Supplementary material On Bayesian New Edge Prediction and Anomaly Detection in Computer Networks

Silvia Metelli, Nicholas Heard

1 Simulation Study

In this section we conduct a simulation study to assess the performance of our model against model misspecification and sensitivity to prior distributions, for both model parameters α and β. Attention will be mainly focused on the sociability parameter β as this represents the quantity of central interest. We simulate 100 datasets each containing 100,000 events, with time-varying covariates and corresponding event times (following exponential distributions) generated according to the model proposed in (4.1). The sizes of the client and server sets are set to $|X| = |Y| = 200$. The number of clusters and their sizes were generated uniformly at random. The nuisance model coefficients are set to $\alpha = (1.5, 1.5, 1.5, 1.5)$. For the clustering formulation, we set $\beta_{1,l} = 2$ for $l = 1, \dots, L$ and $\beta_{2,m} = 2$ for $m = 1, \dots, M$, while for the latent formulation we set the single latent coefficient to $\beta = 2$. For the cluster formulation, we report the values $\bar{\beta}_1$ and $\bar{\beta}_2$ averaged across cluster dimensions, for ease of explanation.

Model misspecification is evaluated omitting important covariates of the model, i.e. either omitting cluster-level or latent-feature covariates. In particular, for the clustering case we consider two different scenarios: in the first we omit server clustering and we only perform client clustering while for the second we omit both client and server clustering. The first scenario corresponds to setting $\beta_{2,m} = 0$, while the second corresponds to setting both $\beta_{1,l} = 0$ and $\beta_{2,m} = 0$, for $l = 1, \dots, L$ and $m = 1, \dots, M$. For the latent case, this corresponds to $\beta = 0$. In addition, for both model formulations we consider the following choices of prior distributions:

1. $\alpha, \beta_{xy} \sim N(0, 1)$,
2. $\alpha, \beta_{xy} \sim N(0, 5)$,

3. $\alpha, \beta_{xy} \sim N(0, 100)$,

namely a standard normal prior, a weakly informative normal prior and a flat prior. Results are shown in Table 1. We report the posterior means of each coefficient, the associated standard error, and the relative bias, calculated as $(\hat{\theta}_{\text{MC}} - \theta)/\theta$, where θ is the true value of the parameter of interest and $\hat{\theta}_{\text{MC}}$ is the Monte Carlo average of the model estimates. Results show that we achieve good performance under the saturated model and all different prior distributions, suggesting that model inference is not significantly affected by the choice of the hyperparameters. In particular, we can notice a slight increase in both relative bias and standard error when the flat prior is used. For the latent formulation, model inference has been performed both under the simple beta-Bernoulli (BeP) process and under the more complex IBP structure. Note that, under the latent model formulation, different prior choices for the elements of the latent position matrices U and V (which are assigned standard normal priors) have not been tested: Changing the variance of the normal prior would only yield a change in the magnitude of the coefficient β, which could be then rescaled. This can be easily viewed considering the equivalence between the two following models:

$$\begin{aligned} 1:\ \lambda_{xy}(t) = r(t) \exp\{\alpha \cdot (N_x^+(t), N_y^-(t), I_{x,1}(t), I_{x,2}(t)) + \beta_{xy} \cdot u_x^T v_y\} \\ \times \mathbb{1}_{(X\times Y)\setminus G_t}\{(x,y)\}, \\ u_{x_i} \sim N(0,1), v_{y_i} \sim N(0,1), \end{aligned}$$

$$\begin{aligned} 2:\ \lambda_{xy}(t) = r(t) \exp\{\alpha \cdot (N_x^+(t), N_y^-(t), I_{x,1}(t), I_{x,2}(t)) + u_x^T v_y\} \\ \times \mathbb{1}_{(X\times Y)\setminus G_t}\{(x,y)\}, \\ u_{x_i} \sim N(0,\beta_{xy}), v_{y_i} \sim N(0,\beta_{xy}). \end{aligned}$$

			Cluster model (full)		**Cluster model** ($\beta_2 = 0$)		**No clusters** ($\beta_1 = \beta_2 = 0$)		**Latent model** (BeP)		**Latent model** (IBP)			
	Coeff	True	Mean (SE)	Bias	Mean (SE)	Bias	Mean (SE)	Bias	Bias	Mean (SE)	Bias	Mean (SE)	Coeff	True
N(0,1)	α_1	1.5	1.42 (0.18)	-0.050	1.32 (0.37)	-0.120	0.90 (0.56)	-0.400	1.40 (0.64)	-0.067	1.43 (0.18)	-0.046	α_1	1.5
	α_2	1.5	1.45 (0.25)	-0.030	1.20 (0.39)	-0.200	1.05 (0.58)	-0.300	1.37 (0.55)	-0.087	1.46 (0.15)	-0.026	α_2	1.5
	α_3	1.5	1.39 (0.21)	-0.073	1.25 (0.38)	-0.160	0.96 (0.45)	-0.360	1.41 (0.58)	-0.060	1.42 (0.17)	-0.053	α_3	1.5
	α_4	1.5	1.42 (0.20)	-0.025	1.34 (0.33)	-0.107	0.90 (0.59)	-0.400	1.38 (0.55)	-0.080	1.47 (0.21)	-0.020	α_4	1.5
	$\bar{\beta}_1$	2.0	2.09 (0.19)	0.045	1.29 (0.39)	-0.355	-	-	1.86 (0.49)	-0.070	2.03 (0.15)	0.015	β	2.0
	$\bar{\beta}_2$	2.0	2.04 (0.22)	0.020	-	-	-	-	-	-	-	-	-	-
N(0,5)	α_1	1.5	1.41 (0.35)	-0.060	1.28 (0.55)	-0.147	0.98 (0.45)	-0.346	1.36 (0.50)	-0.093	1.41 (0.35)	-0.060	α_1	1.5
	α_2	1.5	1.41 (0.33)	-0.060	1.28 (0.34)	-0.147	0.97 (0.55)	-0.353	1.37 (0.44)	-0.087	1.41 (0.33)	-0.060	α_2	1.5
	α_3	1.5	1.33 (0.44)	-0.113	1.22 (0.41)	-0.187	0.96 (0.61)	-0.360	1.29 (0.44)	-0.140	1.33 (0.44)	-0.113	α_3	1.5
	α_4	1.5	1.56 (0.45)	0.040	1.23 (0.28)	-0.180	0.98 (0.54)	-0.346	1.41 (0.50)	-0.060	1.56 (0.45)	0.040	α_4	1.5
	$\bar{\beta}_1$	2.0	2.12 (0.39)	0.060	1.25 (0.31)	-0.167	-	-	1.78 (0.61)	-0.110	1.95 (0.42)	-0.025	β	2.0
	$\bar{\beta}_2$	2.0	1.89 (0.32)	-0.055	-	-	-	-	-	-	-	-	-	-
N(0,100)	α_1	1.5	1.37 (0.55)	-0.086	1.19 (0.61)	-0.210	0.98 (0.49)	-0.346	1.29 (0.54)	-0.140	1.37 (0.55)	-0.086	α_1	1.5
	α_2	1.5	1.39 (0.43)	-0.073	1.21 (0.42)	-0.193	0.99 (0.61)	-0.340	1.35 (0.47)	-0.100	1.39 (0.43)	-0.073	α_2	1.5
	α_3	1.5	1.37 (0.39)	-0.086	1.22 (0.39)	-0.187	0.96 (0.55)	-0.360	1.39 (0.55)	-0.073	1.39 (0.43)	-0.073	α_3	1.5
	α_4	1.5	1.42 (0.45)	-0.025	1.18 (0.45)	-0.210	0.93 (0.54)	-0.380	1.38 (0.45)	-0.080	1.42 (0.45)	-0.025	α_4	1.5
	$\bar{\beta}_1$	2.0	2.17 (0.54)	0.085	1.11 (0.44)	-0.440	-	-	1.82 (0.47)	-0.090	1.93 (0.55)	-0.035	β	2.0
	$\bar{\beta}_2$	2.0	2.15 (0.55)	0.075	-	-	-	-	-	-	-	-	-	-

Table 1: Estimated posterior means (with standard errors) and relative bias of model coefficient parameters for both model formulations under the different prior distributions considered.

2 Posterior inference

The Metropolis-Hastings (M-H) algorithm is used to draw approximate samples from the joint posterior distributions of the clustering and latent feature formulations. For altering α and β, simple random walks with Gaussian steps are applied to a randomly selected component, and so hereafter attention is focused on sampling the clustering configuration or latent features.

2.1 Clustering formulation MCMC algorithm

To initialise the algorithm, row and column cluster configurations are first obtained through the spectral biclustering algorithm described in Section 5.

Let α^t, β^t, $\mathbb{C}^t$ and $\mathbb{S}^t$ be the values of the parameter vectors and cluster configurations after t iterations, and suppose at iteration $t+1$ we wish to propose a change to $\mathbb{C}^t$. A client x is randomly chosen from X, with current cluster label $\mathbb{C}^t(x)$, and then a new cluster label $\mathbb{C}^*(x)$ is proposed from a discrete uniform proposal distribution over the integer set $\{1, \ldots, L^t + 1\}/\{\mathbb{C}^t(x)\}$, where L^t is the current number of client clusters in $\mathbb{C}^t$. The proposed value $\mathbb{C}^*(x)$ suggests a new cluster configuration $\mathbb{C}^*$ with L^* client clusters. If $|\mathbb{C}_{\mathbb{C}^t(x)}| = 1$ and $\mathbb{C}^t(x) \neq L^t + 1$, then $L^* = L^t - 1$; or else if $|\mathbb{C}_{\mathbb{C}^t(x)}| > 1$ and $\mathbb{C}^*(x) = L^t + 1$, then $L^* = L^t + 1$. In both of these cases, the dimension of β must change, either deleting those components corresponding to the emptied client cluster or else proposing a new vector of values for a new cluster from the prior. By the M-H algorithm, the proposed parameters are accepted with probability

$$\min\left(1, \frac{\mathbb{P}(\mathbb{C}^*, \mathbb{S}^t, \alpha^t, \beta^* | \mathcal{T}', \mathcal{E}') L^t}{\mathbb{P}(\mathbb{C}^t, \mathbb{S}^t, \alpha^t, \beta^t | \mathcal{T}', \mathcal{E}') L^*}\right).$$

Sampling of $\mathbb{S}$ is directly analogous.

2.2 Latent feature formulation MCMC algorithm

For the latent feature model, sparse singular value decomposition with stability selection, as described in Section 6, is used to provide reliable initial latent positions of clients and servers, parametrised through $\tilde{U}, \tilde{V}, \Delta_U, \Delta_V$. Suppose at iteration $t+1$ we wish to propose changes to $\tilde{U}^t$ and Δ_U^t. (Analogous approaches are used to sample $\tilde{V}^t$ and Δ_V^t). In the following, for notational convenience we will omit the subscript from Δ_U, which will be simply denoted Δ.

- *Sampling Δ_{xk}:* For a randomly sampled client x, let

$$K_x^t = \{k | 1 \leq k \leq K_U, \Delta_{xk}^t = 1\}$$

be the features currently activated for that client in the latent feature model. Further, for $k \in \{1, \ldots, K_U\}$ let $d_k^t = \sum_{x \in X} \Delta_{xk}^t$be the number of clients with feature k currently active.

For a randomly chosen feature $k \in \{1, \ldots, K_U + 1\}$, we can resample δ_k from its full conditional distribution,

$$\mathbb{P}(\Delta_{xk} | \Delta_{-(xk)}^t, \ldots) \propto \mathbb{P}(\mathcal{E}' | \mathcal{T}', U, V, \alpha, \beta) \mathbb{P}(\Delta_{xk} | \Delta_{-(xk)}^t),$$

where $\Delta_{-(xk)}$ is the Δ matrix excluding the $\Delta(xk)$ element and the second term of the equation is the conditional prior distribution for the new value of Δ_{xk}. If $d_k^t > 1$ or $\Delta_{xk}^t = 0$, then

$$\mathbb{P}(\Delta_{xk} | \Delta_{-(xk)}^t) = \frac{(d_k^t)^{\Delta_{xk}} (|X| - d_k^t)^{1 - \Delta_{xk}}}{|X|}.$$

Alternatively, if $d_k^t = \Delta_{xk}^t = 1$ such that x is the only client with feature k active, then U may potentially decrease in dimension and by the recursive formula for the Poisson distribution we have

$$\frac{\mathbb{P}(\Delta_{xk} = 1 | \Delta_{-(xk)}^t)}{\mathbb{P}(\Delta_{xk} = 0 | \Delta_{-(xk)}^t)} = \frac{\theta}{|X||K_x^t|}.$$

Finally if $k = (K_U + 1)$, proposing an increase in dimension of U,

$$\frac{\mathbb{P}(\Delta_{xk} = 1 | \Delta_{-(xk)}^t)}{\mathbb{P}(\Delta_{xk} = 0 | \Delta_{-(xk)}^t)} = \frac{\theta}{|X|(|K_x^t| + 1)}.$$

- *Sampling $\tilde{u}_{xk}$:* Simple random walks with Gaussian steps are applied to each randomly selected value $\tilde{u}_{xk}$ of $\tilde{U}$.

- *Sampling θ:* Under the IBP model, for each client x the distribution of the number of sampled features is Poisson(θ/x); assuming the conjugate prior $\Gamma(a_\theta, b_\theta)$ for θ, samples can be drawn directly from the the posterior distribution which is $\Gamma(a_\theta + K_U, b_\theta + \sum_{x=1}^{|X|} 1/x)$.

3 Normality test for $\hat{\Lambda}$

In this section we assess approximate normality for the matrix $\hat{\Lambda}$, which is used in the penalised regression problem

$$\|\hat{\Lambda} - suv^T\|_F^2 + \rho_u P_1(s,u) + \rho_v P_2(s,v).$$

We perform Shapiro-Wilk normality tests for normality on each row of the $\hat{\Lambda}$ matrix, corresponding to each client in the network. Figure 1 shows the distribution of the p-values resulting from the tests: we can notice that in the majority of the cases there is no departure from normality.

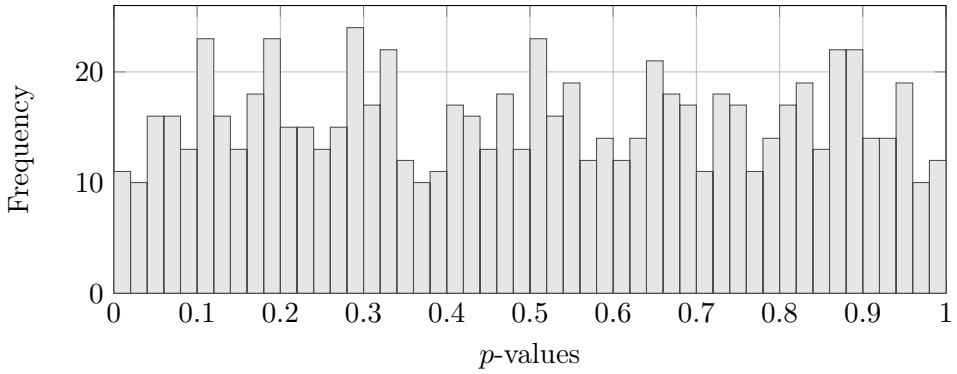

Figure 1: Distribution of Shapiro-Wilk tests p-values for $\hat{\Lambda}$.

4 Evaluation of MCMC sampling

In this section we provide details about the computational performance of the sampling scheme adopted. The total number of iterations for each of the 15 sample repetitions was set to 5,000, after a burn-in phase of size 1,000. Table 2 shows the effective sample sizes and acceptance ratios for each parameter, estimated under both model formulations. Again, we report average ESS values for $\bar{\beta}_1$ and $\bar{\beta}_2$.

Then, we assess the chain convergence and mixing through the marginal likelihood for each MCMC iteration (Figure 2), which measures the goodness-of-fit of the saturated model. The process appears stationary as the number of iterations increases. Finally, to find an optimal estimate for burn-in cut-off, we have run the MCMC chain with different burn-in sizes, and the ESS

Cluster model

	α_1	α_2	α_3	α_4	β_1	β_2
ESS	1423	1766	1401	1245	1343	1733
AR	0.22	0.22	0.23	0.25	0.22	0.21

Latent model

	α_1	α_2	α_3	α_4	β
ESS	1782	1911	1832	1792	1981
AR	0.21	0.26	0.31	0.33	0.23

Table 2: Effective sample size (ESS) and acceptance ratio (AR) for each coefficient in the cluster model (top table) and latent model (bottom table).

of each variable has been plotted against burn-in: if the burn-in period is estimated to be too short this will reduce the ESS size. Analogously, with a too long burn-in period, informative samples are thrown away, thus reducing the ESS. The ESS should be maximised at the optimal estimate of the burn-in. Figure 3 shows the ESS at varying burn-in sizes for the latent structure parameter β, which is the coefficient of main interest. Here, a burn-in phase of size 1000 appears to be a suitable choice.

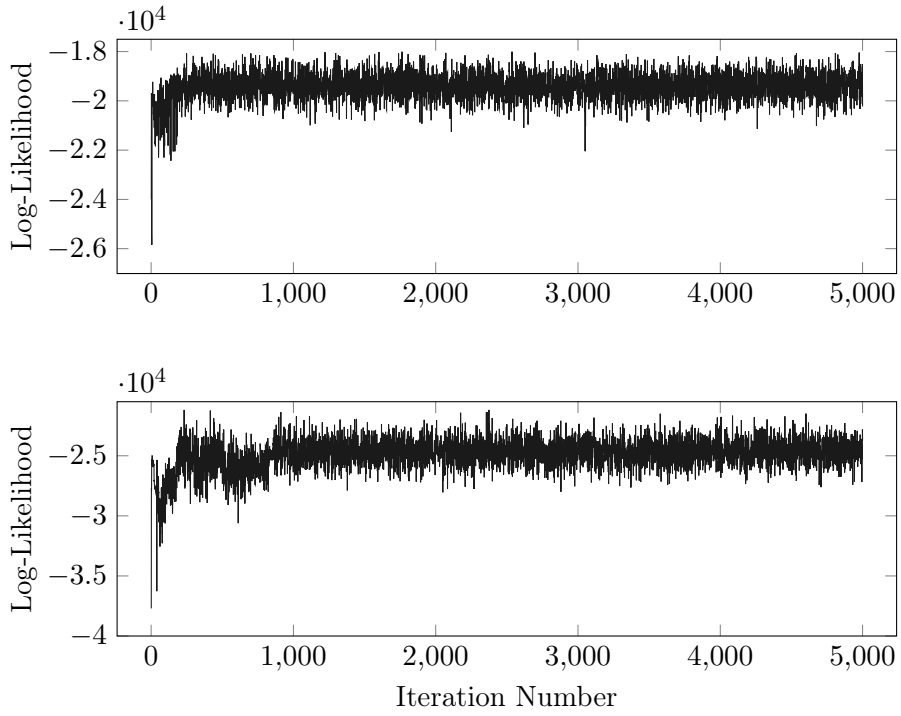

Figure 2: Log-likelihood vs. number of MCMC iterations, for the cluster formulation (top), and for the latent formulation (bottom).

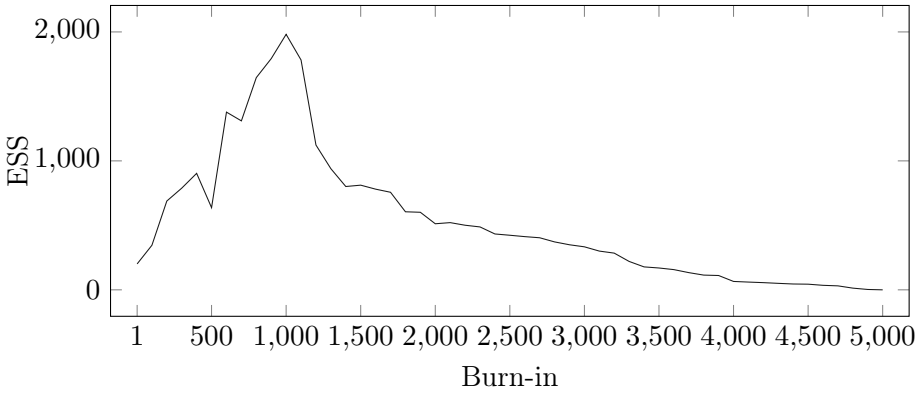

Figure 3: ESS vs burn-in size for the parameter β in the full latent feature model, under the IBP prior.