Online Non-Parametric Changepoint Detection With Application to Monitoring Operational Performance Of Network Devices

Edward Austin[1], Gaetano Romano[2], Idris Eckley[2,*], Paul Fearnhead[2]

Abstract

Motivated by a telecommunications application where there are few computational constraints, a novel nonparametric algorithm, NUNC, is introduced to perform an online detection for changes in the distribution of data. Two variants are considered: the first, NUNC Local, detects changes within a sliding window. Conversely, NUNC Global, compares the current window of data to all of the historic information seen so far and makes use of an efficient update step so that this historic information does not need to be stored. To explore the properties of both algorithms, both real and simulated datasets are analysed. Furthermore, a theoretical result for the choice of test threshold to control the false alarm rate is presented, a result that could be applied in other binary segmentation change detection settings.

Keywords: online changepoint detection, non-parametric statistics, network devices, NUNC

1. Introduction

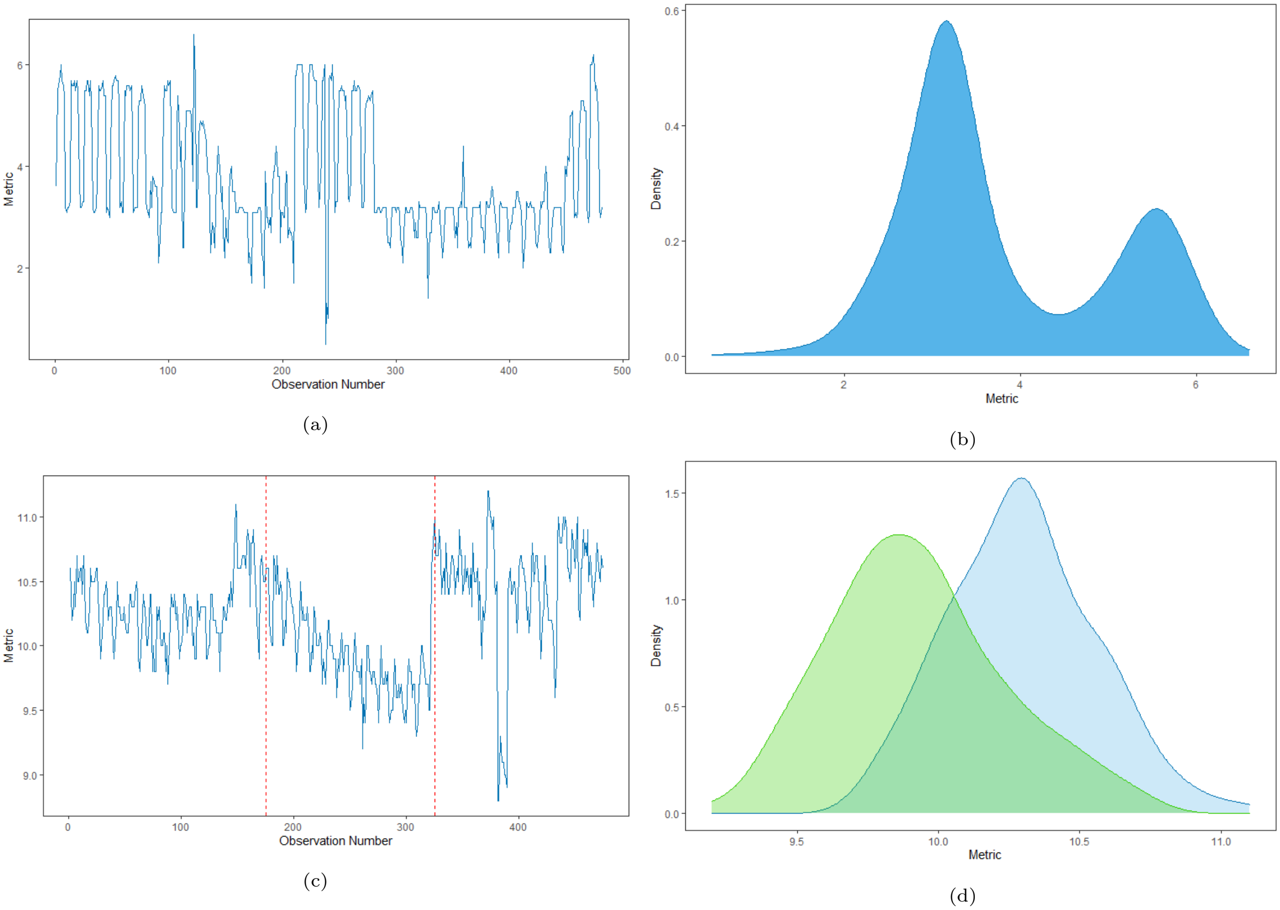
The challenge of sequential nonparametric changepoint detection has seen significant development in recent years. See, for example, Tartakovsky et al. (2014) for an excellent introduction to the area. Contributions include the work of Gordon & Pollak (1994), Ross & Adams (2012), and Padilla et al. (2019) who introduce novel approaches to detect changes in an unknown distribution. Others, including Chakraborti & van de Wiel (2008); Hawkins & Deng (2010); Murakami & Matsuki (2010); Ross et al. (2011); Mukherjee & Chakraborti (2012); Liu et al. (2013); Wang et al. (2017) and Coelho et al. (2017) seek to address a different non-parametric challenge: the sequential detection of changes in the mean, scale, or the location of the data. Such methods have also found application in a range of fields including monitoring financial systems (Pepelyshev & Polunchenko, 2017), monitoring viral intrusion in computer networks (Tartakovsky et al., 2005), detecting changes in social networks (Chen, 2019), genome sequencing (Siegmund, 2013), and radiological data (Padilla et al., 2019).

Our work is motivated by a different challenge, increasingly encountered within many contemporary digital settings, such as those found in the telecommunications sector. In such environments it is important to perform device-side analyses on units with limited computational power and data storage capability. Existing methods, such as those mentioned above, are unsuitable for use in such cases as they require the entire data stream to be stored and analysed *a posteriori*, whereas our memory-constrained setting makes this impossible. As a consequence, an online approach that uses a lighter data footprint is required.

One example of such data, encountered by an industrial collaborator, can be seen in Figure 1. Here we display two sample data sets of a key operational metric that are routinely monitored to identify problems with networking devices. Figure 1(a) displays data from a healthy device, whereas the data in Figure 1(c) contains an event that

[*] Corresponding author
Email address: `i.eckley@lancaster.ac.uk` (Idris Eckley)
[1] STOR-i Centre for Doctoral Training, Lancaster University, Lancaster LA1 4YF, U.K.
[2] Department of Mathematics and Statistics, Lancaster University, Lancaster LA1 4YF, U.K.

triggers a user intervention. Note in particular how the structure of the data changes in the region when the event occurs. This can perhaps be more clearly seen in Figures 1(b,d). The ideal, therefore, is to be able to (i) identify the start of changing structure in advance of the user being required to start an intervention – we call the correct detection of such an event an 'anticipation'; (ii) using an approach that does not necessarily require the same underlying distribution pre- and post-change and (iii) can still permit (more subtle) non-anomalous changes in structure that occur over time due to typical operational issues (e.g. electrical interference, line optimisation, etc).



(a)

(b)

(c)

(d)

Figure 1: Example of telecoms operational data: (a) a series without an event, and (c) a series with an event taking place between the two red lines. The corresponding kernel density estimates for the time series in figures (a) and (c) are presented in (b) and (d) respectively. Note that in (d) the green represents the kernel density estimate of the series when the event is taking place.

Many existing methods, such as those mentioned above, are unsuitable for use in this setting as they typically require the entire data stream to be stored. Unfortunately in our problem setting, such memory constraints are no longer possible. To overcome this one might, for example, consider adopting an online non-parametric changepoint detection approach using a sliding window, such as in the MOSUM test (Chu et al., 1995; Eichinger & Kirch, 2018; Kirch et al., 2018; Meier et al., 2021). Alternatively a control chart based approach, e.g. Ross & Adams (2012), might be considered. Unfortunately, as described above, our telecoms operational metric can exhibit non-anomalous shifts in mean and variance. Such structure, while operationally acceptable, may cause a control chart to return excessive false alarms due to the cumulative nature of the test statistic. Further, existing nonparametric sequential changepoint detection methods prove unsuitable as they do not expect the null distribution to change over time. While there are some methods that attempt to model and account for non-anomalous drift (Gama et al., 2014), these can be challenging to tune when the form of drift is unknown. We thus take a different approach and use a sliding window to deal with drift. To this end, we introduce a new windowed, non-parametric procedure to detect sequential changes in an online setting. Taken from the Latin, *nunc* ('now'), our approach provides a **N**on-Parametric **UN**bounded **C**hangepoint (NUNC) detection.

We propose two variants of NUNC: NUNC Local, and NUNC Global. The first of these algorithms, NUNC Local, performs the detection in a sliding window, considering only the points inside this window. This allows for the implementation to work in an online setting. The second, NUNC Global, uses an efficient updating step to compare the distribution of the historic data seen with the distribution of the data inside the sliding window; if these differ significantly, a change is identified.

The rest of this paper is organised as follows: In Section 2 we outline the methodology behind our new sequential tests. In particular, we detail the existing non-parametric changepoint methods our work is based upon, and in Section 2.1 we provide details of our two new window-based changepoint detection tests. The remainder of the section then explores the properties of the test, including the choice of quantiles and threshold. In particular, a theoretical result is presented concerning the selection of the threshold in order to control the false alarm probability, and this result can also be utilised in an offline changepoint setting. The article concludes by exploring the performance of NUNC Local and NUNC Global using both simulated scenarios (Section 3) and data arising from the previously described telecommunications setting (Section 4).

2. Background and Methodology

Our approach builds on the recent work of Zou et al. (2014) and Haynes et al. (2017), utilising a non-parametric likelihood ratio test as the basis for the proposed sequential Non-Parametric test. In so doing, the method permits a range of data distributions to be modelled, without the need for restrictive parametric assumptions. Below we introduce both NUNC approaches, and provide a discussion of their various features including computational performance and the choice of quantiles. However, prior to doing so, we review the pertinent literature on non-parametric changepoint methods.

We begin by outlining some notation. Assume that we observe a data stream of real valued independent observations $x_1, x_2, \ldots, x_t$, and that the data stream can contain a changepoint, at some (unknown) time point τ. Further, assume that x_1 is the start of this data stream, x_t is the most recently observed point, and that for $j > i$, $x_{i:j} = x_i \ldots, x_j$ denotes a segment of the data. If a change is present in the data at τ_1, then we refer to $x_1, \ldots, x_{\tau_1}$ as the pre-change segment, drawn from a distribution $F_1(\cdot)$. Similarly, $x_{\tau_1+1}, \ldots x_{\tau_2}$ are considered to be drawn from post-change distribution, $F_2(\cdot)$, with $F_1 \neq F_2$.

Following Zou et al. (2014), let $F_{i:t}(q)$ denote the (unknown) cumulative distribution function (CDF) for the segment $x_i, \ldots, x_t$, and $\hat{F}_{1:t}(q)$ as its associated empirical CDF. I.e.

$$\hat{F}_{1:t}(q) = \frac{1}{t} \sum_{j=1}^{t} \left\{ \mathbb{I}(x_j < q) + 0.5 \times \mathbb{I}(x_j = q) \right\}. \tag{1}$$

Under the assumption that the data are independent, then the empirical CDF will follow a Binomial distribution. That is,

$$t\hat{F}_{1:t}(q) \sim \text{Binom}(t, F_{1:t}(q)). \tag{2}$$

Using the Binomial distribution, we write the log-likelihood of the segment $x_{\tau_1+1}, \ldots, x_{\tau_2}$ as

$$\mathcal{L}(x_{\tau_1+1:\tau_2}; q) = (\tau_2 - \tau_1) \left[\hat{F}_{\tau_1+1:\tau_2}(q) \log(\hat{F}_{\tau_1+1:\tau_2}(q)) - (1 - \hat{F}_{\tau_1+1:\tau_2}(q)) \log(1 - \hat{F}_{\tau_1+1:\tau_2}(q)) \right]. \tag{3}$$

Consequently, equation (3) can be used to form a likelihood ratio test statistic for the detection of a change at a single quantile of the distribution as follows:

$$\max_{1 \leq \tau \leq t} 2 \left[\mathcal{L}(x_{1:\tau}; q) + \mathcal{L}(x_{\tau+1:t}; q) - \mathcal{L}(x_{1:t}; q) \right].$$

Following Zou et al. (2014) and Haynes et al. (2017), this test statistic can be averaged over multiple quantiles, $q_1, \ldots, q_K$, in order to search for a change in distribution. The statistic for such a test can be formulated as follows:

$$C_K(x_{1:t}) = \max_{1 \leq \tau \leq t} \frac{1}{K} \sum_{k=1}^{K} 2 \left[\mathcal{L}(x_{1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:t}; q_k) - \mathcal{L}(x_{1:t}; q_k) \right]. \tag{4}$$

Here K is the fixed number of quantiles to be averaged over. Haynes et al. (2017) propose that a value of K is chosen that is proportionate to $\log(t)$. The choice of quantiles at which the empirical CDF can be evaluated will be discussed later in Section 2.2.

Using this test, a changepoint is declared when $C_K(x_{1:t}) - \beta \geq 0$. Thus the stopping time for our test becomes

$$\max_{1 \leq \tau \leq t} \sum_{k=1}^{K} 2 \left[\mathcal{L}(x_{1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:t}; q_k) - \mathcal{L}(x_{1:t}; q_k) \right] \geq K\beta, \tag{5}$$

where β is the threshold for the test.

Having outlined how the existing (offline) non-parametric tests work, and how the cost function from this work can be used to devise a stopping rule for a sequential changepoint detection test, we are now in a position to introduce our two variant nonparametric approaches.

2.1. Two Sequential Changepoint Detection Algorithms

We now introduce two different, yet related, approaches that can be adopted within this nonparametric framework: NUNC Local and NUNC Global. Common to both is the use of a sliding window, and the test statistic given in equation (4). Where the two approaches differ, however, is the manner in which the data observed outside the window are handled. In NUNC Local, a simplistic perspective is adopted, taking the data contained within the sliding window into account – i.e., previously seen points that fall outside this window are forgotten. The advantage of this approach is that the sequential test is immune to false alarms that might be caused, for example, by a natural drift in the underlying distribution of the data. The drawback, however, is that the empirical CDF must be estimated only from the data in the window and so any historic information is lost.

NUNC Global seeks to overcome the short-comings of NUNC Local. Specifically, NUNC Global stores the empirical CDF that has been estimated using all data observed so far, and tests whether the data from such empirical distribution differ from the data observed in the current window. In Section 4 we will seek to contrast the differences between these to variants. However, prior to this, we describe both search methods more carefully, whilst also describing various properties and recommendations.

2.1.1. NUNC Local

Our first method takes a sliding window of size W and performs the test on the data within this sliding window. In the sliding window of points we have that

$$\mathcal{Q}_t^{local} = \max_{t-W+1 \leq \tau \leq t} \sum_{k=1}^{K} 2 \left[\mathcal{L}(x_{t-W+1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:t}; q_k) - \mathcal{L}(x_{t-W+1:t}; q_k) \right], \tag{6}$$

where K is the number of quantiles. Letting β be the the test threshold, when $\mathcal{Q}_t^{local} \geq K\beta$ then the algorithm stops at time t and declares that a change has occurred at time τ. A description of NUNC Local pseudocode can be found in Algorithm 1.

The choice of the parameters for NUNC Local, including the window size, quantiles, and threshold; will be discussed in Section 2.2 and in simulations in Section 3. We remark here, however, that the choice of K and the size of the window is related to the computational cost of NUNC Local. In particular, the naive cost of computing NUNC Local directly is $\mathcal{O}(KW^2)$. However, by using an Ordered Search Tree (Cormen et al., 2001) to calculate the empirical CDF this can be reduced to $\mathcal{O}(KW \log W)$.

```
Algorithm 1: NUNC Local Algorithm
  Data: $\{x_{t-W+1}, ..., x_{t-1}, x_t\}$, the last $W$ realizations from a data generating process $X$.
  Input: $\beta > 0$, $K < W$, $q_1, \dots, q_k$ quantiles
  $\mathcal{Q} \leftarrow \max_{t-W+1 \leq \tau \leq t} \left[\sum_{k=1}^{K} 2\left(\mathcal{L}(x_{(t-W):\tau}; q_k) + \mathcal{L}(x_{\tau+1:t}; q_k) - \mathcal{L}(x_{(t-W):t}; q_k)\right)\right]$;
  $\tau^* \leftarrow \arg\max_{t-W+1 \leq \tau \leq t} \left[\sum_{k=1}^{K} 2\left(\mathcal{L}(x_{(t-W):\tau}; q_k) + \mathcal{L}(x_{\tau+1:t}; q_k) - \mathcal{L}(x_{(t-W):t}; q_k)\right)\right]$;
  if $\mathcal{Q} \geq K\beta$ then
      Return $\tau^*$ as a changepoint
  end
```

In order to reduce the computational requirements of NUNC Local, which is quadratic in window size, it is possible to instead perform the search on a subset of the points in the sliding window. In this setting, we obtain the stopping condition:

$$\max_{\tau \in B_J} \sum_{k=1}^{K} 2\left[\mathcal{L}(x_{t-W+1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:t}; q_k) - \mathcal{L}(x_{t-W+1:t}; q_k)\right] \geq K\beta,$$

where $B_J \subset \{t - W + 1, \dots, t\}$. This corresponds to changing the maximisation in Algorithm 1 to taking place over the set B_J rather than the entire window. Using a subset of size $J << W$, the computational cost of NUNC Approximate is reduced to $\mathcal{O}(JKW)$. To find a suitable subset of values to search inside the sliding window, we first note that intuitively it only makes sense to search for a change in the right hand half of the window. This is because the data in the left of the window has already been scanned for a change several times. Moreover, we can also establish the following: that there exists a point on the right hand side of the window such that a changepoint cannot be detected to the right of this point.

Proposition 1. *For any quantile q the test statistic is bounded such that*

$$\mathcal{L}(x_{t-W+1:\tau}; q) + \mathcal{L}(x_{\tau+1:t}; q) - \mathcal{L}(x_{t-W+1:t}; q) \leq -\frac{\tau}{W} \log \frac{\tau}{W} - (W - \tau) \log\left(\frac{W - \tau}{W}\right). \tag{7}$$

Furthermore, for fixed W, this equation is decreasing as τ increases, and so if τ^* is the point such that

$$-\frac{\tau^*}{W} \log \frac{\tau^*}{W} - (W - \tau^*) \log\left(\frac{W - \tau^*}{W}\right) \leq \frac{\beta}{2}$$

then for $\tau > \tau^*$ detection of a change is impossible.

Proof. See Appendix. □

As a consequence of Proposition 1, only a portion of the right hand side of the window needs to checked, and the value of this cutoff can be found, with the value for τ^* being calculated numerically. Further computational efficiencies can be realised for NUNC Local if it is only performed on a spaced out grid of points. This is due to the value of the test statistic being correlated at nearby points. Consequently, if segmenting the data at t does not return a change, then it is unlikely that a change will be detected at $t + 1$. As a result, we propose to use an equally spaced grid of J points starting from the centre of the window, after it has been trimmed using the value of τ^*. However, one drawback of using the grid method is that there will be a higher detection delay for smaller values of J. This is due to it taking longer for the change to reach a point that we are checking. As such, we conclude that there is a trade-off between computational efficiency and detection delay when using the approximated algorithm.

2.1.2. NUNC Global

NUNC Global differs from the NUNC Local. Specifically it tests whether or not the data in the window comes from a different distribution to all the data seen so far. To store the information in a memory efficient manner, we again fix K quantiles and update the long-run empirical CDF, denoted by $z_W^{(t)}(\cdot)$, each time a point leaves the sliding window. The recursive equations for this update step are as follows:

$$\begin{aligned} z_W^{(t)}(q) &= \hat{F}_{1:(t-W)}(q), \\ z_W^{(t+1)}(q) &= \frac{1}{t-W+1}\left[(t-W)z_W^{(t)}(q) + \hat{F}_{(t-W+1):(t-W+1)}(q)\right], \quad t \geq W. \end{aligned} \tag{8}$$

I.e. the long-run empirical CDF is updated to take into account the point that will leave the sliding window at the next iteration. The Global algorithm then compares the distribution for the long-run empirical CDF to the distribution of the data in the sliding window, denoted by $\hat{F}_{t-W+1:m}(\cdot)$.

To implement this approach, we need to obtain a CDF estimate of the full data. This is given by a weighted mixture of the long-run empirical CDF and the current segment empirical CDF estimate. Assuming we are at time t, and have a sliding window of size, W, we write this as

$$\hat{F}_{\text{full}}(q) = \hat{F}_{1:t}(q) = \frac{t-W}{t} z_W^{(t)}(q) + \frac{W}{t}\hat{F}_{t-W+1:t}(q).$$

With these distributions in place, we can obtain the equivalent likelihoods, given respectively by

$$\begin{aligned} \mathcal{L}(x_{1:t-W};t) &= (t-W)\left[z_W^{(t)}(q)\log(z_W^{(t)}(q)) - (1-z_W^{(t)}(q))\log(1-z_W^{(t)}(q))\right] \\ \mathcal{L}(x_{t-W+1:t};t) &= W\left[\hat{F}_{t-W+1:t}(q)\log(\hat{F}_{t-W+1:t}(q)) - (1-F_{t-W+1:t}(q))\log(1-\hat{F}_{t-W+1:t}(q))\right] \\ \mathcal{L}(x_{1:t};t) &= t\left[\hat{F}_{\text{full}}(q)\log(\hat{F}_{\text{full}}(q)) - (1-\hat{F}_{\text{full}}(q))\log(1-\hat{F}_{\text{full}}(q))\right]. \end{aligned} \tag{9}$$

The test statistic is then given by:

$$\mathcal{Q}_t^{global} = \sum_{k=1}^{K} 2\left[\mathcal{L}(x_{1:t-W};q_k) + \mathcal{L}(x_{t-W+1:t};q_k) - \mathcal{L}(x_{1:t};q_k)\right]. \tag{10}$$

When $\mathcal{Q}_t^{global} \geq K\beta$ we stop and declare a change at time t. Pseudocode outlining the Global algorithm is provided in Algorithm 2. We note that the computational cost of NUNC Global is $\mathcal{O}(KW)$. As with NUNC Local, this can be reduced to $\mathcal{O}(K\log W)$ by implementing an Ordered Search Tree for the empirical CDF (Cormen et al., 2001).

The advantage of this approach, over NUNC Local, is that only K pieces of information are required to store information about the estimate of the CDF of the null distribution, irrespective of the number of points observed so far or the size of the sliding window, satisfying the memory constraint requirement of our application.

2.2. Parameter selection

The execution of both NUNC Local and Global requires the selection of various parameters, including the K quantiles $q_1, \ldots, q_k$ and threshold β. Additionally, the size of the sliding window W must be chosen with care. In practice, W be chosen based on specific knowledge of the application and data generating process at hand. We

Algorithm 2: NUNC Global Algorithm

```
**Data:** $x_{(t-W+1):W}$, the last $W$ realizations from a data generating process $X$; $z_W^{(t)}(q_k)$ for $q_1, \ldots, q_K$.
**Input:** $\beta > 0$; $K < W$; $q_{1:K}$ the fixed quantiles.
$\mathcal{Q} \leftarrow \sum_{k=1}^{K} 2\left[\mathcal{L}(x_{1:t-W}; q_k) + \mathcal{L}(x_{t-W+1:t}; q_k) - \mathcal{L}(x_{1:t}; q_k)\right]$;
**if** $\mathcal{Q} \geq K\beta$ **then**
    Return $t - W$ as a changepoint.
**else**
    $z_W^{(t+1)}(q_k) \leftarrow \frac{1}{t-W+1}\left[(t-W)z_W^{(t)}(q_k) + F_{(t-W+1):(t-W+1)}(q_k)\right]$    for $q_k \in q_{1:K}$.
**end**
```

defer further discussion of this until Section 3, where we consider the impact of W on different simulation scenarios, and Section 4 where we explore selecting W in a practical application by considering a range of different window sizes and selecting the one that offers the best detection power.

Next, we turn to the challenge of choosing the K quantiles $q_1, \ldots, q_k$. The value of K itself should be chosen to be proportionate to $\log(W)$, in line with the method proposed by Haynes et al. (2017). In particular, the value $K = \lceil 4\log(n) \rceil$ was proposed, see Haynes et al. (2017, Section 4.3) for details. Given K, one approach to choosing the $\{q_k\}$ would be to evaluate evenly spaced empirical quantiles. However, an alternative approach is motivated by Haynes et al. (2017, Section 3.1)). That is, we select q_k such that

$$q_k = \hat{F}^{-1}\left(1 + (2W+1)\exp\left[\frac{c}{K}(2k-1)\right]\right)^{-1}, \tag{11}$$

where $c = -\log(2W-1)$. The reason for making such a choice is that this gives a higher weight to values in the tail of the distribution (Haynes et al., 2017), allowing for more effective change detection. In the Local algorithm, the q_k will be updated as the window changes; in the Global algorithm, however, these K points are fixed in time. As such, the values of q_k must be obtained using the first W points of data the algorithm analyses. In some situations, however, this issue can be avoided because there is prior knowledge of the underlying distribution of the data. In this case known quantiles can be utilised rather than estimating them from the data.

Another important requirement for the two algorithms presented here, as in other sequential changepoint methods, is the ability to control the false alarm rate (Tartakovsky et al., 2014). In general, the value of β will be tuned so that the probability of a false alarm for data under the null hypothesis is set to some level α. This will be the case, for instance, in the telecommunications application where the threshold value will be tuned on devices where no even is detected. That said, we can follow a similar approach to that of Eichinger & Kirch (2018) to obtain an idea of how β relates to the probability of a false alarm. Indeed, we can (asymptotically) approximate the distribution of each term in the sum of equation (6) by a chi-squared-1 distribution (Wilks, 1938). This is the asymptotic distribution of the likelihood-ratio test for a fixed t, τ, and q, assuming independent identically distributed (i.i.d.) data. With this approximation, it can be shown that:

Proposition 2. *If β is chosen such that $\beta = \max\{\beta_1, \beta_2\}$, where*

$$\beta_1 = 1 - 8K^{-1}\log\left(\frac{\alpha}{W(t-W+1)}\right)$$
$$\beta_2 = 1 + 2\sqrt{2\log\left(\frac{W(t-W+1)}{\alpha}\right)},$$

then the probability of a false detection by time t using NUNC Local is bounded above by α.

Proof. This result follows from bounds on the tail of sums of chi-squared distributions and a Bonferonni correction – see the Appendix for details. □

It should be noted, that in situations where the window size is large this bound may be conservative due to it being an asymptotic bound. Furthermore, when the assumptions of data independence and identical distribution are not met, this bound may not hold, as illustrated in simulations. In such settings we suggest selecting a threshold by tuning on a data stream that does not contain any changes.

If using an approximate grid of size $J < W$, then it is necessary to replace the values of W in the above proposition with the value J. Furthermore, as a corollary to Proposition 2, a bound can be obtained for use in NUNC Global.

Corollary 1. *If β is chosen such that $\beta = \max\{\beta_1, \beta_2\}$, where*

$$\beta_1 = 1 - 8K^{-1} \log\left(\frac{\alpha}{(t - W + 1)}\right)$$

$$\beta_2 = 1 + 2\sqrt{2\log\left(\frac{(t - W + 1)}{\alpha}\right)},$$

then the probability of a false detection by time t using NUNC Global is bounded above by α.

Proof. The proofs follow similarly to Proposition 2, however we perform only one test, rather than W tests, per window. □

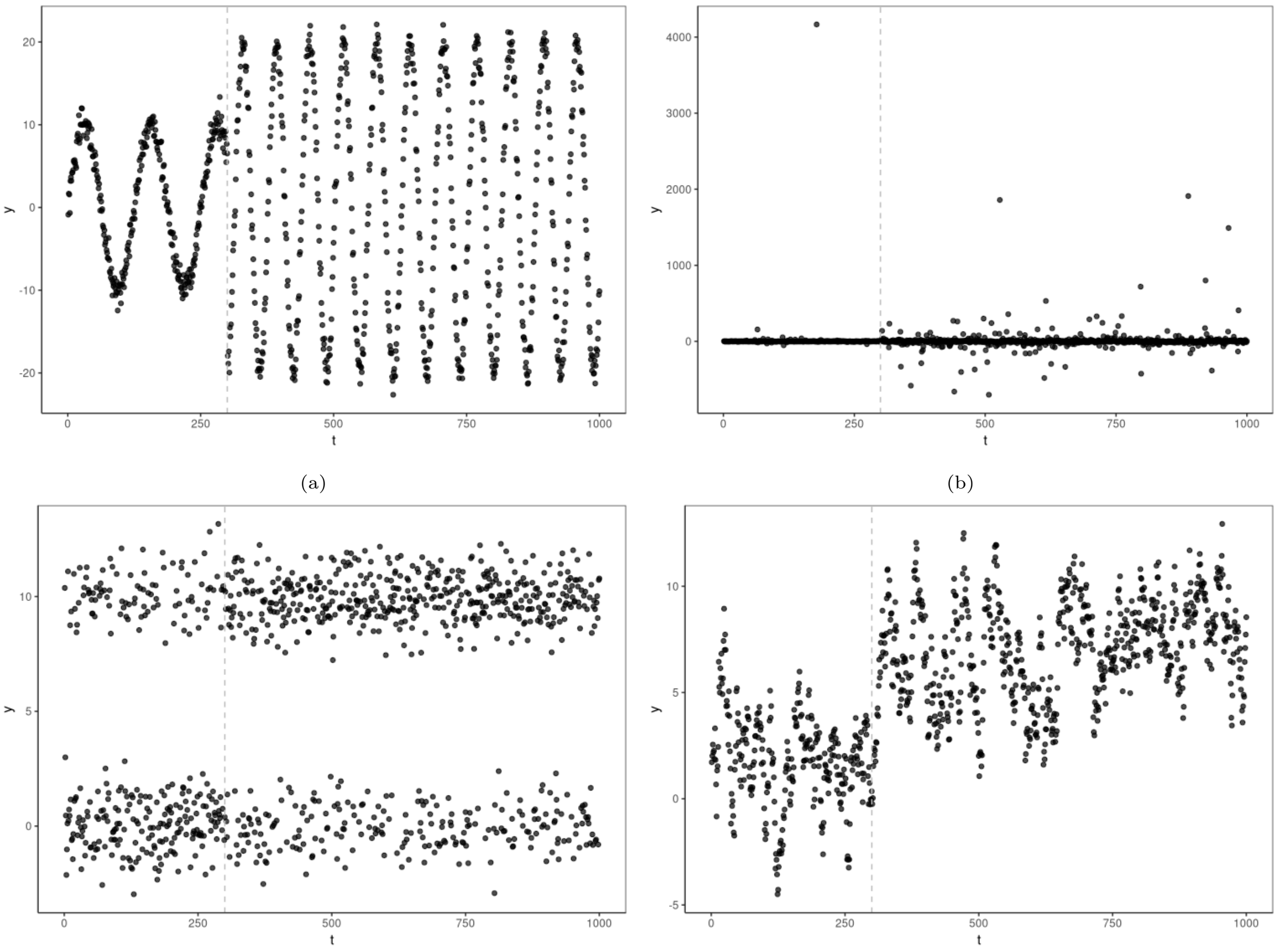
Due to the connection between our online methodology and the test of Zou et al. (2014) and Haynes et al. (2017) these results can also be extended to the offline setting. In particular selecting β, as per Proposition 2, as the penalty in these changepoint detection algorithms would control the probability of a false alarm in a segment of length t.

Now that the methodology behind NUNC has been presented, and methods for quantile and threshold selection discussed, we consider its performance within various simulation settings.

3. Simulation Study

In this section we examine the properties of both NUNC approaches in various simulation settings. These can be seen in Figure 2. The first (Figure 2(a)) is a change in the mixture proportions of a bi-modal Gaussian distribution, whilst the second example considers a change in the scale of a Cauchy Distribution. The third setting is a change in the amplitude of a sinusoidal process, and the final setting is that of a change in the drift parameter of an Ornstein-Uhlenbeck (OU) process. Both the sinusoidal and OU examples are included to highlight how NUNC performs when the independence assumption is not met. These latter two scenarios are also motivated by our telecommunications application as both exhibit drift. Realisations of each of these data generating processes can be seen in Figure 2.

In what follows, we explore the performance of each method across the four given scenarios. In particular we consider the influence of window size on the power, and the detection delay, of the test. In each setting we will also compare the NUNC-based tests against a MOSUM test, as implemented by Meier et al. (2021), a competitor non-parametric online changepoint algorithm that has a lightweight data footprint.



(a)

(b)

(c)

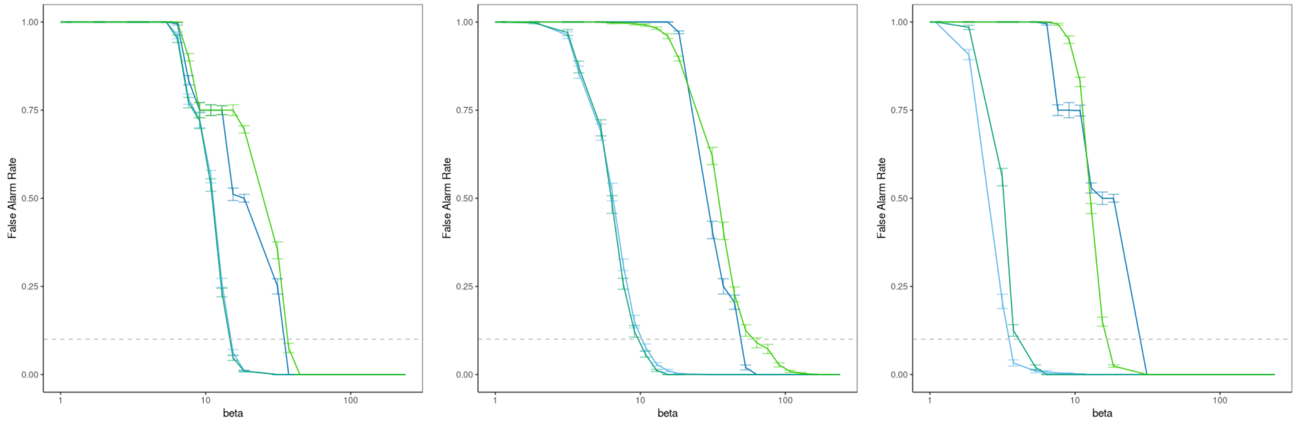
(d)

Figure 2: Four different simulations scenarios: (a) change in the amplitude of a sinusoidal process; (b) change in scale of a Cauchy distribution; (c) change in mixture proportions of a bi-modal Gaussian distribution; and (d) change in the drift parameter of an Ornstein–Uhlenbeck process.

3.1. False Alarm Probability

We begin by considering the false alarm rates returned by the three methods (NUNC Local, NUNC Global and MOSUM) for 100 replicates of each of our four data generating scenarios, without a change being present. In each case, the series generated was of length 1000, with $K = 20$ and $W = 150$ for NUNC Local and NUNC Global. The test was performed for a range of values of the test threshold, β. For comparison, we also compared against the equivalent MOSUM procedure (i.e. $W = 150$, and other settings set to default). The resulting false alarm rates for a range of thresholds can be seen in Figure 3.

To explore the practical utility of Proposition 2, we compare the thresholds required for the i.i.d. multi-modal Gaussian and Cauchy change-in-scale false alarm rate when seeking to achieve an illustrative false alarm rate of 10%. Our study highlights that penalty values of 9 and 10 are required by the NUNC Global algorithm for the multi-modal Gaussian and Cauchy scenarios respectively. In these two settings, penalties of 11.6 and 12.6 respectively were required for NUNC Local. This compares favourably with the approximate penalty values selected using Proposition 2 (9.51 and 12.30 for the Global and Local cases respectively). Unsurprisingly, in the case of the (non-i.i.d., temporally dependent) sinusoidal and OU scenarios, the penalties required for a 10% false alarm rate differ from those provided by Proposition 2.



(a) (b) (c)

Figure 3: The False Alarm rate for increasing threshold values for the four different simulation scenarios analysed with (a) NUNC Local, (b) NUNC Global and (c) MOSUM. The dotted line indicates a false alarm rate of 0.10, and the error bars indicate two standard deviations. In each plot the simulation scenarios are represented by sinusoidal change-in-amplitude (dark blue); the Cauchy change-in-scale (light blue); in emerald green, the change-in-mixture proportions (emerald green); and the change-in-drift in a OU process (light green).

3.2. Detection Power and Detection Delay

We now turn to consider the detection power and detection delay of the NUNC algorithms in a variety of settings. Following Tartakovsky et al. (2014), we define the detection power as the probability that a changepoint is detected after it has occurred, and the detection delay as the difference between the stopping time of the test and the time the changepoint is known to have emerged. Again we focus on 100 replicates of each of the four scenarios displayed in Figure 2, where each series is of length 1000 and the change occurs at time $t = 300$. In each case we seek to estimate the detection power and detection delay, controlling the false alarm rate at 10% by fixing the value of β using the results from Section 3.1, and allowing the window size W of the algorithms to vary.

Results for the detection power, and detection delay, are summarised in Tables 1 and 2 respectively. It is notable that NUNC is able to detect changes in a variety of settings, including those where the data has a time-dependent structure. We also note that NUNC Global outperforms NUNC Local in most cases, except when the underlying distribution is sinusoidal. This is perhaps to be expected since NUNC Global incorporates the long-run empirical CDF which stores the historical data. This allows for better identification of departures from the null when the data is stationary. When the data is non-stationary, however, this is not so beneficial and so the performance of NUNC Local is comparable.

Turning to consider the results obtained for the detection delay, displayed in Table 2, it is evident that NUNC Local demonstrates stronger performance than that of NUNC Global. This is as expected, because NUNC Global checks if the distribution of the data in the window differs from the long-run empirical CDF, whereas NUNC Local checks each point in the window (after pruning as per Proposition 1) for a changepoint within the window.

In comparison to the MOSUM, the detection power of NUNC typically exceeds it except in the specific case of multi-modal data being analysed with a large window. The reason MOSUM performs so well in this case is due to the fact that the change in mixture proportions can also be cast as a change in mean. For the sinusoidal process, however, the non-stationarity of the data means that the threshold that is required is too high for detection to take place.

The results in Table 1 also illustrate how, as one might expect, the performance of NUNC Local improves for stationary data as the size of the window increases. Specifically, the larger window provides a better estimate of the CDF of the (stationary) data stream, which in turn makes it easier to identify when a change has occurred. The price for this increased power, however, comes in the form of an increase in computational cost due to the larger window size. As such, there is a trade-off between detection power and the computational burden of NUNC Local. For NUNC Global, on the other hand, in many situations the use of the long-run CDF provides a better estimate of the distribution under the null. This somewhat reduces the need to increase the window size.

4. Application

We now revisit the telecommunications example, briefly introduced in Section 1, to explore the utility of NUNC in this setting. Recall that the data consists of historic records of a key operational metric routinely monitored on devices that have limited computational power and data storage capability. We have records for 473 such devices, of which 133 were known to contain a (series specific) event that triggered a user intervention. Due to the specifics of the application, engineers believed that it is possible to identify the start of the event in the operational data *before* a user identifies and makes an intervention. If this is true, then it would be desirable to identify the start of changing structure in advance of the user identifying and making an intervention. We call the correct detection of such a change in advance of user identification, an ‘anticipation’. Conversely, the detection of such an event before it is resolved is called a ‘detection’. The aim of this exploratory analysis, therefore, is to identify to what extent NUNC can (a) identify the correct (event-containing) series and (b) to what extent it can be used to ‘anticipate’ or ‘detect’.

Before summarising the results, we briefly discuss the various parameter choices made: specifically, the threshold β, the window size W, and the choice of K. In line with Haynes et al. (2017), we choose $K = \lceil 4\log(W) \rceil$. The choice of β was made to control the false alarm rate at a desired level after discussion with domain experts. In this particular setting, false alarms can be tolerated if this results in improved identification of real events. Consequently a false alarm rate of 15% was selected. In order to identify the appropriate value of β to achieve this, we first fix a window size and then perform NUNC on the 340 data series without the event, choosing a value of β that gives the desired false alarm rate. NUNC is then applied to the 133 series known to contain an event using this β, for the chosen window size. This is repeated for a range of different window sizes, with the value of W offering the best anticipation power utilised. We note that other metrics could be used for this assessment, however our application concerns event anticipation. A similar process is also used to implement the MOSUM test; again, the threshold is chosen to control the false alarm rate at 15% for a given window size.

Window	50			100			150			200		
Process	Local	Global	MOSUM	Local	Global	MOSUM	Local	Global	MOSUM	Local	Global	MOSUM
Cauchy	0.56	**0.9**	0.02	0.71	**0.89**	0.05	0.81	**0.91**	0.02	0.85	**0.91**	0.01
Multimodal	0.45	**0.87**	0.43	0.31	**0.79**	0.71	0.58	0.8	**0.92**	0.58	0.83	**0.92**
Sinusoidal	**0.95**	0	0.93	**1**	0.92	0	0.17	**0.91**	0	**0.98**	0.92	0
OU	0.25	**0.92**	0.42	0.47	**0.91**	0.68	0.36	**0.9**	0.86	0.79	**0.93**	**0.93**

Table 1: Table showing the detection power of NUNC Local, NUNC Global, and the MOSUM for various window sizes, with the best performance for each scenario highlighted in bold.

Window	50			100			150			200		
Process	Local	Global	MOSUM	Local	Global	MOSUM	Local	Global	MOSUM	Local	Global	MOSUM
Cauchy	150.16	**67.13**	345	**51.67**	109.39	211	**72.99**	133.64	494	**48.96**	143.69	285
Multimodal	285.29	**131.46**	169.21	285.82	238.63	**49.23**	242.59	266.96	**48.03**	206.09	297.05	**59.07**
Sinusoidal	**19.59**	Inf	411.69	**5**	102.03	Inf	**9.16**	159.3	Inf	**7.58**	238.61	Inf
OU	284.32	**115.88**	173.88	246.71	136.81	**77.65**	242.85	173.01	**78.83**	155.7	221.06	**81.34**

Table 2: Table showing the average detection delay of NUNC Local, NUNC Global, and the MOSUM for various window sizes, with the best performance for each scenario highlighted in bold. Note that in line with Tartakovsky et al. (2014), if no detection takes place then this corresponds to a detection delay of infinity.

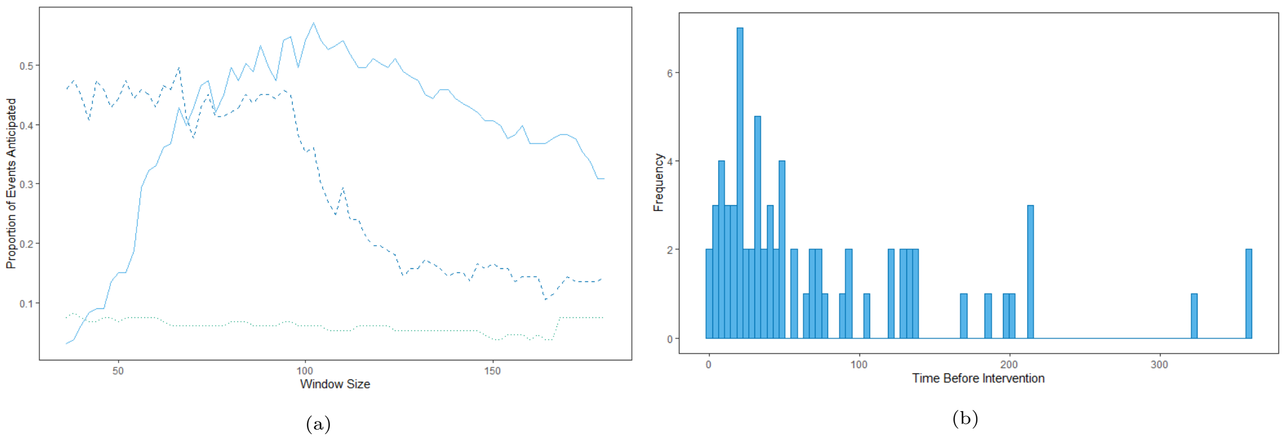


Figure 4: Comparison of anticipation rates achieved by the Local (Solid Line) and Global (Dashed Line) variants of NUNC, and the MOSUM (Dotted Line), for varying window sizes, for a false alarm rate of 15% in (a). A histogram illustrating the distribution of the time between the detection of an event by NUNC Local and the report by a customer, for a window size of 100 and a false alarm rate of 15% in (b).

In Figure 4(a) we present the anticipation rate for a range of window sizes. As can be seen, a window of size between 80 and 120 performs the best, with NUNC Local correctly identifying (i.e. anticipating) $> 50\%$ of events in advance of user intervention. We also note that NUNC outperforms the MOSUM for various choices of W. One reason for this is because the MOSUM test threshold is set to avoid detecting the non-anomalous changes in mean that many of the series exhibit, and this reduces detection power.

False Alarms	1%	5%	10%	15%
Local	0.08 (0.37)	0.28 (0.59)	0.38 (0.68)	0.51 (0.77)
Global	0.03 (0.36)	0.06 (0.51)	0.20 (0.67)	0.35 (0.76)
MOSUM	0.02 (0.02)	0.04 (0.08)	0.05 (0.10)	0.06 (0.12)

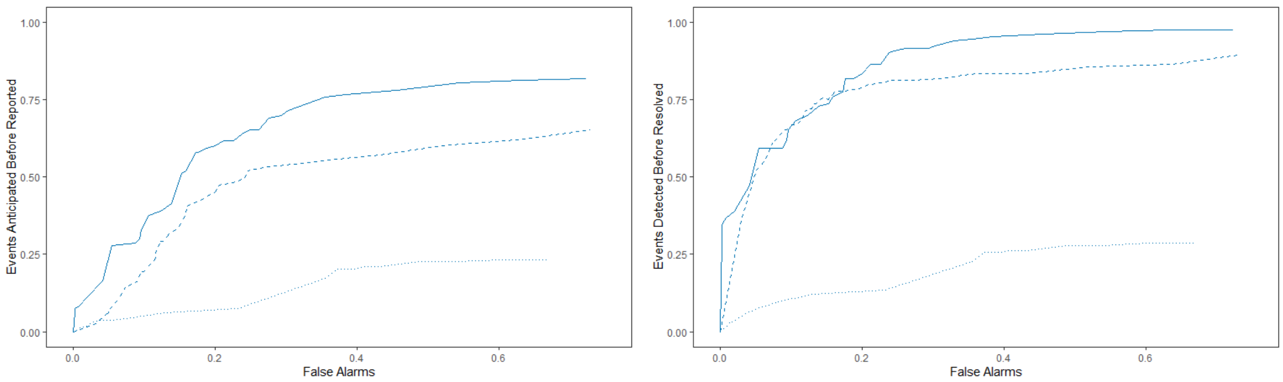
Table 3: Table illustrating proportion of events anticipated (and detected) for varying rates of false alarms for both NUNC Local and NUNC Global.

Finally, for a fixed window size ($W = 100$) we explore the anticipation and detection rate as the false alarm rate (or equivalently β) varies. The results are summarised in Table 3, and Figure 5.

From the results presented, one remark that can be made is that the detection power for NUNC Local and NUNC Global is similar, with both achieving over 75% for a false alarm rate of 15% and window of $W = 100$, but the anticipation power of NUNC Local is significantly better for a range of false alarm rates and window sizes. As in the simulations on detection delay, this is due to the way that NUNC Local checks every point within the window for a change, allowing for a shorter detection delay. A second observation is that NUNC achieves better results than the MOSUM in terms of both anticipation, and power, as the false alarm rate varies.

5. Concluding Remarks

This paper has introduced NUNC Local and NUNC Global: two related, non-parametric changepoint methods with a lightweight data footprint. NUNC Global offers greater power in instances where there is a stationary underlying null distribution for the data (*cf.* Section 3.2). Conversely NUNC Local shows greater resilience, and is able to outperform NUNC Global, in settings where the process contains time-dependent or other non-independent structure, such as the telecommunications example that we consider. Finally, we have explored how both forms



(a)

(b)

Figure 5: Comparison of event anticipation (a) and event detection (b) rates achieved by both the Local (Solid Line) and Global (Dashed Line) variants of NUNC, and the MOSUM (dotted line), for a window of size $W = 100$ and a varying false alarm rate.

of NUNC compare against MOSUM, an existing non-parametric window based changepoint detection test with a lightweight footprint, in both simulations and the telecommunications application.

To support the implementation of the algorithms, we also consider their computational properties and key theoretical results. The first of these provides a cutoff value for the sliding window in NUNC Local, so that only relevant parts of the sliding window are checked for a change. This provides a theoretical justification for reducing the computational cost of the algorithm. To search within the window, the evenly spaced grid of values suggested by Proposition 7 favours detection power in spite of detection delay. To improve on the detection delay, one might consider the use of a logarithmic grid of values as a search strategy. This can be achieved starting again from the center of the window, but including more values towards the right side of the window following a logarithmic increase. This introduces a trade-off between computational efficiency, detection delay and power of the test. Lastly, a method for (approximately) selecting the threshold was also provided, enabling the control of the false alarm rate at a fixed level.

As with any approach, NUNC has various weaknesses that might be identified for criticism. For example, the estimation of the empirical CDF from windowed data means that gradual changes are likely to go undetected. In addition, as identified by our simulation study, NUNC Global struggles with changing structure in time-dependent series. The investigation of potential alternatives to the NUNC framework, that can resolve such weaknesses, are left as avenues for future research.

The reason for developing NUNC was to enable to data-driven identification of an event that would anticipate a user intervention related to a telecommunications device, rather than simply waiting for direct communication from end-users. As seen in Section 4 this can be achieved with some success, with NUNC Local allowing for detection in advance of the event in a meaningful number of cases. This provides a powerful diagnostic tool, and demonstrates the ability of NUNC to successfully detect changes that anticipate user intervention in a large number of cases.

6. Acknowledgements

The authors are grateful to David Yearling at BT for the helpful conversations that helped motivate this research, and Dan Grose for advice on efficient computational implementation. In addition Fearnhead and Romano acknowledge the financial support of the EPSRC and Lancaster University via grant EP/N031938/1. Austin and Eckley also gratefully acknowledge the support of BT and EPSRC via the NG-CDI Prosperity Partnership (EP/N031938/1). Open-source code, making the methods developed in this paper accessible, will be made available via an appropriate repository in due course.

References

Chakraborti, S., & van de Wiel, M. A. (2008). A nonparametric control chart based on the mann-whitney statistic. *Institute of Mathematical Statistics Collections*, (p. 156–172). URL: http://dx.doi.org/10.1214/193940307000000112. doi:10.1214/193940307000000112.

Chen, H. (2019). Sequential change-point detection based on nearest neighbors. *The Annals of Statistics*, *47*, 1381–1407. doi:10.1214/18-AOS1718.

Chu, C.-S. J., Hornik, K., & Kuan, C.-M. (1995). Mosum tests for parameter constancy. *Biometrika*, *82*, 603–617. URL: http://www.jstor.org/stable/2337537. doi:10.2307/2337537.

Coelho, M., Graham, M., & Chakraborti, S. (2017). Nonparametric signed-rank control charts with variable sampling intervals. *Quality and Reliability Engineering International*, *33*, 2181–2192. URL: https://doi.org/10.1002/qre.2177. doi:10.1002/qre.2177.

Cormen, T., Cormen, T., Leiserson, C., Books24x7, I., of Technology, M. I., Press, M., Rivest, R., Stein, C., & Company, M.-H. P. (2001). *Introduction To Algorithms*. Introduction to Algorithms. MIT Press. URL: https://books.google.co.uk/books?id=NLngYyWFl_YC.

Eichinger, B., & Kirch, C. (2018). A MOSUM procedure for the estimation of multiple random change points. *Bernoulli*, *24*, 526 – 564. URL: https://doi.org/10.3150/16-BEJ887. doi:10.3150/16-BEJ887.

Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, *46*, 1–37.

Gordon, L., & Pollak, M. (1994). An Efficient Sequential Nonparametric Scheme for Detecting a Change of Distribution. *The Annals of Statistics*, *22*, 763 – 804. URL: https://doi.org/10.1214/aos/1176325495. doi:10.1214/aos/1176325495.

Hawkins, D. M., & Deng, Q. (2010). A nonparametric change-point control chart. *Journal of Quality Technology*, *42*, 165–173. URL: https://doi.org/10.1080/00224065.2010.11917814. doi:10.1080/00224065.2010.11917814.

Haynes, K., Fearnhead, P., & Eckley, I. A. (2017). A computationally efficient nonparametric approach for changepoint detection. *Statistics and Computing*, *27*, 1293–1305. URL: https://doi.org/10.1007/s11222-016-9687-5. doi:10.1007/s11222-016-9687-5.

Kirch, C., Weber, S. et al. (2018). Modified sequential change point procedures based on estimating functions. *Electronic Journal of Statistics*, *12*, 1579–1613. URL: https://doi.org/10.1214/18-EJS1431. doi:10.1214/18-EJS1431.

Liu, L., Zi, X., Zhang, J., & Wang, Z. (2013). A sequential rank-based nonparametric adaptive ewma control chart. *Communications in Statistics - Simulation and Computation*, *42*, 841–859. URL: https://doi.org/10.1080/03610918.2012.655829. doi:10.1080/03610918.2012.655829.

Meier, A., Kirch, C., & Cho, H. (2021). mosum: A package for moving sums in change-point analysis. *Journal of Statistical Software*, *97*, 1–42. URL: https://www.jstatsoft.org/v097/i08. doi:10.18637/jss.v097.i08.

Mukherjee, A., & Chakraborti, S. (2012). A distribution-free control chart for the joint monitoring of location and scale. *Quality and Reliability Engineering International*, *28*, 335–352. URL: https://doi.org/10.1002/qre.1249. doi:10.1002/qre.1249.

Murakami, H., & Matsuki, T. (2010). A nonparametric control chart based on the mood statistic for dispersion. *The International Journal of Advanced Manufacturing Technology*, *49*, 757–763. URL: https://doi.org/10.1007/s00170-009-2439-3. doi:10.1007/s00170-009-2439-3.

Padilla, O. H. M., Athey, A., Reinhart, A., & Scott, J. G. (2019). Sequential nonparametric tests for a change in distribution: An application to detecting radiological anomalies. *Journal of the American Statistical Association*, *114*, 514–528. URL: https://doi.org/10.1080/01621459.2018.1476245. doi:10.1080/01621459.2018.1476245.

Pepelyshev, A., & Polunchenko, A. (2017). Real-time financial surveillance via quickest change-point detection methods. *Statistics and its interface*, *10*, 93–106. URL: https://doi.org/10.48550/arXiv.1509.01570. doi:10.4310/SII.2017.v10.n1.a9.

Ross, G. J., & Adams, N. M. (2012). Two nonparametric control charts for detecting arbitrary distribution changes. *Journal of Quality Technology*, *44*, 102–116. URL: https://doi.org/10.1080/00224065.2012.11917887. doi:10.1080/00224065.2012.11917887.

Ross, G. J., Tasoulis, D. K., & Adams, N. M. (2011). Nonparametric monitoring of data streams for changes in location and scale. *Technometrics*, *53*, 379–389. URL: https://doi.org/10.1198/TECH.2011.10069. doi:10.1198/TECH.2011.10069.

Siegmund, D. (2013). Change-points: From sequential detection to biology and back. *Sequential Analysis*, *32*, 2–14. doi:10.1080/07474946.2013.751834.

Tartakovsky, A., Nikiforov, I., & Basseville, M. (2014). *Sequential Analysis: Hypothesis Testing and Changepoint Detection*. CRC Press.

Tartakovsky, A. G., Rozovskii, B. L., & Shah, K. (2005). A nonparametric multichart cusum test for rapid intrusion detection. In *Proceedings of Joint Statistical Meetings* (p. 11). volume 7.

Vershynin, R. (2018). High-dimensional probability: An introduction with applications in data science. In *High-Dimensional Probability: An Introduction with Applications in Data Science* Cambridge Series in Statistical and Probabilistic Mathematics (p. 37). Cambridge University Press. URL: https://books.google.co.uk/books?id=NDdqDwAAQBAJ.

Wainwright, M. (2019). *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press. URL: https://books.google.co.uk/books?id=IluHDwAAQBAJ.

Wang, D., Zhang, L., & Xiong, Q. (2017). A non parametric cusum control chart based on the mann–whitney statistic. *Communications in Statistics - Theory and Methods*, *46*, 4713–4725. doi:10.1080/03610926.2015.1073314.

Wilks, S. S. (1938). The Large-Sample Distribution of the Likelihood Ratio for Testing Composite Hypotheses. *The Annals of Mathematical Statistics*, *9*, 60 – 62. URL: https://doi.org/10.1214/aoms/1177732360. doi:10.1214/aoms/1177732360.

Zou, C., Yin, G., Feng, L., Wang, Z. et al. (2014). Nonparametric maximum likelihood approach to multiple change-point problems. *The Annals of Statistics*, *42*, 970–1002. URL: https://doi.org/10.1214/14-AOS1210. doi:10.1214/14-AOS1210.

7. Appendix

7.1. Proof of Proposition 1

Proof. In order to prove the desired bound, we focus on the extreme case where either $x_{1:\tau} < q$ and $x_{\tau+1:W} > q$, or vice versa, and note that this case maximises the expression

$$\mathcal{L}(x_{t-W+1:\tau}; q) + \mathcal{L}(x_{\tau+1:t}; q) - \mathcal{L}(x_{t-W+1:t}; q), \tag{12}$$

for any quantile q.

By writing out the likelihoods in the above equation, it can be observed that $\mathcal{L}(x_{t-W+1:\tau}; q) = 0$, $\mathcal{L}(x_{\tau+1:t}; q) = 0$, and

$$\mathcal{L}(x_{t-W+1:t}; q) = \frac{\tau}{W} \log \frac{\tau}{W} - (W-\tau) \log \left(\frac{W-\tau}{W} \right) \tag{13}$$

in the case where each $x_{1:\tau} > q$ and $x_{\tau+1:W} < q$. We also have

$$\mathcal{L}(x_{t-W+1:t}; q) = -\frac{\tau}{W} \log \frac{\tau}{W} + (W-\tau) \log \left(\frac{W-\tau}{W} \right) \tag{14}$$

when $x_{1:\tau} < q$ and $x_{\tau+1:W} > q$. Given both $\frac{\tau}{W}$ and $\frac{W-\tau}{W}$ are less than one, the log terms in equations (13) and (14) are both negative. As such, we can bound both equations (13) and (14) above by the following bound:

$$\mathcal{L}(x_{t-W+1:t}; q) \leq -\frac{\tau}{W} \log \frac{\tau}{W} - (W-\tau) \log \left(\frac{W-\tau}{W} \right).$$

As this case dealt with the maximum of equation (12), this then means that for any quantile q and window of data $x_1, \ldots, x_W$ we have that

$$\mathcal{L}(x_{t-W+1:\tau}; q) + \mathcal{L}(x_{\tau+1:t}; q) - \mathcal{L}(x_{t-W+1:t}; q) \leq -\frac{\tau}{W} \log \frac{\tau}{W} - (W-\tau) \log \left(\frac{W-\tau}{W} \right), \tag{15}$$

as required. Additionally, we note that this equation is decreasing in τ for fixed W.

We then consider the test statistic, given by equation (5). As a result of the bound in equation (15) if

$$2K \left[-\frac{\tau}{W} \log \frac{\tau}{W} - (W-\tau) \log \left(\frac{W-\tau}{W} \right) \right] \leq K\beta$$

then detection is impossible, as the bound for the test statistic does not exceed the threshold for the test. As a result, we conclude that if

$$-\frac{\tau^*}{W} \log \frac{\tau^*}{W} - (W-\tau^*) \log \left(\frac{W-\tau}{W} \right) \leq \frac{\beta}{2}$$

then due to the fact the expression decreases as τ increases then for $\tau > \tau^*$ detection is impossible. This completes the proof. □

7.2. Proof of Proposition 2

Proof. A false alarm by the time t under NUNC Local can be written as

$$\begin{aligned} &= \mathrm{P}\left(\bigcup_{s=W}^{t} \max_{s-W+1 \leq \tau \leq s} \sum_{k=1}^{K} 2 \left[\mathcal{L}(x_{s-W+1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:s}; q_k) - \mathcal{L}(x_{s-W+1:s}; q_k) \right] \geq K\beta \right) \\ &\leq \sum_{s=W}^{t} \sum_{\tau=s-W+1}^{s} \mathrm{P}\left(\sum_{k=1}^{K} 2 \left[\mathcal{L}(x_{s-W+1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:s}; q_k) - \mathcal{L}(x_{s-W+1:s}; q_k) \right] \geq K\beta \right) \end{aligned} \tag{16}$$

The next part of the proof uses the fact that, under the i.i.d. assumption, asymptotically for any quantile the following holds

$$2\left[\mathcal{L}(x_{s-W+1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:s}; q_k) - \mathcal{L}(x_{s-W+1:s}; q_k)\right] \sim \chi^2_1.$$

As in Wainwright (2019), it can be shown that if a random variable X_i follows a χ^2_1 distribution then it is Sub-Exponential with parameters 4 and 4. That is, $X_i \sim \mathrm{SE}(4, 4)$.

Under dependence, as is the case between different quantiles, if $X_i \sim \mathrm{SE}(\nu_i^2, b_i)$ then $\sum_{i=1}^n X_i - \mathrm{E}(X_i) \sim \mathrm{SE}\left(\left(\sum_{i=1}^n \nu_i\right)^2, \max_i b_i\right)$, and so

$$\sum_{k=1}^{K} \left[\mathcal{L}(x_{s-W+1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:s}; q_k) - \mathcal{L}(x_{s-W+1:s}; q_k)\right] - K \sim \mathrm{SE}(4K^2, 4).$$

We then use a well known bound on the subexponential tail (Vershynin, 2018) to obtain

$$\begin{aligned}
&\sum_{s=W}^{t} \sum_{\tau=s-W+1}^{s} \mathrm{P}\left(\sum_{k=1}^{K} 2\left[\mathcal{L}(x_{s-W+1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:s}; q_k) - \mathcal{L}(x_{s-W+1:s}; q_k)\right] \geq K\beta\right) \\
&= \sum_{s=W}^{t} \sum_{\tau=s-W+1}^{s} \mathrm{P}\left(\sum_{k=1}^{K} 2\left[\mathcal{L}(x_{s-W+1:\tau}; q_k) + \mathcal{L}(x_{\tau+1:s}; q_k) - \mathcal{L}(x_{s-W+1:s}; q_k)\right] - K \geq K\beta - K\right) \\
&\leq W(t - W + 1) \exp\left(-\frac{1}{2} \min\left\{\frac{K\beta - K}{4}, \frac{(K\beta - K)^2}{4K^2}\right\}\right). \qquad (17)
\end{aligned}$$

We can set this final line equal to α to control our desired false alarm rate. We have two cases in equation (17) and must bound above by the largest of these. The first case is that

$$W(t - W + 1) \exp\left(-\frac{K\beta_1 - K}{8}\right) = \alpha$$

in which case we choose

$$\beta_1 = 1 - 8K^{-1} \log\left(\frac{\alpha}{W(t - W + 1)}\right).$$

On the other hand we have the case where

$$W(t - W + 1) \exp\left(-\frac{(K\beta_2 - K)^2}{8K^2}\right) = \alpha$$

and solving this gives

$$\beta_2 = 1 + 2\sqrt{2 \log\left(\frac{W(t - W + 1)}{\alpha}\right)}.$$

We then choose the larger of β_1 and β_2, completing the proof. □