Statistics and Computing (2020) 30:1241–1254
https://doi.org/10.1007/s11222-020-09943-9

Classification of periodic arrivals in event time data for filtering computer network traffic

Francesco Sanna Passino[1] · **Nicholas A. Heard**[1]

Received: 22 May 2019 / Accepted: 8 April 2020 / Published online: 24 April 2020

Abstract
Periodic patterns can often be observed in real-world event time data, possibly mixed with non-periodic arrival times. For modelling purposes, it is necessary to correctly distinguish the two types of events. This task has particularly important implications in computer network security; there, separating automated polling traffic and human-generated activity in a computer network is important for building realistic statistical models for normal activity, which in turn can be used for anomaly detection. Since automated events commonly occur at a fixed periodicity, statistical tests using Fourier analysis can efficiently detect whether the arrival times present an automated component. In this article, sequences of arrival times which contain automated events are further examined, to separate polling and non-periodic activity. This is first achieved using a simple mixture model on the unit circle based on the angular positions of each event time on the p-clock, where p represents the main periodicity associated with the automated activity; this model is then extended by combining a second source of information, the time of day of each event. Efficient implementations exploiting conjugate Bayesian models are discussed, and performance is assessed on real network flow data collected at Imperial College London.

Keywords Circular statistics · Network flow data · Mixture modelling · Periodic arrival times · Periodicity detection · Statistical cyber-security · Wrapped normal

1 Introduction

Event time data exhibit periodic behaviour in many real-life applications, for example astrophysics (Cicuttin et al. 1998), bioinformatics (Kocak et al. 2013), object tracking (Li et al. 2010) and computer networks (Heard et al. 2014; Price-Williams et al. 2017). The periodic arrival times can often be mixed with non-periodic events. Therefore, to model the generating process appropriately, it is required to correctly distinguish the event types. This article proposes a statistical method for classification of periodic arrivals within a sequence of event times.

This work is motivated by important applications in computer network security. In particular, network flow (NetFlow) data are analysed. Network flow (NetFlow) data provide information about Internet Protocol (IP) connections between nodes in a computer network and have been successfully used to monitor network traffic (Hofstede et al. 2014). These data are routinely collected in bulk at internet routers, providing large databases of IP address connections. Commonly, a large proportion of the connections from a network host can be ascribed to legitimate, automated polling to various services. It is therefore an important step in the model-building process to be able to correctly identify which connections are due to the presence of a human at the machine, and which others are purely automated. Making this distinction is crucial for network monitoring and statistical intrusion detection: anomalies related to the presence of an intruder within the network will be significantly easier to detect when the polling connections are filtered out from the analysis. Realistic modelling strategies seek to treat the two components separately: Price-Williams and Heard (2020) show that a nonparametric Wold process with step function excitation is a suitable choice for modelling human events

The authors gratefully acknowledge funding from the EPSRC and the Heilbronn Institute for Mathematical Research.

✉ Francesco Sanna Passino
francesco.sanna-passino16@imperial.ac.uk

Nicholas A. Heard
n.heard@imperial.ac.uk

[1] Department of Mathematics, Imperial College London, 180 Queen's Gate, London SW7 2AZ, UK

in computer network traffic data. That model can only be applied when periodic connections are not present: this article provides a statistical framework for filtering automated traffic when human events are mixed with polling connections.

A useful network-wide filtering approach for polling behaviour, based on Fisher's g-test for periodicities, was proposed in Heard et al. (2014). For each pair of network nodes, the method looks for strong peaks in the periodogram of the event series of connections along that edge. The methodology is specifically developed in the context of computer network data, but it can be applied to any sequence of arrival times. A limitation of this approach is that all connections from an edge are deemed to be automated if the maximal periodicity for that edge is found to be significant, whereas activity on some network edges can contain a mixture of both automated and human activity. For example, connections to an email server are continuously refreshed with a fixed periodicity, but the user might also manually ask if new messages have been received. It is therefore potentially valuable to further understand which of the events on such edges are actually associated with the presence of a user. This article aims to complement the existing methodologies and provide a data filtering algorithm for network connection records, where each connection on an edge will be classified as periodic or non-periodic through a mixture probability model. Note that the aim of the paper is not to discern malicious automated activities, such as those generated by botnets, from human activities, but to provide a statistical technique for separating purely automated, polling activity, either malicious or legitimate, from non-periodic connections, which also include human activity.

The problem of periodicity detection in computer network traffic has been extensively studied in the computer science literature. Common approaches include spectral analysis (Barbosa et al. 2012; AsSadhan and Moura 2014; Heard et al. 2014; Price-Williams et al. 2017), which are often combined with thresholding methods (Bartlett et al. 2011; Huynh et al. 2016; Chen et al. 2016). Alternatives include modelling of inter-arrival times (Bilge et al. 2012; Qiao et al. 2012; Hubballi and Goyal 2013), where distributional assumptions are imposed and the behaviour is tested under the null of no periodicities (He et al. 2009; McPherson and Ortega 2011). Finally, some authors identify signals of periodicities in the autocorrelation function (Gu et al. 2008; Qiao et al. 2013), using changepoint methods (Price-Williams et al. 2017) or summary statistics computed sequentially in time windows (Eslahi et al. 2015). Price-Williams et al. (2017) also use wrapped distribution for detecting automated subsequences of events, and their methodology is able to handle changes in the periodicity and parameters in the model, but the human activity within a periodic subsequence is not captured. Most models proposed in the literature are aimed at classifying the entire edge as purely periodic or non-periodic. The model proposed in this article further analyses the edges with dominant periodicities, with the objective of recovering the human connections, when present; each observation is separately classified as periodic or non-periodic. The models described in this article could make a direct contribution to real-world network analysis, providing an efficient method for separating human and polling connections on the same edge, allowing deployment of the existing methodologies (Price-Williams and Heard 2020) for analysis of the filtered events.

The remainder of the article is organised as follows: Sect. 2 summarises the use of Fisher's g-test for identifying the dominant periodicity in event time data. Using that periodicity, Sect. 3 introduces two transformations of event times which will be used to classify individual events as periodic or non-periodic. Models for these two quantities are presented in Sects. 4 and 5, respectively. Applications on real and synthetic data are discussed in Sect. 6.

2 Fisher's *g*-test for detecting periodicities in event time data

Let $t_1 < t_2 < \cdots < t_N$ be a sequence of arrival times, and $N(\cdot)$ be a counting process recording the number of events over time. In the computer network application, $N(\cdot)$ counts connections over time from the client to the server, for any particular client and server pair. It is most practical to treat $N(\cdot)$ as a discrete-time process, with connection counts aggregated within bins of fixed width δ. Thus, $N(t)$ will denote the number of events after $t\delta$ seconds. The increments of the process are the corresponding bin counts $\mathrm{d}N(t) = N(t) - N(t-1)$.

After T time units of observation, the discrete Fourier transform for the zero-mean corrected process yields the periodogram

$$\hat{S}(f) = \frac{1}{T}\left|\sum_{t=1}^{T}\{\mathrm{d}N(t) - N(T)/T\}\mathrm{e}^{-2\pi\imath ft}\right|^2.$$

The fast Fourier transform (FFT) allows efficient computation of $\hat{S}(f_k)$ at the discrete-time Fourier frequencies $f_k = k/T, k = 1, \ldots, m$ where $m = \lfloor T/2 \rfloor$, in $\mathcal{O}(T \log T)$ operations. Peaks in the periodogram values might correspond to periodic signals in the sequence of arrival times. Fisher (1929) proposed an exact test for the null hypothesis of no periodicities using the *g-statistic*,

$$g(\hat{S}) = \frac{\max_{1\le k\le m}\hat{S}(f_k)}{\sum_{1\le j\le m}\hat{S}(f_j)}. \tag{1}$$

The test arises in the theory of harmonic time series analysis and is the uniformly most powerful symmetric invariant procedure (Anderson 1971) against the alternative hypothesis of a periodicity existing at a single Fourier frequency, for a null hypothesis of a white noise spectrum (for further details, see Percival and Walden 1993). Under such a null hypothesis, Fisher (1929) derived an exact p-value for a realised value g of $g(\hat{S})$, for which there also exists a convenient asymptotic approximation (Jenkins and Priestley 1957):

$$\mathbb{P}\{g(\hat{S}) > g\} = \sum_{j=1}^{\min\{\lfloor 1/g \rfloor, m\}} (-1)^{j-1} \binom{m}{j} (1 - jg)^{m-1} \approx 1 - \{1 - \exp(-mg)\}^m. \quad (2)$$

If an sequence of arrival times is found to be periodic at a given significance level, the corresponding *period* is

$$p = \delta \left\{ \text{argmax}_{f_k : 1 \le k \le m} \hat{S}(f_k) \right\}^{-1}. \quad (3)$$

In a Bayesian setting, methods to detect periodicities have been developed in astrophysics and astrostatistics (Jaynes 1987), or in biostatistics and bioinformatics (de Lichtenberg et al. 2005; Kocak et al. 2013). None of these methods are fully scalable and as easy to interpret as the g-test; therefore, for the purposes of this work, the periodicities will be obtained using (1) and the corresponding p-value (2).

In computer network traffic, if the p-value is below a pre-specified small significance level, then the *entire* edge is deemed to be periodic. Otherwise, if an edge is found to be not significantly periodic, then it is assumed that the majority of the activity on that edge can be ascribed to non-periodic events, possibly related to the presence of a human at the machine. If an edge is classified as periodic using the g-test, it is also possible that the observed connections contain a mixture of both polling and human activity. The objective of this paper is to further refine the classification performance for such mixed-type edges, classifying not only the entire edge activity as periodic or non-periodic, but each observed event on the edge.

The performance of the g-test on mixtures of periodic and non-periodic event times can be investigated via simulation. A sequence of 1000 events repeating every $p = 10$ s is generated and mixed with events generated from a Poisson process on the same time frame, with different rates λ. For each value of λ, the simulation is repeated 100 times to estimate the expected p-value (2) from the g-test and the results are reported in Fig. 1. For interpretability, the mean proportion of periodic events, which is monotonically decreasing in λ, is plotted on the horizontal axis. It is clear from Fig. 1 that the expected p-value decreases when the proportion of periodic events increases, but the p-value is sufficiently small

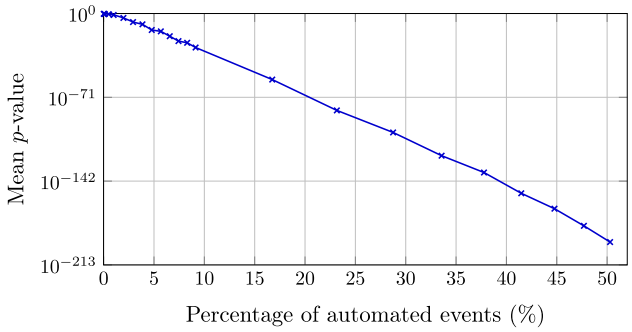


Fig. 1 Expected p-value for the g-test against the percentage of periodic events

even when the proportion of automated events is small. For example, for a 2% percentage of polling arrival times, the resulting expected p-value of the g-test is ≈ 0.0001.

3 Circular statistics for classifying event times

Let $t_1 < t_2 < \cdots < t_N$ be a sequence of arrival times, and let $\boldsymbol{z} = (z_1, \ldots, z_N)$ be a vector of binary indicator variables, such that $z_i = 1$ if the ith event was periodic, and $z_i = 0$ if it was non-periodic or human-generated. For each event time t_i, the following circular transformation can be defined

$$x_i = \frac{2\pi}{p}(t_i \bmod p). \quad (4)$$

This transformation is particularly suitable for sequences presenting *fixed phase polling* (Price-Williams et al. 2017): the event times are expected to occur every p seconds, with a zero-mean random error. Wrapping the sequence to $[0, 2\pi)$ also makes the methodology robust to *dropouts* in the observations. If the events occur p seconds after the preceding arrival time, plus error, then the sequence exhibits *fixed duration polling*, and a more appropriate transformation might be:

$$x_i = \frac{2\pi}{p}\{(t_i - t_{i-1}) \bmod p\},$$

with $x_1 = 0$. This article mostly concerns with fixed phase polling, but the methodology could be adapted to the case of fixed duration polling.

The aim of this article is to use the observed vector $\boldsymbol{x} = (x_1, \ldots, x_N)$ to estimate $\boldsymbol{z}$. For the ith event, the first measurement, x_i, will reveal whether it was synchronous with the polling found to occur on those arrival times.

In some applications, a second *known* periodic effect will be present, such as a daily or annual seasonality. Denote this second periodicity p', where typically $p \ll p'$. A second circular transformation can then be defined:

$$y_i = \frac{2\pi}{p'}(t_i \bmod p'). \tag{5}$$

Within the application in computer networks, it could be assumed $p' = 86{,}400$ s. Such measurement will show the time of day (which is 86,400 s long when there are no clock changes) at which the event occurred, and can be compared against an inferred diurnal model corresponding to human activity. More generally, one could be interested in the estimation of the density of the non-periodic events on the entire observation period, which yields the generic transformation $\tilde{t}_i = 2\pi t_i / T$.

In the next section, a mixture probability model for $\boldsymbol{x}$ is proposed, which can be used to classify events purely on their synchronicity with the polling signal. Then, in Sect. 5 the model is extended to incorporate $\boldsymbol{y}$, to see how much extra discriminative information can be extracted from the time of day. Note that the measurements (4) and (5) have both been scaled to lie on the unit circle with domain $[0, 2\pi)$. This consistency in scaling will be convenient for specifying the full probability model (22) for event times in Sect. 5, since this makes simultaneous use of both quantities.

4 A wrapped normal–uniform mixture model

If a sequence of arrival times is classified periodic with period p (3), then a majority of the wrapped values $\boldsymbol{x}$ from (4) will be concentrated around a peak. A wrapped normal distribution $\mathbb{WN}_{[0,2\pi)}(\mu, \sigma^2)$ model is therefore proposed for those events, where $\sigma > 0$ quantifies the variability of event times around the peak location $\mu \in [0, 2\pi)$. The density of $\mathbb{WN}_{[0,2\pi)}(\mu, \sigma^2)$ is

$$\phi_{\mathrm{WN}}^{[0,2\pi)}(x; \mu, \sigma^2) = \sum_{k=-\infty}^{\infty} \phi(x + 2\pi k; \mu, \sigma^2) \mathbb{1}_{[0,2\pi)}(x), \tag{6}$$

where $\phi(\cdot; \mu, \sigma^2)$ and later $\Phi\{\cdot; \mu, \sigma^2\}$ will represent, respectively, the density and distribution functions of the Gaussian distribution $\mathbb{N}(\mu, \sigma^2)$.

In practical applications, p will usually be relatively small; hence, it is reasonable to assume that the density of the non-periodic events is smooth and therefore locally well approximated by a uniform distribution on the unit circle. Together, these components imply a density for x_i conditional on the latent variable z_i,

$$f(x_i|z_i) = \phi_{\mathrm{WN}}^{[0,2\pi)}(x_i; \mu, \sigma^2)^{z_i} (2\pi)^{z_i - 1} \mathbb{1}_{[0,2\pi)}(x_i). \tag{7}$$

Let $\theta \in [0, 1]$ be the unknown proportion of events which are generated automatically and periodically, such that $\mathbb{P}(z_i = 1) = \theta$. Finally, let

$$\boldsymbol{\psi} = (\mu, \sigma^2, \theta)$$

be the three model parameters which have been introduced. Then, assuming the individual values of $\boldsymbol{x}$ are drawn independently of one another, the likelihood function of the three model parameters is

$$L(\boldsymbol{\psi}|\boldsymbol{x}) = \prod_{i=1}^{N} \left\{ \theta \phi_{\mathrm{WN}}^{[0,2\pi)}(x_i; \mu, \sigma^2) + \frac{1-\theta}{2\pi} \right\} \mathbb{1}_{[0,2\pi)}(x_i). \tag{8}$$

It is not analytically possible to optimise the likelihood in (8) directly; instead, an expectation–maximisation (EM) algorithm (Dempster et al. 1977), common for mixture models, is proposed in the next section.

4.1 An EM algorithm for parameter estimation

In order to develop an EM algorithm for estimating $\boldsymbol{\psi}$, it is necessary to introduce additional latent variables $\boldsymbol{\kappa} = (\kappa_1, \ldots, \kappa_N)$ for the mixture components in the wrapped normal model (6). For $1 \le i \le N$, if $z_i = 0$, then let $\kappa_i = 0$ with probability 1; for $z_i = 1$ and $k \in \mathbb{Z}$, let

$$\begin{aligned} &\mathbb{P}(\kappa_i = k | z_i = 1, \mu, \sigma^2) \\ &\quad = \Phi\{2\pi(k+1); \mu, \sigma^2\} - \Phi\{2\pi k; \mu, \sigma^2\}. \end{aligned} \tag{9}$$

Further, let

$$x_i | z_i = 1, \kappa_i = k, \mu, \sigma^2 \sim \tilde{\mathbb{N}}_{[0,2\pi)}(\mu - 2\pi k, \sigma^2),$$

denoting a normal distribution with mean $\mu - 2\pi k$ and variance σ^2, truncated to $[0, 2\pi)$. Then, the conditional density for x_i given z_i is again (7). The role of the latent variable κ_i is depicted in Fig. 2.

Using the latent assignments $\boldsymbol{z}$ and $\boldsymbol{\kappa}$, the revised likelihood function is

$$L(\boldsymbol{\psi}|\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\kappa}) \propto \prod_{i=1}^{N} \left(\frac{1-\theta}{2\pi} \right)^{1-z_i} \left\{ \theta \phi(x_i + 2\pi\kappa_i; \mu, \sigma^2) \right\}^{z_i}. \tag{10}$$

At iteration m of the EM algorithm, given an estimate $\boldsymbol{\psi}^{(m)}$ of $\boldsymbol{\psi}$, the $\mathbb{E}$-step computes the Q-function

$$Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(m)}) = \mathbb{E}_{\boldsymbol{z},\boldsymbol{\kappa}|\boldsymbol{x},\boldsymbol{\psi}^{(m)}} \left\{ \log L(\boldsymbol{\psi}|\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\kappa}) \right\}, \tag{11}$$

where the expectation is taken with respect to the conditional distribution of $\boldsymbol{z}$ and $\boldsymbol{\kappa}$, given $\boldsymbol{x}$ and $\boldsymbol{\psi}^{(m)}$. This amounts to evaluating the so-called *responsibilities*,

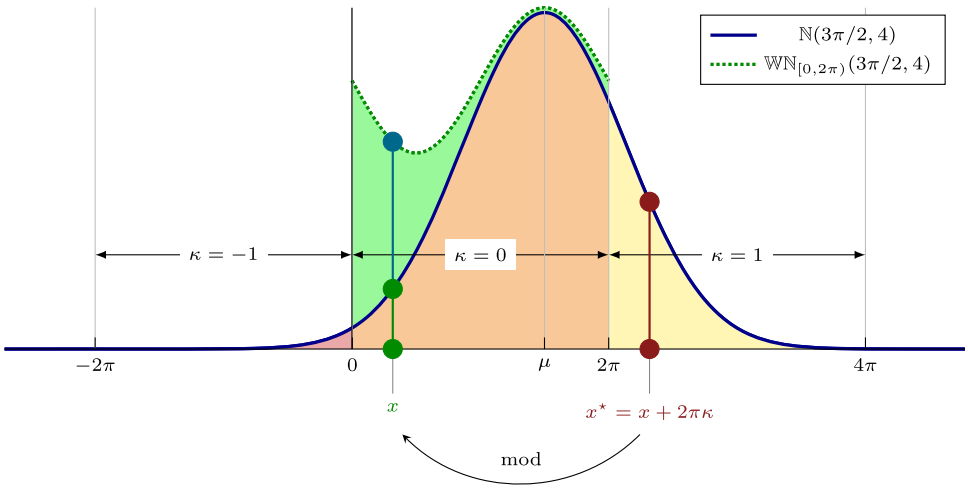


Fig. 2 Interpretation of the latent variable κ. Suppose $x^\star \sim \mathbb{N}(\mu, \sigma^2)$, $x = x^\star \bmod 2\pi$ and $\kappa = (x^\star - x)/(2\pi)$. Then, $x \sim \mathbb{WN}_{[0,2\pi)}(\mu, \sigma^2)$ and $\kappa = k$ with probability given by (9)

$$\begin{aligned}\zeta_{i(j,k)} &= \mathbb{E}_{\boldsymbol{z},\boldsymbol{\kappa}|\boldsymbol{x},\boldsymbol{\psi}^{(m)}}[\mathbb{1}_{(j,k)}(z_i, \kappa_i)|x_i, \boldsymbol{\psi}^{(m)}] \\ &= \mathbb{P}(z_i = j, \kappa_i = k|x_i, \boldsymbol{\psi}^{(m)}),\end{aligned} \tag{12}$$

since, using (10), the Q-function (11) then simplifies to

$$\sum_{i=1}^{N}\left[\zeta_{i(0,0)} \log\left(\frac{1-\theta_{(m)}}{2\pi}\right) + \sum_{k=-\infty}^{\infty} \zeta_{i(1,k)} \log\left\{\theta\phi(x_i; \mu_{(m)} - 2\pi k, \sigma^2_{(m)})\right\}\right]. \tag{13}$$

The responsibilities in (12) can be calculated using Bayes theorem, giving

$$\zeta_{i(j,k)} \propto \{\theta_{(m)}\phi(x_i; \mu_{(m)} - 2\pi k, \sigma^2_{(m)})\}^j \left(\frac{1-\theta_{(m)}}{2\pi}\right)^{1-j}, \tag{14}$$

where the normalising constant is given by the sum $\theta_{(m)} \sum_{k'=-\infty}^{\infty} \phi(x_i; \mu_{(m)} - 2\pi k', \sigma^2_{(m)}) + (1 - \theta_{(m)})/2\pi$. Finally, maximising (13) with respect to $\boldsymbol{\psi}$ as the $\mathbb{M}$-step gives:

$$\begin{aligned}\tilde{\mu}_{(m+1)} &= \frac{\sum_{i=1}^{N}\sum_{k=-\infty}^{\infty}(x_i + 2\pi k)\zeta_{i(1,k)}}{\sum_{i=1}^{N}\sum_{k=-\infty}^{\infty}\zeta_{i(1,k)}}, \\ \mu_{(m+1)} &= \tilde{\mu}_{(m+1)} \bmod 2\pi, \\ \sigma^2_{(m+1)} &= \frac{\sum_{i=1}^{N}\sum_{k=-\infty}^{\infty}(x_i + 2\pi k - \tilde{\mu}_{(m+1)})^2\zeta_{i(1,k)}}{\sum_{i=1}^{N}\sum_{k=-\infty}^{\infty}\zeta_{i(1,k)}}, \\ \theta_{(m+1)} &= \frac{1}{N}\sum_{i=1}^{N}\sum_{k=-\infty}^{\infty}\zeta_{i(1,k)} = 1 - \frac{1}{N}\sum_{i=1}^{N}\zeta_{i(0,0)}.\end{aligned} \tag{15}$$

In practical computations, the infinite sums must be truncated to a suitable level.

4.2 A Bayesian formulation

Data augmentation (Higdon 1998) can be used to construct an analogue of the EM algorithm in a Bayesian setting, with a Gibbs sampler for the latent variables $\boldsymbol{z}$ and $\boldsymbol{\kappa}$. A convenient choice of prior distribution assumes a factorisation $p(\boldsymbol{\psi}) = p(\mu, \sigma^2)p(\theta)$, where $\theta \sim \text{Beta}(\gamma_0, \delta_0)$ and $(\mu, \sigma^2) \sim \text{NIG}(\mu_0, \lambda_0, \alpha_0, \beta_0)$ and NIG denotes the normal-inverse gamma distribution. The chosen prior distributions are conjugate for the likelihood and therefore allow the inferential process to be analytically tractable (see Bernardo and Smith 1994, for more details). The prior and posterior probabilities for the latent assignments, conditional on $\boldsymbol{\psi}$, are the same as (9) and (14), respectively. Conditional on $\boldsymbol{z}$, the posterior distribution for the mixing proportion is

$$\theta|\boldsymbol{z} \sim \text{Beta}(\gamma_0 + N_1, \delta_0 + N_0),$$

where $N_1 = \sum_{i=1}^{N} z_i$ is the number of automated events and $N_0 = N - N_1$ is the number of human and non-periodic automated events. The conditional posterior distribution of μ and σ^2 is $\text{NIG}(\mu_{N_1}, \lambda_{N_1}, \alpha_{N_1}, \beta_{N_1})$, where

$$\begin{aligned}\tilde{x} &= \sum_{i:z_i=1}(x_i + 2\pi\kappa_i)/N_1, \\ \mu_{N_1} &= \frac{\lambda_0\mu_0 + N_1\tilde{x}}{\lambda_0 + N_1}, \\ \lambda_{N_1} &= \lambda_0 + N_1, \\ \alpha_{N_1} &= \alpha_0 + N_1/2,\end{aligned} \tag{16}$$

$$\beta_{N_1} = \beta_0 + \frac{1}{2}\left\{\sum_{i:z_i=1}(x_i + 2\pi\kappa_i - \tilde{x})^2 + \frac{\lambda_0 N_1}{\lambda_{N_1}}(\tilde{x} - \mu_0)^2\right\}. \tag{17}$$

Similar to the case of (15), samples $\mu = \tilde{\mu} \bmod 2\pi$ from the posterior for (μ, σ^2) should be used, where $\tilde{\mu}$ is sampled from $\text{NIG}(\mu_{N_1}, \lambda_{N_1}, \alpha_{N_1}, \beta_{N_1})$.

5 Incorporating time of day

The model presented in Sect. 4 only made use of $\boldsymbol{x}$, the arrival times once wrapped onto the unit circle according to the estimated periodicity p (4); recall that these values reveal the synchronicity of each event with the automated polling signal. However, further information might potentially be obtained from $\boldsymbol{y}$ (5), the times of day at which each event occurred. In computer networks, this is a reasonable assumption, since any human-generated events should be subjected to some level of diurnality. This section introduces a model for the daily distribution of human connections to help extract this extra information. Following Heard and Turcotte (2014), a flexible model for the distribution of arrivals of human events through a typical day will be obtained by assuming the density to be a step function with $\ell \geq 1$ segments, written

$$s(y; \ell, \boldsymbol{\tau}, \boldsymbol{h}) = \frac{\mathbb{1}_{[0,\tau_1)\cup[\tau_\ell,2\pi)}(y)\, h_\ell}{2\pi - \tau_\ell + \tau_1} + \sum_{j=1}^{\ell-1} \frac{\mathbb{1}_{[\tau_j,\tau_{j+1})}(y)\, h_j}{\tau_{j+1} - \tau_j}. \tag{18}$$

The segment probabilities $\boldsymbol{h} = (h_1, \ldots, h_\ell) \in [0,1]^\ell$ satisfy $\sum_{j=1}^{\ell} h_j = 1$, and the circular changepoints $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_\ell)$, $0 \leq \tau_1 < \ldots < \tau_\ell < 2\pi$ determine the step positions.

The number of segments ℓ is treated as unknown and assigned a geometric prior with parameter $\nu \in (0,1)$ and mass function $\nu(1-\nu)^{\ell-1}$. The natural prior for $\boldsymbol{h}|\boldsymbol{\tau}, \ell$ (Bernardo and Smith 1994) is

$$\begin{aligned}&\text{Dirichlet}[\eta\Lambda\{(\tau_1,\tau_2)\}, \ldots, \eta\Lambda\{(\tau_{\ell-1},\tau_\ell)\},\\ &\quad \eta\Lambda\{(\tau_\ell, 2\pi)\cup(0,\tau_1)\}],\end{aligned} \tag{19}$$

where $\eta > 0$ is a concentration parameter and $\Lambda\{\cdot\}$ is here taken to be the Lebesgue measure. The hierarchical specification of the model is completed with an uninformative prior on the segment locations: they are assumed to be the order statistics of ℓ draws from the uniform distribution on $[0, 2\pi)$.

Given ℓ segments defined by changepoints $\boldsymbol{\tau}$, the Dirichlet probabilities $\boldsymbol{h}$ can be integrated out to yield the marginal likelihood of observing daily arrival times $\boldsymbol{y}$, which is given by

$$\frac{c(N)\Gamma\{N'_\ell + \eta(2\pi - \tau_\ell + \tau_1)\}}{\Gamma\{\eta(2\pi - \tau_\ell + \tau_1)\}(2\pi - \tau_\ell + \tau_1)^{N'_\ell}} \prod_{j=1}^{\ell-1} \frac{\Gamma\{N'_j + \eta(\tau_{j+1} - \tau_j)\}}{\Gamma\{\eta(\tau_{j+1} - \tau_j)\}(\tau_{j+1} - \tau_j)^{N'_j}}, \tag{20}$$

where $N'_j = \sum_{i=1}^{N} \mathbb{1}_{[\tau_j,\tau_{j+1})}(y_i)$ is the number of observations in the jth segment, $1 \leq j \leq \ell - 1$, $N'_\ell = N - \sum_{j=1}^{\ell-1} N'_j$ and $c(N) = \Gamma(2\pi\eta)/\Gamma(2\pi\eta + N)$ is a normalising constant.

In contrast to the human events, automated periodic events are generated regularly by the underlying polling mechanism, which is likely to be irrespective of the time of day. Recall from Sect. 2 that the binary indicator variable z_i is defined to be equal to 1 if the ith event was periodic, and 0 otherwise. The approach which will now be adopted is to model the conditional density for the unwrapped event time t_i, depending on the value of z_i.

For simplicity of presentation, it will be assumed that the length of the observation period, T, will be both a whole number of days and an integer multiple of p. Under this assumption

$$\begin{aligned} f(t_i|z_i) &= \frac{2\pi}{T} f(x_i|z_i = 1)^{z_i} f(y_i|z_i = 0)^{1-z_i} \\ &= \frac{2\pi}{T} \phi_{\text{WN}}^{[0,2\pi)}(x_i; \mu, \sigma^2)^{z_i} s(y_i; \ell, \boldsymbol{\tau}, \boldsymbol{h})^{1-z_i}, \end{aligned} \tag{21}$$

implying the marginal distribution mixture density

$$f(t_i) = \frac{2\pi}{T}\left\{\theta\phi_{\text{WN}}^{[0,2\pi)}(x_i; \mu, \sigma^2) + (1-\theta)s(y_i; \ell, \boldsymbol{\tau}, \boldsymbol{h})\right\}. \tag{22}$$

Figure 3 provides a graphical summary of the full model (22), and Fig. 4 shows an illustrative example of the mixture density. Note that relaxing the assumptions of divisibility of T by p or p' simply requires straightforward calculation of corresponding normalising constants in (21), and this adjustment will be negligible when $\lfloor T\delta/p \rfloor$ is large.

Since most of the prior distributions have been chosen to be conjugate, it is possible to explicitly integrate out the segment heights $\boldsymbol{h}$, see (20), and the mixing proportion θ, leading to a collapsed Gibbs sampler (Liu 1994) for inference. This is advantageous, since it reduces the simulation effort to sampling the latent variables $\boldsymbol{z}$ and $\boldsymbol{\kappa}$, the parameters μ and σ^2 for the wrapped normal component (6), and the number of circular changepoints ℓ and their locations $\boldsymbol{\tau}$ in the human event density (18). The algorithm is described in detail in Appendix A.

In principle, the model could potentially be further extended. An even more general framework for density estimation in a Bayesian setting is the Dirichlet process mixture

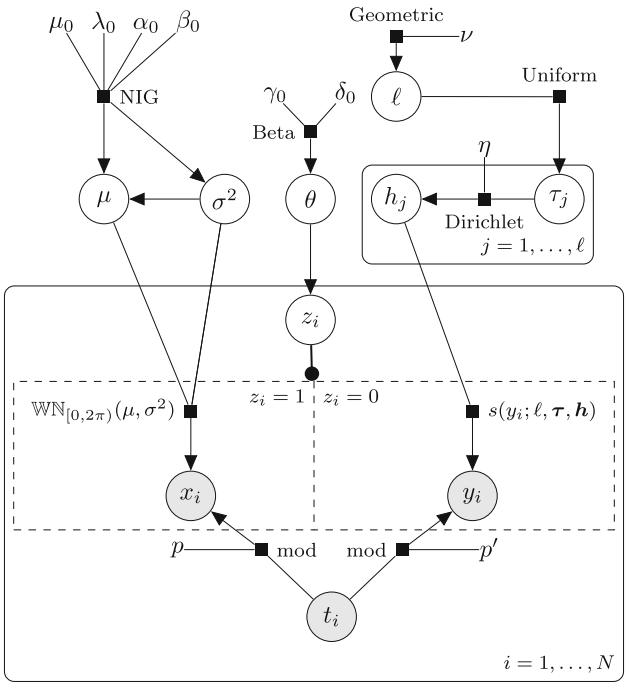


Fig. 3 Graphical representation of the extended Bayesian mixture model for separation of human and automated activity

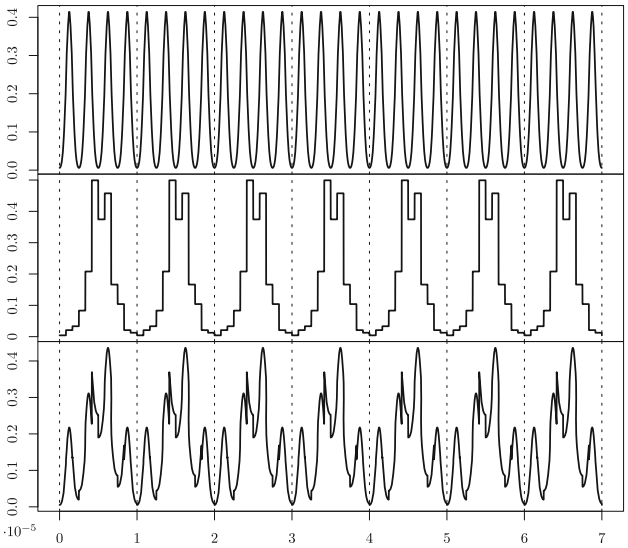


Fig. 4 Example of the component densities in (22), with $T = 7 \times 86{,}400$ (7 days), $p = 21{,}600$ (6 h), $\mu = \pi$, $\sigma^2 = 1$, $\theta = 0.5$, $\ell = 12$, equally spaced segment locations, and step function heights $\boldsymbol{h}$ chosen to resemble a human daily distribution of arrival times. Upper panel: density of automated events (4 peaks per day are recorded since $p =$ 6 h). Middle: density of human events (daily distribution repeated each day). Lower: the resulting mixture density (22)

(Escobar and West 1995). Inference in this case is cumbersome, but algorithms exist for relatively fast implementation (Neal 2000).

6 Applications

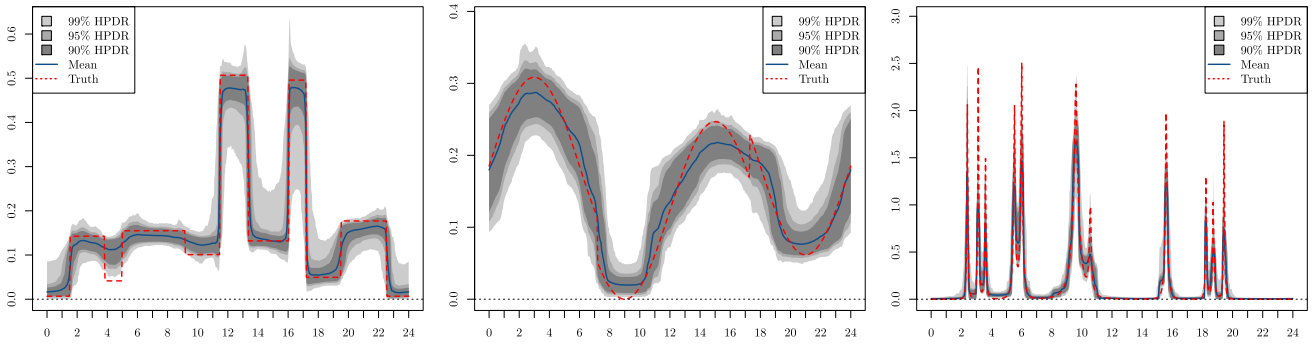
The algorithms described in the previous sections have been applied on computer network flow data collected at Imperial College London, for a single client IP address X, setting $p' = 86{,}400$. In order to show the efficacy of the methods for filtering polling traffic, examples are presented using simulated data, a synthetically fused mixture and some raw network flow data.

6.1 Simulated data

The performance of the Gibbs sampler in recovering the correct densities for the model in Sect. 5 is first assessed on simulated data. Non-periodic events were simulated from a range of densities of increasing complexity, inspired by the test signals in Donoho and Johnstone (1994), rescaled and shifted to represent probability distributions on $[0, 2\pi)$. Three distributions are used: *(a)* a step function density with 10 segments, where the changepoints and segment probabilities were sampled from a Uniform$[0, 2\pi)$ and Dirichlet$(1, \ldots, 1)$, respectively, *(b)* a heavisine function on $[0, 2\pi)$, $f(y) \propto 6 + 4\sin(2y) - \mathrm{sgn}(y/2\pi - 0.3) - \mathrm{sgn}(0.72 - y/2\pi)$, *(c)* a function $f(y) \propto \sum_{j=1}^{11} u_j(1 + |(y/2\pi - v_j)/w_j|)^{-4}$ with 11 bumps, with the same choices of Donoho and Johnstone (1994) for the parameters u_j, v_j and w_j, scaled to $[0, 2\pi)$. 3000 events are simulated from the chosen distributions and then assigned to a random day of the week, implying $p' = 86{,}400$. Those events are mixed with 2000 periodic events generated from a wrapped normal distribution with mean $\mu = 5$ and variance $\sigma^2 = 1$ on $[0, 2\pi)$, rescaled and assigned at random to windows of $p = 10$ s over one week. Note that the variance of the periodic signal is chosen to be relatively large to make the inferential procedure more complicated. In practical applications, the value of σ^2 is expected to be much smaller.

The results of the Gibbs sampling procedure for estimation of the density of non-polling events, using the model in Sect. 5, are reported in Fig. 5. The algorithm is able to recover the density with good confidence, even in case of departures from the step function assumption. Note that it is not possible to expect the fit of the estimated density to correspond perfectly to the density used to simulate the data, since the simulation is repeated only once, for a sample of size 3000, and the variability for the wrapped normal component was chosen to be large.

The estimates for the remaining parameters in the simulation using the step function density resulted in $(\hat{\mu}, \hat{\sigma}^2, \hat{\theta}) = (5.0162, 0.9890, 0.4022)$. The performance of the classification algorithm can be assessed using the area under the receiver operating characteristic (ROC) curve, commonly denoted as AUC. For the step function density, the resulting AUC score is 0.8161. For the heavisine function, the esti-



(a) Step function with 10 changepoints **(b)** Heavisine function **(c)** Function with 11 bumps

Fig. 5 Estimated daily density of non-polling events for the three simulated examples described in Sect. 6.1

Table 1 Elapsed time (s) for 1000 sweeps of the collapsed Gibbs sampler for the model in Sect. 5, as a function of the number of observations N

N	100	1000	5000	10, 000	25, 000	50, 000
Time	1.82	10.68	52.01	122.58	292.88	729.04

mates of the parameters are (5.0268, 0.9868, 0.3882), and AUC = 0.8007. Finally, for the function with bumps, the estimates are (5.0162, 0.9890, 0.4022), and AUC = 0.9337. The parameter estimates correspond to the values used in the simulation, and the AUC values are acceptable considering the complexity of the simulation and the fact that $\sigma^2 = 1$, much larger than the values expected in applications.

The computational efficiency of the collapsed Gibbs sampler for the model in Sect. 5 has been evaluated via simulation. Table 1 reports the elapsed time for 1000 sweeps of the sampler for different values of N, the number of observed arrival times. The experiments were performed running *python* code on a MacBook Pro 2017 with a 2.3 GHz Intel Core i5 dual-core processor, and the events were generated using the same simulation described in this section, with a step function density with 10 changepoints for the non-polling events.

6.2 Synthetically labelled data: a mixture of automated and human connections

A fusion of two different network edges is considered: first, the activity between the client X and the Dropbox server 108.160.162.98, found to be strongly periodic at period $p \approx 55.66$ s, with associated p-value < 0.0001; and second, the activity between the client X and the Midasplayer server addresses 217.212.243.163 and 217.212.243.186, which exhibits activity exclusively during day, relating to a human user playing the popular online game Candy Crush. Seven days of data starting from the first observation time on each edge were used in the present analysis, resulting in 32, 865 Dropbox events and 4779 Candy Crush connections. The histograms of daily activity for the two edges are presented in Fig. 6. Notice that Dropbox is slightly more active at night than during the day, which makes the analysis more difficult. This is not an uncommon behaviour for automated edges, which tend to 'stand down' during the day when a human sits at the machine. On the other hand, Candy Crush events only happen during working hours.

The uniform-wrapped normal mixture model (cf. Sect. 4) fitted to the fused data using the EM algorithm quickly converges to the parameter estimates $(\hat{\mu}, \hat{\sigma}^2, \hat{\theta}) = (4.3376, 0.4059, 0.8585)$. The same results are obtained using different initialisation points and comparing different convergence criteria. Given the output of the EM algorithm, it is possible to filter the connections, keeping those such that $\zeta_{i(0,0)} > \sum_{k=-\infty}^{\infty} \zeta_{i(1,k)}$ at the final iteration, where the infinite sum is in practice truncated to a suitable level. These filtered events are those which would be assigned to the uniform (non-periodic) component of the mixture in (7). In total, 2818 wrapped times were classified as non-periodic, and 2386 of these are connections to Candy Crush servers, resulting in a false positive rate FPR = 0.013 and false negative rate FNR = 0.501. Note that it is not surprising that approximately 50% of the Candy Crush edges are missed, because these fall into the high density area of the wrapped normal by chance, being approximately uniform on the p-clock.

The results from the EM algorithm were then compared to the inferences obtained from the posterior distribution of the parameters $\boldsymbol{\psi}$ using the Bayesian algorithm of Sect. 4.2. The prior parameters were set to the uninformative values $\mu_0 = \pi, \lambda_0 = 1, \alpha_0 = \beta_0 = \gamma_0 = \delta_0 = 1$, although given the large quantity of data available, the choice of the prior is in practice not influential on the results of the procedure. The resulting mean of the posterior distribution for $\boldsymbol{\psi}$ is $\hat{\boldsymbol{\psi}} = (\hat{\mu}, \hat{\sigma}^2, \hat{\theta}) = (4.3375, 0.4064, 0.8583)$, almost identical to the result obtained using the EM algorithm. This is expected, since the two methods represent two different inferential approaches for the same model. Very similar

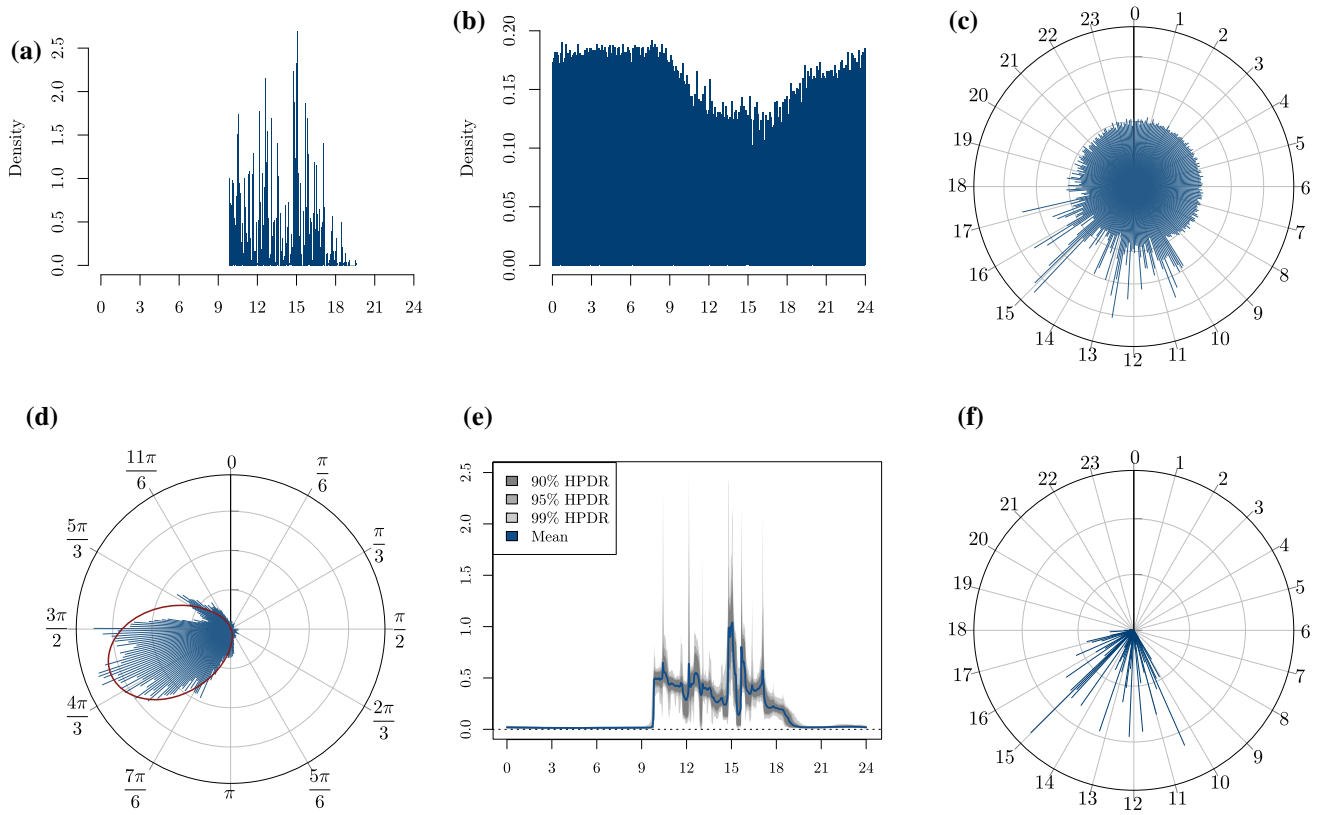


Fig. 6 Analysis on the synthetic Dropbox–Candy Crush data set. Top panel: histogram of the daily arrival times y_i of the Candy Crush (left) and Dropbox (middle) events (bin size: 5 min), and polar histogram of the daily arrival times for the mixed data (right). Bottom panel: polar histogram of the wrapped arrival times x_i with period $p = 55.66$ for the filtered periodic events and estimated wrapped normal density (left), estimated daily density of non-periodic events (middle) and histogram of the daily arrival times for the filtered non-periodic events (right)

results are also obtained when filtering the data. For event i, let $\hat{z}_i$ be the Monte Carlo estimate of z_i; then, classifying events as non-periodic if $\hat{z}_i < 0.5$ yields 2810 events, and 2377 of those are Candy Crush connections, corresponding to FPR $= 0.013$ and FNR $= 0.502$. In practice, for the uniform-wrapped normal model, it is recommended to use the EM algorithm, which converges faster than the Bayesian Markov chain Monte Carlo (MCMC) procedure, providing, as expected, equivalent results.

Finally, it is of interest to see whether the classification performance can be improved using the extended model presented in Sect. 5. The algorithm was initialised from the output of the EM algorithm, and the additional parameters were set to the uninformative values $\nu = 0.1$ and $\eta = 1$, although again the algorithm is robust to different starting points. The resulting posterior mean estimates of wrapped normal distribution parameters are $(\hat{\mu}, \hat{\sigma}^2) = (4.362, 0.376)$ and $\hat{\theta} = 0.8506$ for the mixing proportion, which are slightly different from the previous analysis; in particular, the variance is lower. The estimated daily distribution of the non-periodic arrival times is plotted in Fig. 6. Note that its mean almost perfectly reproduces the histogram of the daily arrival times of the Candy Crush events. The estimated density has been obtained by sampling from the posterior distribution $h|\boldsymbol{\tau}, \ell, \boldsymbol{y}, \boldsymbol{z}$ for each iteration of the Gibbs sampler, which has known form under the conjugate prior (19), and then averaging the density across the iterations. In this case, 3947 filtered events are labelled as non-periodic ($\hat{z}_i < 0.5$), with 2948 true positives, corresponding to FPR $= 0.030$ and FNR $= 0.383$. The resulting histogram of the filtered data is plotted in Fig. 6. The posterior distribution for the number of changepoints in the human density is approximately normally distributed around the value $\ell = 28$, which roughly corresponds to one changepoint per hour of the day.

The algorithms proposed in the article can be more efficiently compared for classification purposes using a ROC curve for different values of the threshold for $\hat{z}_i$. The plot for this example is reported in Fig. 7, and it clearly shows that the proposed methodologies correctly classify a significant proportion of the events very well, with low false positive rates for the threshold 0.5. Furthermore, including the daily arrival times $\boldsymbol{y}$ in the model is clearly beneficial. For practical applications, it is recommended to choose a threshold

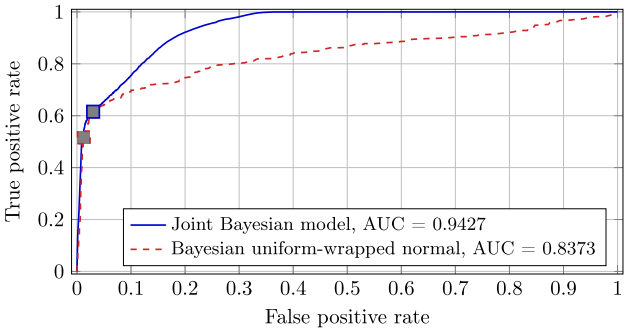


Fig. 7 ROC curves and AUC values evaluating the performance of two methods for classification of human events: Bayesian uniform-wrapped normal mixture (Sect. 4.2) and joint Bayesian model (Sect. 5). The grey squares correspond to the threshold 0.5

that guarantees low false positive rates for detection of human events: in this example, 0.5 seems an appropriate choice.

6.3 Real data: Imperial College NetFlow

In this example, the activity between a client Y and the server IP 13.107.42.11, used by the software *Outlook*, is analysed. The arrival times refer to a time period between August 2017 and November 2017, and 7 days of activity after the first observation were considered. The daily distribution of the activity on the edge is reported in Fig. 8. A total number of 7583 connections were recorded. It can be observed from the histogram that the activity on the edge is almost entirely automatic, but the number of connections slightly increases during working hours compared to the night (compared with the dip observed in other automated services like Dropbox). This suggests a mixture between human activity and polling behaviour on this edge, which is further supported by the nature of the software. The arrival times on the edge have been found to be strongly periodic at period $p \approx 8$ s, with an associated g-test p-value $< 10^{-7}$.

The uniform-wrapped normal mixture model (cf. Sect. 4), with period 8 s, fitted using the EM algorithm converges to the parameter estimate $(\hat{\mu}, \hat{\sigma}^2, \hat{\theta}) = (1.872, 0.670, 0.714)$. In this case study, 1246 of the 7583 events were assigned to the uniform (non-periodic) category using the criterion $\zeta_{i(0,0)} > 0.5$. Identical parameter estimates are obtained using the Bayesian mixture model, and classification of the connections as periodic or non-periodic is again almost the same, with 1232 connections classified as non-periodic from the model. Furthermore, most of the activity in the filtered events is concentrated in working hours, even though this is not explicitly encouraged by this model. This is promising since the algorithm has been able to recover human-like activity from an edge that apparently seems almost entirely automated.

Next, the Gibbs sampler was used to infer the parameters of the joint Bayesian model (cf. Sect. 5). The same prior parameter values as the previous section were used. The convergence of the sampler to the correct target is again almost immediate. The number of non-periodic connections was estimated as 1430. The resulting posterior mean for the parameters of the wrapped normal distribution for the polling component is $(\hat{\mu}, \hat{\sigma}^2) = (1.885, 0.6249)$ and $\hat{\theta} = 0.6935$ for the mixing proportion. The daily distribution of the non-periodic connections is reported in Fig. 8 and displays a strong diurnal pattern, suggesting human behaviour has been classified well.

However, it is also evident that in this example, the algorithm classifies as human a proportion of connections occurring during the night. Potential issues that can arise are multiple periodicities or phase shifts within the same data stream. A possible solution would be to iteratively repeat the analysis on the filtered non-periodic events until no significant short-term periodicities are obtained using the g-test. In this example, repeating the analysis with period 8 s allows the residual automated activity to be filtered out, thereby obtaining an estimated daily distribution which is entirely consistent with human-like behaviour, shown in Fig. 8e. After this last stage of the analysis, only 181 events are retained as human-generated, corresponding to $\approx 2.5\%$ of the initial 7583 events. This proportion is consistent with results obtained in previous studies on computer network data (Price-Williams et al. 2017).

The performance of the algorithm for filtering polling activity can also be assessed by comparing the model fit to both the filtered and unfiltered event streams when applying the nonparametric Wold process model of Price-Williams and Heard (2020), which has been shown to be suitable for human-like events in computer network traffic. There, a counting process of human-generated events is modelled with a conditional intensity represented as a step function with an inferred number of changepoints. If $y_1, y_2, \ldots$ are the event times of such a counting process $Y(t)$, the conditional intensity has the form

$$\lambda_Y(t) = \lambda + \sum_{j=1}^{\ell} \lambda_j \mathbb{I}_{[\tau_{j-1},\tau_j)}(t - y_{Y(t)}), \tag{23}$$

where $0 \equiv \tau_0 < \tau_1 < \ldots < \tau_\ell$ are a finite sequence of changepoints and $\lambda_1 > \ldots > \lambda_\ell$ are a decreasing sequence of corresponding step heights, representing the fall in intensity experienced as the waiting time increases between the current time t and the most recent event $y_{Y(t)}$. In contrast, periodic network events are not self-exciting; their conditional intensity would decrease immediately after an event, and only increase when the next periodic signal is anticipated.

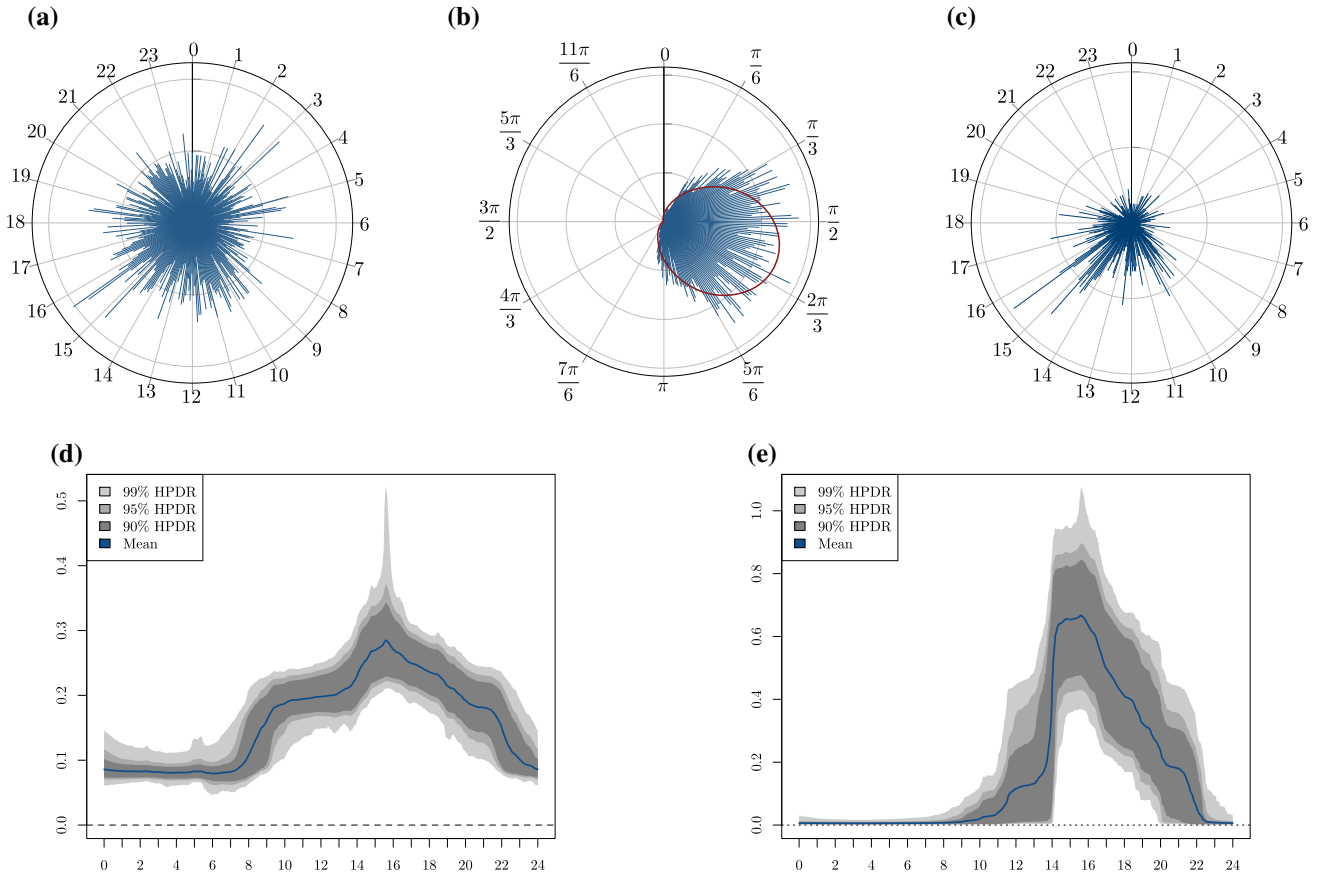


Fig. 8 Analysis on the edge $Y \rightarrow$ 13.107.42.11 (*Outlook*). Top: polar histograms of the daily arrival times y_i of the events (left), and of the wrapped arrival times x_i for periodicity $p \approx 8$ s with fitted wrapped normal density (middle) and finally of the filtered non-periodic events (right). Bottom: resulting estimated daily density of non-periodic events (left) from applying the algorithm once, and then the estimated daily density of human events (right) obtained from re-applying the algorithm with the same periodicity on the filtered events from plot (c)

Price-Williams and Heard (2020) used predictive p-values to assess model fit of the intensity model (23); defining $y_0 \equiv 0$ and the compensator function $\Lambda(t) = \int_{s=0}^{t} \lambda_Y(s)ds$, a lower-tail p-value of the ith waiting time is

$$p_i = 1 - \exp\left[-\{\Lambda(y_i) - \Lambda(y_{i-1})\}\right]. \tag{24}$$

Figure 9 reports the Q–Q plot of the distribution of predictive p-values (24) obtained using the first 4 days of observations as training data and the remaining days as test data, for both the unfiltered and filtered non-periodic events from Fig. 8c. The distribution of the p-values clearly improves when the filtered non-periodic events are used. The Kolmogorov–Smirnov (KS) score, based on the maximum absolute difference between the empirical and theoretical CDFs, significantly decreases for the filtered events, reaching a value which is consistent with the results obtained by Price-Williams and Heard (2020) on Imperial College NetFlow data.

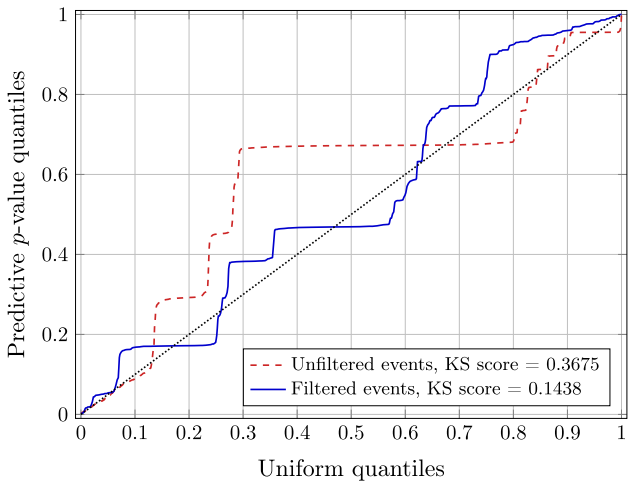


Fig. 9 Uniform Q–Q plots of predictive p-values on the unfiltered and filtered non-periodic events, obtained using the nonparametric Wold process model of Price-Williams and Heard (2020), and corresponding Kolmogorov–Smirnov scores

This example strikingly shows characteristics of real computer network traffic data: the activity on automatic edges only slightly increases during the day due to the presence of a human at the machine. Despite these difficulties, the algorithm was successfully able to derive a reasonable distribution for the human events.

7 Conclusion

In this article, a statistical framework for classification of arrival times in event time data has been proposed. The methodology was motivated by application to computer network modelling for cyber-security. In particular, the filtering methodology developed in Heard et al. (2014) has been extended to network edges that present a mixture of human and automated polling activity, in order to prevent the loss of information caused by totally removing a seemingly automated edge from the analysis. This has initially been achieved using a simple mixture model based on a uniform distribution and a wrapped normal distribution on the unit circle. Frequentist and Bayesian algorithms for the estimation of the parameters have been presented. The model has then been extended to include available information on the daily arrival times of the events, demonstrating significant performance improvements on synthetic data sets with known labels. Bayesian inference is straightforward since simple conjugate distributions are used, and therefore, minimal adaptation is required from the user. Synthetically fused and real data examples show that the model is able to successfully recover a significant amount of the non-periodic activity and its distribution.

After fitting the model, the estimated values of the parameters can be used for instantaneous estimation of $z_{i'}$ for classification of future arrival times $t_{i'}$. Depending on the application, it might be necessary to update the parameter estimates from time to time as more data become available. The Bayesian framework naturally allows for prior-posterior updates, where the estimated posterior parameters can be used as prior hyperparameters when new data are available (Bernardo and Smith 1994). In that case, it would be necessary to perform the inferential procedure again, including the newly observed arrival times, and possibly removing a subset of the old observations to both fix the overall computational cost of the inferential procedure, which otherwise grows in N as shown in Table 1, and allow for any adaptation in the model.

The methodology proposed in this article generically fits within the literature on Bayesian model-based clustering (see Lau and Green 2007, for example), where MCMC methods are commonly used for inference on the latent allocations and model parameters (West et al. 1994; Richardson and Green 1997, for example). The proposed model complements and extends this literature, providing a Bayesian framework for classification of event time data, when a mixture of periodic and non-periodic events is observed.

Further possible extensions of the model could allow explicit accounting for phase shifts, using mixtures of wrapped normals with shared variances for the automated component, or allowing for changepoints in the mean μ of the wrapped normal distribution, accounting for the arrival order of each x_i. Furthermore, the case of multiple periodicities could be considered, using tests for multiple polling frequencies, for example Siegel (1980), yielding periodicities $p_1, \ldots, p_K$, and obtaining a mixture with multiple transformations $x_{ik} = (t_i \bmod p_k) \times 2\pi/p_k,\ k = 1, \ldots, K$. The model could also be adapted to allow for fixed duration polling, and alternative distributions could also be considered for the automated component, for example the wrapped Laplace distribution.

Within the application of computer network security, improvements might be achieved by including host specific information; unified data sets of this type have recently become available (Turcotte et al. 2018). Finally, the algorithm can be applied independently on multiple computer network edges, or, in principle, the same human density could be fitted for all edges emanating from the same source node, but allowing for different periodicities for traffic on each edge.

Supplementary material

The *python* code and datasets used in this article are publicly available in the repository https://github.com/fraspass/human_activity.

A Bayesian inference

This section describes the collapsed Gibbs sampler used for Bayesian inference on the model in Sect. 5.

Note that the full conditional distribution for the latent variable pair (z_i, κ_i) for the ith event factorises as

$$\begin{aligned}\mathbb{P}\{(z_i,\kappa_i)=(z,k)|\boldsymbol{z}_{-i},\boldsymbol{\kappa}_{-i},\boldsymbol{t},\mu,\sigma^2,\ell,\boldsymbol{\tau}\}&=\\ =\mathbb{P}(\kappa_i=k|z_i=z,t_i,\mu,\sigma^2)&\\ \mathbb{P}(z_i=z|\boldsymbol{z}_{-i},\boldsymbol{t},\mu,\sigma^2,\ell,\boldsymbol{\tau}),&\end{aligned}\tag{25}$$

where the subscript $-i$ on a vector denotes the same vector with the ith element removed. The first term on the right-hand side of (25) is

$$\mathbb{P}(\kappa_i=k|z_i,t_i,\mu,\sigma^2)\propto\{\phi(x_i+2\pi k;\mu,\sigma^2)\}^{z_i}\{\mathbb{1}_0(k)\}^{1-z_i}.$$

For the second term, it is easily seen that

$$\begin{aligned}&\mathbb{P}(z_i=z|\boldsymbol{z}_{-i},\boldsymbol{t},\mu,\sigma^2,\ell,\boldsymbol{\tau})\\ &\quad\propto\mathbb{P}(z_i=z|\boldsymbol{z}_{-i})f(t_i|\boldsymbol{t}_{-i},\boldsymbol{z},\mu,\sigma^2,\ell,\boldsymbol{\tau}).\end{aligned}\tag{26}$$

The first term on the right-hand side of (26) can be rewritten as the marginalised prior probability ratio $\mathbb{P}(\boldsymbol{z})/\mathbb{P}(\boldsymbol{z}_{-i})$. Note that:

$$\mathbb{P}(\boldsymbol{z})=\frac{\Gamma(N_1+\gamma_0)\Gamma(N_0+\delta_0)}{\Gamma(N+\gamma_0+\delta_0)}\frac{\Gamma(\gamma_0+\delta_0)}{\Gamma(\gamma_0)\Gamma(\delta_0)}.$$

Hence, letting $N_1^{-i}=\sum_{i'\neq i}z_{i'}$ be the number of classified periodic events excluding the ith event,

$$\mathbb{P}(z_i=1|\boldsymbol{z}_{-i})=\frac{N_1^{-i}+\gamma_0}{N-1+\gamma_0+\delta_0},$$

and $\mathbb{P}(z_i=0|\boldsymbol{z}_{-i})=1-\mathbb{P}(z_i=1|\boldsymbol{z}_{-i})$. For the second term of (26), for $z_i=1$, $f(t_i|z_i=1,\mu,\sigma^2)\propto\phi_{\mathrm{WN}}^{[0,2\pi)}(x_i;\mu,\sigma^2)$. For $z_i=0$, from (20) it follows that

$$\begin{aligned}&f(t_i|\boldsymbol{t}_{-i},z_i=0,\boldsymbol{z}_{-i},\ell,\boldsymbol{\tau})\propto f(y_i|\boldsymbol{y}_{-i},z_i=0,\boldsymbol{z}_{-i},\ell,\boldsymbol{\tau})\\ &=\frac{\sum_{i'\neq i}\mathbb{1}_0(z_{i'})\mathbb{1}_{[\tau_{j^\star},\tau_{j^\star+1})}(y_{i'})+\eta(\tau_{j^\star+1}-\tau_{j^\star})}{(N_0^{-i}+2\pi\eta)(\tau_{j^\star+1}-\tau_{j^\star})},\end{aligned}\tag{27}$$

where $j^\star\in\{1,\ldots,\ell-1\}$ is the segment $[\tau_{j^\star},\tau_{j^\star+1})$ containing y_i. If $j^\star=\ell$, then $\tau_{j^\star+1}-\tau_{j^\star}$ in (27) is substituted by $\Lambda\{(0,\tau_1)\cup(\tau_\ell,2\pi)\}=2\pi-\tau_\ell+\tau_1$.

For inference on μ and σ^2 conditional on the samples for which $z_i=1$, the results in (16) and (17) still apply. Furthermore, inference for the number and location of circular changepoints is possible using reversible-jump Markov chain Monte Carlo (RJMCMC) (Green 1995), with a combination of birth, death and shift moves for changepoints.

References

Anderson, T.W.: The Statistical Analysis of Time-Series. Wiley, New York (1971)

AsSadhan, B., Moura, J.M.F.: An efficient method to detect periodic behavior in botnet traffic by analyzing control plane traffic. J. Adv. Res. **5**(4), 435–448 (2014)

Barbosa, R.R.R., Sadre, R., Pras, A.: Towards periodicity based anomaly detection in SCADA networks. In: Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies Factory Automation (ETFA 2012), pp. 1–4 (2012)

Bartlett, G., Heidemann, J., Papadopoulos, C.: Low-rate, flow-level periodicity detection. In: 2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 804–809 (2011)

Bernardo, J.M., Smith, A.F.M.: Bayesian Theory. Wiley Series in Probability and Statistics. Wiley, New York (1994)

Bilge, L., Balzarotti, D., Robertson, W., Kirda, E., Kruegel, C.: DISCLOSURE: detecting botnet command and control servers through large-scale netflow analysis. In: ACSAC 2012, 28th Annual Computer Security Applications Conference, December 3–7, 2012, Orlando, Florida, USA (2012)

Chen, L.M., Hsiao, S.W., Chen, M.C., Liao, W.: Slow-paced persistent network attacks analysis and detection using spectrum analysis. IEEE Syst. J. **10**(4), 1326–1337 (2016)

Cicuttin, A., Colavita, A.A., Cerdeira, A., Mutihac, R., Turrini, S.: A simple method for detecting periodic signals in sparse astronomical event data. Astrophys. J. **498**(2), 666–670 (1998)

de Lichtenberg, U., Jensen, L.J., Fausbøll, A., Jensen, T.S., Bork, P., Brunak, S.: Comparison of computational methods for the identification of cell cycle-regulated genes. Bioinformatics **21**(7), 1164–1171 (2005)

Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. B **39**(1), 1–38 (1977)

Donoho, D.L., Johnstone, I.M.: Ideal spatial adaptation by wavelet shrinkage. Biometrika **81**(3), 425–455 (1994)

Escobar, M.D., West, M.: Bayesian density estimation and inference using mixtures. J. Am. Stat. Assoc. **90**(430), 577–588 (1995)

Eslahi, M., Rohmad, M.S., Nilsaz, H., Naseri, M.V., Tahir, N.M., Hashim, H.: Periodicity classification of HTTP traffic to detect HTTP botnets. In: 2015 IEEE Symposium on Computer Applications Industrial Electronics (ISCAIE), pp. 119–123 (2015)

Fisher, R.A.: Tests of significance in harmonic analysis. Proc. R. Soc. Lond. Ser. A Contain. Pap. Math. Phys. Charact. **125**(796), 54–59 (1929)

Green, P.J.: Reversible Jump Markov Chain Monte Carlo computation and Bayesian model determination. Biometrika **82**(4), 711–732 (1995)

Gu, G., Zhang, J., Lee, W.: BotSniffer: Detecting botnet command and control channels in network traffic. In: Proceedings of the 15th Annual Network and Distributed System Security Symposium (2008)

He, X., Papadopoulos, C., Heidemann, J., Mitra, U., Riaz, U.: Remote detection of bottleneck links using spectral and statistical methods. Comput. Netw. **53**(3), 279–298 (2009)

Heard, N.A., Rubin-Delanchy, P.T.G., Lawson, D.J.: Filtering automated polling traffic in computer network flow data. In: Proceedings—2014 IEEE Joint Intelligence and Security Informatics Conference, JISIC 2014, pp. 268–271 (2014)

Heard, N., Turcotte, M.: Monitoring a Device in a Communication Network, Chapter 6, pp. 151–188. Imperial College Press, London (2014)

Higdon, D.M.: Auxiliary variable methods for Markov Chain Monte Carlo with applications. J. Am. Stat. Assoc. **93**(442), 585–595 (1998)

Hofstede, R., Čeleda, P., Trammell, B., Drago, I., Sadre, R., Sperotto, A., Pras, A.: Flow monitoring explained: from packet capture to data analysis with NetFlow and IPFIX. IEEE Commun. Surv. Tutor. **16**(4), 2037–2064 (2014)

Hubballi, N., Goyal, D.: FlowSummary: summarizing network flows for communication periodicity detection. In: Maji, P., Ghosh, A.,

Murty, M.N., Ghosh, K., Pal, S.K. (eds.) Pattern Recognition and Machine Intelligence, pp. 695–700. Springer, Berlin (2013)
Huynh, N.A., Ng, W.K., Ulmer, A., Kohlhammer, J.: Uncovering periodic network signals of cyber attacks. In: 2016 IEEE Symposium on Visualization for Cyber Security (VizSec), pp. 1–8 (2016)
Jaynes, E.T.: Maximum entropy and Bayesian spectral analysis and estimation problems. In: Bayesian Spectrum and Chirp Analysis, pp. 1–37. Dordrecht (1987)
Jenkins, G.M., Priestley, M.B.: The spectral analysis of time-series. J. R. Stat. Soc. Ser. B (Methodol.) **19**(1), 1–12 (1957)
Kocak, M., George, E.O., Pyne, S., Pounds, S.: An empirical Bayes approach for analysis of diverse periodic trends in time-course gene expression data. Bioinformatics **29**(2), 182–188 (2013)
Lau, J.W., Green, P.J.: Bayesian model-based clustering procedures. J. Comput. Graph. Stat. **16**(3), 526–558 (2007)
Li, Z., Ding, B., Han, J., Kays, R., Nye, P.: Mining periodic behaviors for moving objects. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '10, ACM, New York, NY, USA, pp. 1099–1108 (2010)
Liu, J.: The collapsed Gibbs sampler in Bayesian computations with applications to a gene regulation problem. J. Am. Stat. Assoc. **89**(427), 958–966 (1994)
McPherson, S., Ortega, A.: Detecting low-rate periodic events in internet traffic using renewal theory. In: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4336–4339 (2011)
Neal, R.M.: Markov Chain sampling methods for Dirichlet process mixture models. J. Comput. Graph. Stat. **9**(2), 249–265 (2000)
Percival, D.B., Walden, A.T.: Spectral Analysis for Physical Applications. Cambridge University Press, Cambridge (1993)
Price-Williams, M., Heard, N.A.: Nonparametric self-exciting models for computer network traffic. Stat. Comput. **30**, 209–220 (2020)
Price-Williams, M., Heard, N.A., Turcotte, M.J.M.: Detecting periodic subsequences in cyber security data. In: 2017 European Intelligence and Security Informatics Conference (EISIC), pp. 84–90 (2017)
Qiao, Y., Yang, Y., He, J., Liu, B., Zeng, Y.: Detecting parasite P2P botnet in eMule-like networks through quasi-periodicity recognition. In: Kim, H. (ed.) Information Security and Cryptology—ICISC 2011, pp. 127–139. Springer, Berlin (2012)
Qiao, Y., Yang, Y.X., He, J., Tang, C., Zeng, Y.Z.: Detecting P2P bots by mining the regional periodicity. J. Zhejiang Univ. Sci. C **14**(9), 682–700 (2013)
Richardson, S., Green, P.J.: On Bayesian analysis of mixtures with an unknown number of components. J. R. Stat. Soc. B **59**(4), 731–792 (1997)
Siegel, A.F.: Testing for periodicity in a time series. J. Am. Stat. Assoc. **75**(370), 345–348 (1980)
Turcotte, M.J.M., Kent, A.D., Hash, C.: Unified Host and Network Data Set, Chapter 1, pp. 1–22. World Scientific, Singapore (2018)
West, M., Müller, P., Escobar, M.D.: Hierarchical priors and mixture models, with applications in regression and density estimation. Aspects of Uncertainty: A Tribute to D. V. Lindley, pp. 363–386 (1994)