Typical goals of time-series data analyses include anomaly detection, forecasting and clustering. In scenarios where there is a sensor or an equivalent source that produces streaming data samples, which can potentially undergo changes in the properties of the data generating process (DGP) due to external stimuli, it is essential to perform an online detection of time-points when the change has occurred. To serve this purpose, online changepoint detection algorithms play an instrumental role in deriving useful real-time insights from streaming data. Online changepoint detection techniques aim to identify changes as and when they receive samples in real-time from the source. On the other hand, offline methods attempt to detect changes once all the samples are available[16]. In this context, changepoints refer to time-points when abrupt changes in parameters of the distribution of the DGP have occurred. Although changepoint detection algorithms can be classified into various categories, Bayesian frameworks have been extremely useful in modelling the incoming data in a generative approach, thereby enabling probabilistic detection of changepoints in data. In this article, the preprint publication on "Bayesian Online Changepoint Detection," authored by Ryan Prescott Adams and David J.C. MacKay[1], is critically reviewed.

Overview of the paper

In the preprint article on "Bayesian Online Changepoint Detection" (BOCD), the authors have formulated a recursive message-passing algorithm for online detection of changepoints in one-dimensional time-series data. The paper starts with a brief introduction and the motivation behind deploying online changepoint detection algorithms on data generating systems to characterise abrupt changes in the process in real-time. By pointing out a few example applications on DNA segmentation[3] and EEG analysis[10], the paper also highlights the importance of the advantages of online Bayesian techniques as opposed to other approaches such as online Frequentist and offline Bayesian changepoint detection methods. One of the key assumptions of the BOCD algorithm is that the sequence of observations pertaining to each non-overlapping partition is from the same distribution, while being independent and identically distributed (i.i.d.). In other words, all of the data plotted on a time-scale axis can be divided into a finite number of non-overlapping partitions, and each of them contains independently generated data samples that come from the same probability distribution whose parameters are unknown. The term run length r_t at time t refers to the time since the occurrence of the preceding changepoint. This paper hypothesises that by modelling the posterior distribution of r_t given all the data points till the instant t, which is denoted by $\boldsymbol{x_{1:t}}$, we can probabilistically determine the occurrence of a changepoint at the instant t. The fact that this posterior distribution $p(r_t|\boldsymbol{x_{1:t}})$ can be mathematically modified to be expressed in terms of the joint distribution of r_{t-1} and $\boldsymbol{x_{1:t-1}}$, brings in a recursive component to this algorithm. A wide class of probability distributions can be used to model the data and the prior, making the algorithm highly modular. Overall, this paper aims to provide a framework for detecting changepoints in real-time by computing the posterior probability of the run length conditioned on the observed data. The performance of the BOCD algorithm was evaluated on three popular benchmark time-series datasets – Well Log data, Dow Jones Returns data and Coal Mine Disaster data. By showing that the algorithm accurately predicts changepoints at multiple locations in these datasets with high probability, the authors of the BOCD paper describe the object-oriented, "pluggable" and modular nature of this type of Bayesian architecture.

Short description of BOCD algorithm

The assumption of the distribution of the data and the conjugate prior is made before implementing the algorithm. Let $p(x_t|\eta_t^{(r)})$ be the likelihood distribution with the parameter vector at time t and run length r_t denoted as $\eta_t^{(r)}$. The prior distribution $p(\eta_t^{(r)}|\nu_t^{(r)})$ is characterised by the hyperparameter vector $\nu_t^{(r)}$ at time t. The expression for the run length posterior is given by

$$p(r_t|\boldsymbol{x_{1:t}}) = \frac{p(r_t, \boldsymbol{x_{1:t}})}{p(\boldsymbol{x_{1:t}})} = \frac{\sum_{r_{t-1}} p(r_{t-1}, \boldsymbol{x_{1:t-1}}) p(x_t|r_{t-1}, \boldsymbol{x_{1:t-1}}) p(r_t|r_{t-1})}{\sum_{r_{t'}} p(r'_t, \boldsymbol{x_{1:t}})} \tag{1}$$

In the above expression, the term $p(r_t|r_{t-1})$ is called the changepoint prior probability. By using a memoryless Hazard function $H(r_t|r_{t-1})$ to model this prior probability, we can deduce the following

$$p(r_t|r_{t-1}) = H(r_t|r_{t-1}) = \begin{cases} \frac{1}{\lambda_{CP}} & r_t = 0 \\ 1 - \frac{1}{\lambda_{CP}} & r_t = r_{t-1} + 1 \\ 0 & \text{else} \end{cases} \tag{2}$$

Here, λ_{CP} is the timescale factor.

The first step to begin the algorithm is to initialize initial run length (which in most cases is initialized to 0), and the hyperparameters of the prior distribution $\nu_0^{(0)}$. After computing the predictive, growth and changepoint probabilities, we can arrive at the run length posterior probability $p(r_t|\boldsymbol{x_{1:t}})$ for all $r_t \in \{0, 1, \cdots, t\}$ recursively using the expression given in equation (1). Following this, the sufficient statistics, which are the hyperparameters $\nu_{t+1}^{(r_t+1)}$ of the conjugate prior are updated according to the recursive equations governed by the parameters of the probability distribution of the posterior. The process is repeated until all the data points are observed. Changepoints are detected at points that have high probability of $p(r_t \approx 0|\boldsymbol{x_{1:t}})$. In the process, the parameters in $\eta_t^{(r)}$ can be predicted at each instant of time.

Impactful contributions and potential applications

Several studies have pointed out the drawbacks and shortcomings of changepoint detection algorithms based on offline and Frequentist techniques. Offline techniques, sometimes referred to as retrospective signal segmentation[16], have applications in areas like genomic data analysis and climatology, where it is more meaningful to derive insights by processing the entire data at once. However, these algorithms cannot be considered effective when it comes to tasks such as EEG monitoring, quality control and anomaly detection. Meanwhile, studies that evaluated different changepoint detection techniques on real-world datasets have proved that Bayesian frameworks are better in terms of performance than the Frequentist counterparts[6]. In one of such analyses, this paper's BOCD algorithm was among the best performing methods on real-world univariate and multivariate time series datasets. In this regard, this paper presents a stepping stone to plenty of other research studies that proposed improvements to the BOCD algorithm. For example, *Turner et al.* suggested an improvised and robust modification to the BOCD algorithm that learns hyperparameters in a better way, resulting in improvement in performance[17]. Likewise, *Wilson et al.* proposed a hierarchical extension to the algorithm by enabling the hazard function H to be inferred from the data, rather than assigning it a constant value[19]. Online changepoint detection techniques are widely applied in speech processing[4], health monitoring[12], network traffic data analysis[11], bioinformatics[3], finance[7], econometrics and Internet of Things (IoT). One of the studies formulated a novel extension to the BOCD algorithm that can detect more meaningful changepoints in noisy biological and clinical time-series datasets[8].

Drawbacks and Challenges

The BOCD algorithm provides a simplistic Bayesian framework for detecting changepoints in real-time. Nevertheless, the inherent complexity of the problem due to various factors contributes to many drawbacks and challenges of this algorithm, most of which have not been mentioned by the authors in their paper. A few shortcomings of the paper and the algorithm are listed below.

- Assumption of independent observations: The underlying assumption of this algorithm is that the data points in a partition are i.i.d., implying no dependence of one data point with another. However, the algorithm might fail when the data generating process is nonstationary, such as autoregressive and moving average processes.
- Non-Conjugate prior: The paper does not mention the case when the prior distribution over the parameters of the likelihood function is not a conjugate prior, in which case the posterior probability will have a different class of distribution as that of the prior. Choosing a non-conjugate prior might not allow the hyperparameters to be recursively updated.
- Parametric form: BOCD algorithm is a parametric changepoint detection technique that assumes the distribution of the data. In scenarios when we do not know about the underlying process that generates data, wrong assumptions of the data distribution might lead to incorrect results. Further, results obtained through parametric approaches are highly sensitive to initial values of hyperparameters[2].
- Evaluation Metrics: The authors of the paper have evaluated the algorithm's performance on three different real-world datasets through visual inspection of intensity plots of run length posterior probabilities. However, they have not used evaluation metrics or statistical tests to quantitatively characterise the accuracy of the algorithm. Recent studies have come up with appropriate metrics to assess the performance of changepoint detection algorithms [18][2].
- Hyperparameter tuning: As mentioned before, the BOCD algorithm is sensitive to prior hyperparameter initializations. The paper does mention ways to tune the initial set of prior hyperparameters when they are unknown.
- Latency in changepoint detection: Due to practical reasons, it is impossible to detect changepoints at the exact same instance of the introduction of disturbance. Instead, there would be a small latency, or delay in detecting the changepoint.
- Underflow and overflow issues: The BOCD algorithm is prone to underflow and overflow issues, especially with high dimensional data. Although these issues are common in Bayesian statistics, there exist solutions that can resolve such issues[14].

Discussion and Conclusion

Changepoint detection algorithms can be either offline or online, Frequentist or Bayesian, and parametric or non-parametric. With time, algorithms in each of these categories are rapidly evolving in terms of improvised performance and robustness. For example, recent advancements in non-parametric techniques have resulted in the development of neural network based changepoint detection algorithms[5][15][9]. The Bayesian Online Changepoint Detection uses a Bayesian model with a message-passing algorithm for determining the posterior distribution of the run length conditioned on all the data points observed so far. This preprint paper, authored by Ryan Prescott Adams and David J.C. MacKay, is cited by 576 other research articles, some of which propose variants and extensions to the original algorithm [13]. In conclusion, choosing the most appropriate changepoint detection algorithm depends on the nature of the problem and the characteristics of the time-series data. Underlying assumptions of certain algorithms might hold for one type of problem, but not for others. Computational cost complexity and the ability of an algorithm to scale well with large and multidimensional datasets are other important factors that should be considered while choosing the best technique.

Part (b)

Bayesian Online Changepoint Detection (BOCD) [1] algorithm was implemented on the given NMR time series data. The hyperparameters ($\mu_0,\ k,\ \alpha,\ \beta$) of the recursive Bayesian model were initialised to values given in the question. Gaussian model was used to model the likelihood of the data, whose mean and variance were unknown at different intervals between two consecutive changepoints. Further, normal-gamma conjugate prior was used to model the joint distribution of the unknown mean and variance. The steps of the algorithm, along with other relevant details is as follows.

Algorithm

Step 1: Initialisation

The process is assumed to have a run length of 0, at 0^{th} instant (or, $r_0 = 0$). The initialisation of the hyperparameters is given below.

$$\mu_{0_0}^{(0)} = \mu_{0_{prior}} = 1.15,\ k_0^{(0)} = k_{prior} = 0.01,\ \alpha_0^{(0)} = \alpha_{prior} = 20,\ \beta_0^{(0)} = \beta_{prior} = 2$$

$$p(r_0 = 0) = 1$$

$$t = 1$$

Step 2: Data point observation

Incoming data x_t is observed.

Step 3: UPM prediction probability calculation

For all run length values $l \in \{0, 1, \cdots, t\}$, the Underlying Probabilistic Model (UPM) predictive probabilities conditioned on the hyperparameters of the normal-gamma distribution are calculated.

$$\pi_{t-1}^{(l)} = p(x_t|\mu_{0_{t-1}}^{(l)}, k_{t-1}^{(l)}, \alpha_{t-1}^{(l)}, \beta_{t-1}^{(l)}) \tag{3}$$

We can write the above probability in terms of the product of the likelihood distribution (Gaussian) and prior distribution (normal-gamma), and marginalizing over the parameters $\mu,\ \lambda$.

$$p(x_t|\mu_{0_{t-1}}^{(l)}, k_{t-1}^{(l)}, \alpha_{t-1}^{(l)}, \beta_{t-1}^{(l)}) = \int_\mu \int_\lambda \underbrace{p(x_t|\mu, \lambda\mu_{0_{t-1}}^{(l)}, k_{t-1}^{(l)}, \alpha_{t-1}^{(l)}, \beta_{t-1}^{(l)})}_{\mathcal{N}(\mu,\lambda)} \underbrace{p(\mu, \lambda, \mu_{0_{t-1}}^{(l)}, k_{t-1}^{(l)}, \alpha_{t-1}^{(l)}, \beta_{t-1}^{(l)})}_{p(\mu|\lambda;\alpha,\beta)p(\lambda;\alpha,\beta)} d\lambda d\mu \tag{4}$$

Upon integrating, we can notice that the UPM follows a scaled variant of student's t-distribution, as follows

$$p(x_t|\mu_{0_{t-1}}^{(l)}, k_{t-1}^{(l)}, \alpha_{t-1}^{(l)}, \beta_{t-1}^{(l)}) = t_{2\alpha_{t-1}^{(l)}}\left(\mu_{0_{t-1}}^{(l)}|\frac{\beta_{t-1}^{(l)}(k_{t-1}^{(l)}+1)}{\alpha_{t-1}^{(l)}\beta_{t-1}^{(l)}}\right) \tag{5}$$

Step 4: Recursively calculate growth probabilities

The probability that the run length $r_t = l + 1$ is given by,

$$p(r_t = l+1|\boldsymbol{x_{1:t}}) = p(r_{t-1} = l|\boldsymbol{x_{1:t-1}}) * \pi_{t-1}^{(l)} * (1 - H(r_{t-1})) \tag{6}$$

where the changepoint prior probability $H(r_{t-1}) = \frac{1}{\lambda_{CP}}$ at all times, thereby making it a constant and memoryless Hazard function.

Step 5: Calculate changepoint probability

The probability that $r_t = 0$, i.e., a changepoint has ocurred at instant t is given by

$$p(r_t = 0|\boldsymbol{x_{1:t}}) = \sum_{r_{t-1}} p(r_{t-1}|\boldsymbol{x_{1:t-1}}) * \pi_{t-1}^{(l)} * H(r_{t-1}) \tag{7}$$

Step 6: Normalising posterior probabilities

To make the posterior probabilities for all possible run lengths sum up to one, each probability is normalized by dividing by the marginalized probability summed over all run length values, i.e., $\sum_{r_t} p(r_t|\boldsymbol{x_{1:t}})$. This term is called the evidence.

Step 7: Update hyperparameters

The update equations for each of the hyperparameters is as follows

$$\mu_{0_t}^{(0)} = \mu_{0_{prior}} \tag{8}$$

$$k_t^{(0)} = k_{prior} \tag{9}$$

$$\alpha_t^{(0)} = \alpha_{prior} \tag{10}$$

$$\beta_t^{(0)} = \beta_{prior} \tag{11}$$

When the run length is zero, the hyperparameters are not updated, otherwise,

$$\mu_{0_t}^{(l+1)} = \mu_{0_{t-1}}^{(l)}(1 - \frac{1}{k_{t-1}^{(0)} + t}) + \frac{x_t}{k_{t-1}^{(0)} + t} \tag{12}$$

$$k_t^{(l+1)} = k_{t-1}^{(l)} + 1 \tag{13}$$

$$\alpha_t^{(l+1)} = \alpha_{t-1}^{(l)} + 1/2 \tag{14}$$

$$\beta_t^{(l+1)} = \beta_{t-1}^{(l)} + \frac{k_{t-1}^{(l)}(x - \mu_{0_{t-1}}^{(l)})^2}{2(k_{t-1}^{(l)} + 1)} \tag{15}$$

Step 8: Back to Step 2

The predicted mean at time t is estimated. We then assign $t = t + 1$ and return to step 2 or terminate when all data points are observed.

Results

After implementing the BOCD algorithm, plots of the true data with predicted mean, run length corresponding to maximum posterior probability, and a scaled intensity plot of the posterior probabilities of run lengths against time were generated, as shown in Figure 1. Visually inspecting Figure 1, it can be noticed that the algorithm has detected multiple changepoints in the data, yielding accurate results with high probability. The changepoints were detected occurred at time points $t = 85,\ 220,\ 452,\ 812,\ 873,\ 941,$ and 999. The maximum run length was approximately 360, from time 452 to 812.

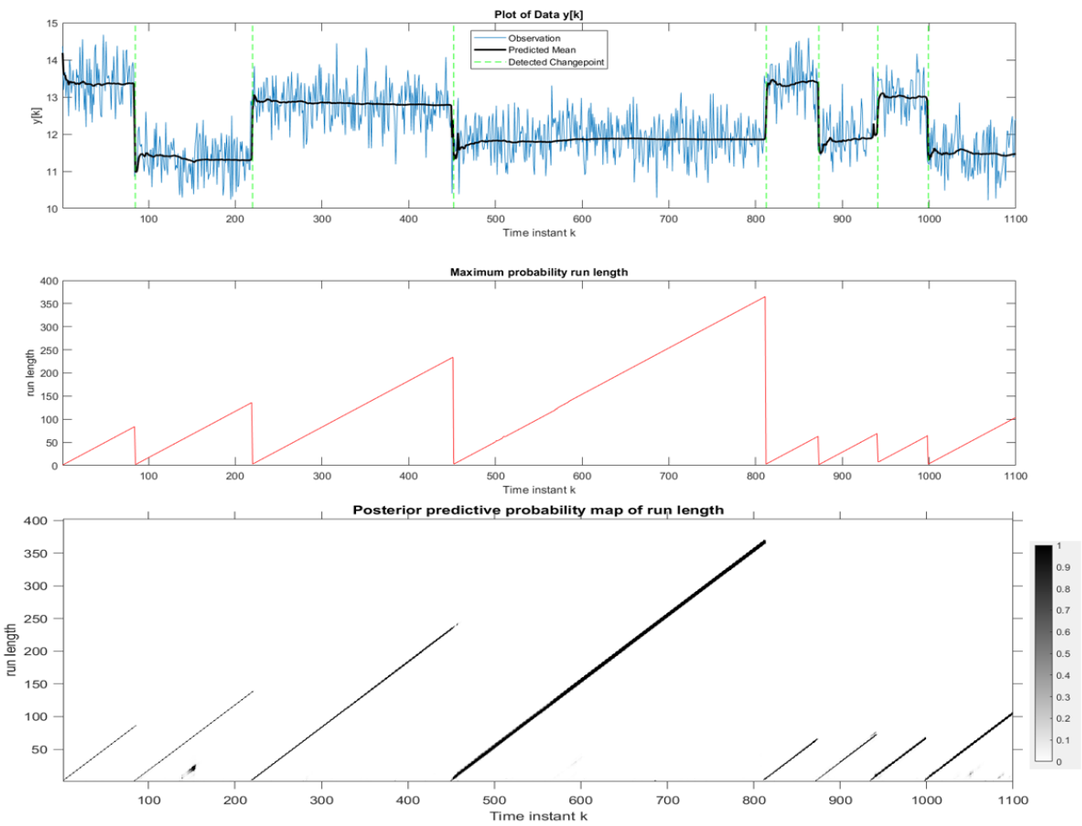

Figure 1: BOCD Algorithm on NMR Data. First plot - Plot of data points, predicted mean and detected change-points; Second plot - Plot of run length corresponding to maximum posterior probability; Third plot - Scaled intensity plot of run length posterior probability

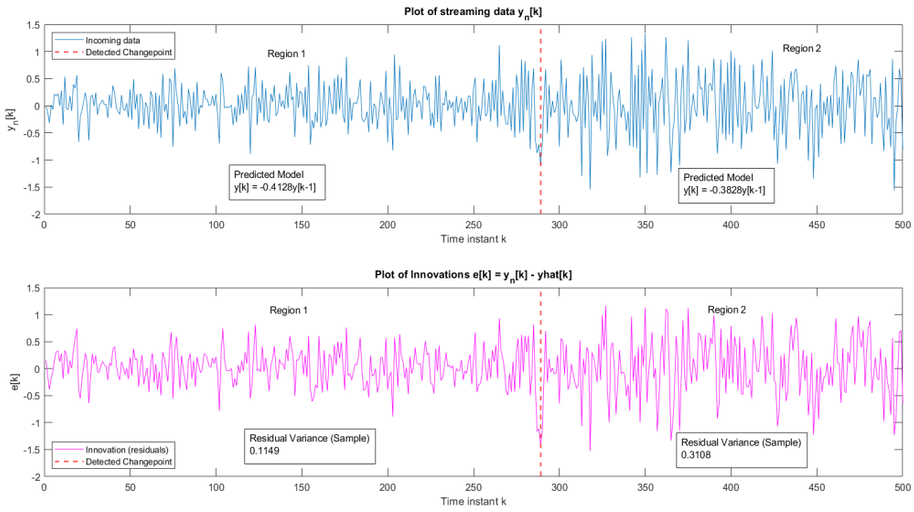

Figure 2: First plot - RLS on streaming data; Second plot - Residual plot

Question 2

Part (a)

In order to determine the order of the AR model that best suits the given historical sample data, in file historic.mat, whiteness test of residuals was performed, followed by testing of the significance of the parameters. It was determined that AR(1) model fits the data best when compared to higher-order autoregressive models. The coefficient of the AR(1) process was determined using Linear Least Squares, and the appropriate initial model is given below

$$y_h[k] = -0.60894 y_h[k-1] \tag{16}$$

Here, y_h represents the historic data. Using this initial model consisting of the parameter estimate and covariance of regressors, a Recursive Least Squares (RLS) algorithm was implemented, starting from the first data point in the file new.mat. Simultaneously, BOCD algorithm with model and prior distributions similar to the previous question was implemented after 200 time steps, on the innovation samples. In this process, the first changepoint was detected at time point 289. The parameter of the AR(1) model was estimated until the 289^{th} time, which is given below

$$y_n^{(1)}[k] = -0.4128 y_n^{(1)}[k-1], \quad k \leq 289 \tag{17}$$

where $y_n^{(1)}$ represent the data points before the changepoint - region 1.

Part (b)

After the changepoint was detected at the 289^{th} instant, a new RLS object was instantiated to recursively determine the AR(1) coefficient using the data points in region 2. The following model was obtained for the 2^{nd} region

$$y_n^{(2)}[k] = -0.3828 y_n^{(2)}[k-1], \quad k > 289 \tag{18}$$

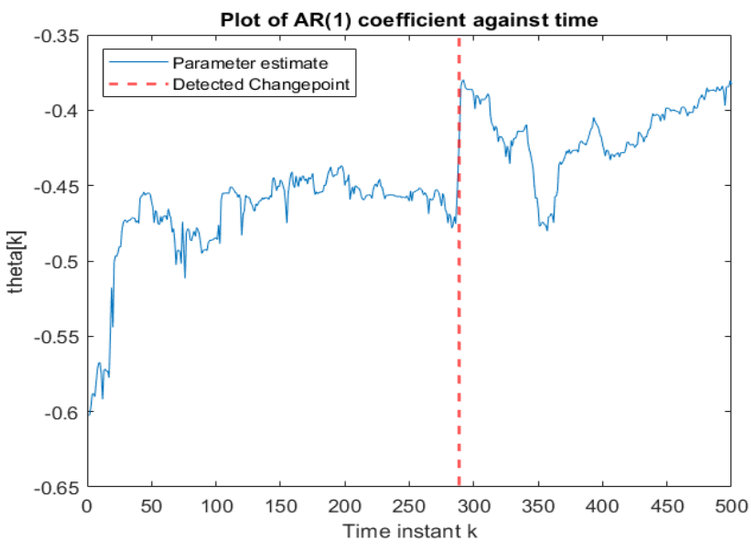

Figure 3: Parameter Estimate of AR(1) process against time

The innovation term (residual) was calculated at each step, as given below

$$e[k] = y_n[k] - \hat{y_n}[k] \tag{19}$$

First subplot of Figure 2 shows the plot of the streaming data, marked changepoint time and the model equations for each region. Second subplot of Figure 2 shows the residuals (or innovations) plotted against time. It can be noted that before the changepoint, the sample variance of innovations in Region 1 was 0.1149. The introduction of disturbance caused the variance of the white noise in Region 2 to increase to approximately 3 times (0.3108) as that of Region 1. Figure 3 shows the plot of the estimate of the AR(1) coefficient after every RLS update. After the changepoint, the estimate increases sharply to a higher value. The terminal value of the coefficient in Region 2 is greater than that of the coefficient of Region 1.

References

[1] Ryan Prescott Adams and David JC MacKay. Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*, 2007.

[2] Samaneh Aminikhanghahi and Diane J. Cook. A survey of methods for time series change point detection. *Knowledge and information systems*, 51(2):339–367, May 2017. 28603327[pmid].

[3] Jerome V Braun, RK Braun, and Hans-Georg Müller. Multiple changepoint fitting via quasilikelihood, with application to dna sequence segmentation. *Biometrika*, 87(2):301–314, 2000.

[4] Frédéric Desobry, Manuel Davy, and Christian Doncarli. An online kernel change detection algorithm. *IEEE Transactions on Signal Processing*, 53(8):2961–2974, 2005.

[5] Zahra Ebrahimzadeh, Min Zheng, Selcuk Karakas, and Samantha Kleinberg. Deep learning for multi-scale changepoint detection in multivariate time series. *arXiv preprint arXiv:1905.06913*, 2019.

[6] IA Eckley, P Fearnhead, and R Killick. Probabilistic methods for time series analysis. *Bayesian Time Series Models. Cambridge University Press, Cambridge, UK*, 2010.

[7] VASILE GEORGESCU. Online change-point detection in financial time series: challenges and experimental evidence with frequentist and bayesian setups. In *Methods For Decision Making In An Uncertain Environment*, pages 131–145. World Scientific, 2012.

[8] Nathan Gold, Martin G Frasch, Christophe L Herry, Bryan S Richardson, and Xiaogang Wang. A doubly stochastic change point detection algorithm for noisy biological signals. *Frontiers in physiology*, 8:1112, 2018.

[9] Mikhail Hushchyn, Kenenbek Arzymatov, and Denis Derkach. Online neural networks for change-point detection. *arXiv preprint arXiv:2010.01388*, 2020.

[10] A Ya Kaplan and Sergei L Shishkin. Application of the change-point analysis to the investigation of the brain's electrical activity. In *Non-parametric statistical diagnosis*, pages 333–388. Springer, 2000.

[11] Céline Lévy-Leduc, François Roueff, et al. Detection and localization of change-points in high-dimensional network traffic data. *Annals of Applied Statistics*, 3(2):637–662, 2009.

[12] Siqi Liu, Adam Wright, and Milos Hauskrecht. Change-point detection method for clinical decision support system rule monitoring. In *Conference on Artificial Intelligence in Medicine in Europe*, pages 126–135. Springer, 2017.

[13] Matthew R Nassar, Robert C Wilson, Benjamin Heasly, and Joshua I Gold. An approximately bayesian delta-rule model explains the dynamics of belief updating in a changing environment. *Journal of Neuroscience*, 30(37):12366–12378, 2010.

[14] Andisheh Partovi and Gholamreza Haffari-Ingrid Zukerman. Bayesian changepoint detection in textual data streams.

[15] Michalis K Titsias, Jakub Sygnowski, and Yutian Chen. Sequential changepoint detection in neural networks with checkpoints. *arXiv preprint arXiv:2010.03053*, 2020.

[16] Charles Truong, Laurent Oudre, and Nicolas Vayatis. Selective review of offline change point detection methods. *Signal Processing*, 167:107299, 2020.

[17] Ryan Turner, Yunus Saatci, and Carl Edward Rasmussen. Adaptive sequential bayesian change point detection. In *Temporal Segmentation Workshop at NIPS*, pages 1–4. Citeseer, 2009.

[18] Gerrit JJ van den Burg and Christopher KI Williams. An evaluation of change point detection algorithms. *arXiv preprint arXiv:2003.06222*, 2020.

[19] Robert C Wilson, Matthew R Nassar, and Joshua I Gold. Bayesian online learning of the hazard rate in change-point problems. *Neural computation*, 22(9):2452–2476, 2010.