Seasonality in Dynamic Stochastic Block Models

Jace Robinson
Dept. of Computer Science & Engineering
Kno.e.sis Research Center
Wright State University
Dayton, Ohio
robinson.329@wright.edu

Derek Doran
Dept. of Computer Science & Engineering
Kno.e.sis Research Center
Wright State University
Dayton, Ohio
derek.doran@wright.edu

ABSTRACT

Sociotechnological and geospatial processes exhibit time varying structure that make insight discovery challenging. This paper proposes a new statistical model for such systems, modeled as dynamic networks, to address this challenge. It assumes that vertices fall into one of k types and that the probability of edge formation at a particular time depends on the types of the incident nodes and the current time. The time dependencies are driven by unique seasonal processes, which many systems exhibit (e.g., predictable spikes in geospatial or web traffic each day). The paper defines the model as a generative process and an inference procedure to recover the seasonal processes from data when they are unknown. Evaluation with synthetic dynamic networks show the recovery of the latent seasonal processes that drive its formation.

KEYWORDS

Dynamic Networks, Kalman Filter, Structural Time Series, State Space Model

1 INTRODUCTION & MOTIVATION

Many complex systems exhibit regular, time dependent, *seasonal* patterns. For example, human movement patterns are driven by the time of day [8], and vehicle traffic densities exhibit predictable increases at certain hours causing rush hours and decreases at night [6]. This same 'seasonal', time dependent effect occurs when monitoring network bandwidth usage [13] or when counting the number of clicks per day on a web page[2].

We look to bring the notion of seasonality to statistical network modeling with a new kind of dynamic stochastic block model (DSBM). A DSBM asserts that system components (nodes) are grouped into several types, and the probability of observing a component relation or interaction (edges) are determined by the types
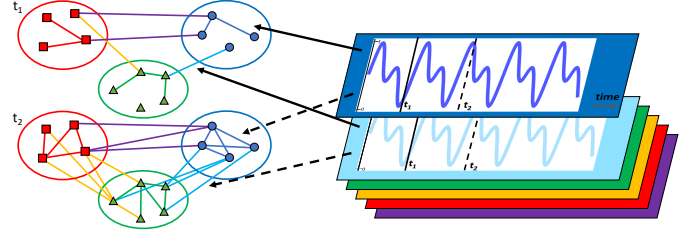
Figure 1: Example of seasonality in dynamic networks. In this three class network, a latent seasonal process determines a probability of edge formation at times t_1 and t_2, subject to both process and measurement noise. A different process (colored plates on the right) affect probabilities for edges connecting unique pairs of node types. The paper presents a statistical model codifying these ideas, which may be useful in comparison, prediction, and anomaly detection tasks on dynamic complex systems.

of the incident nodes and time. Different kinds of DSBM consider different assumptions about the network formation process [4, 9–12], but none consider seasonality. A conceptual overview of the model we propose is given in Figure 1. It fuses structural time series (plates on the right of Figure 1) with a generative network model. We call this a seasonal DSBM (SDSBM). The model is transformed to a state-space model for scalable fitting to data by Kalman Filters, and expectation-maximization is used for parameter learning.

2 MODEL SPECIFICATION

We first specify the model of the seasonal processes controlling edge dynamics. We assume that time is discrete, with the current time t representing a time period of some resolution. We also assume the node types are provided. For each pair of node types a and b, we consider a structural time series with a *bias* $m_t^{(a,b)}$ establishing an anchor for values of the time series and a *seasonal offset* $s_t^{(a,b)}$ that shifts the bias by the current seasonality position. The process at time t, denoted $c_t^{(a,b)}$, is

$$c_t^{(a,b)} = m_t^{(a,b)} + s_t^{(a,b)} \tag{1}$$

with bias $m_t^{(a,b)}$ described by $m_t^{(a,b)} = m_{t-1}^{(a,b)} + \delta_{m_t^{(a,b)}}$ where $\delta_{m_t^{(a,b)}} \sim \mathcal{N}(0, q_m^{(a,b)})$ models possible process noise. The set of seasonal offsets are stored in a vector $\mathbf{s}^{(a,b)} = (s_1^{(a,b)}, s_2^{(a,b)}, ..., s_d^{(a,b)})$ having d components. d reflects either the length or the resolution of a seasonal process (e.g., $d = 60$ to model per minute changes over a process that cycles per hour) and is assumed to be provided

Jace Robinson and Derek Doran

by the user of the model. The components are:

$$s_t^{(a,b)} = -\sum_{i=1}^{d-1} s_{t-i}^{(a,b)} + \delta_{s_t^{(a,b)}} \tag{2}$$

where $\delta_{s_t^{(a,b)}} \sim \mathcal{N}(0, q_s^{(a,b)})$. This form enforces a zero-sum constraint to increase identifiability [7]. It should be emphasized that $q_m^{(a,b)}$ and $q_s^{(a,b)}$ control the noise of the underlying seasonal process, where low noise represents a stable seasonality that does not vary significantly between realized periods.

To model how the seasonal processes govern the shape of a dynamic network, we define a random variable $E_t^{(a,b)} \in [0,1]$ as the *expected density* of edges spanning node types (a,b) at time t:

$$E_t^{(a,b)} = c_t^{(a,b)} + \epsilon_{E_t^{(a,b)}} \tag{3}$$

where $\epsilon_{E_t^{(a,b)}} \sim \mathcal{N}(0, r^{(a,b)})$ models possible measurement noise. Next we define an adjacency matrix A_t, where $[A_t]_{ij} = 1$ if there exists an edge between nodes i and j at time t and $[A_t]_{ij} = 0$ otherwise. Denote $A_t^{(a,b)}$ as the submatrix of A_t only containing the rows and columns representing type a and type b nodes. Then $A_t^{(a,b)}$ is defined by the random variable:

$$[A_t^{(a,b)}]_{ij} \sim Bernoulli(E_t^{(a,b)}) \tag{4}$$

Repeating this process for all blocks (a,b) and time steps t will generate a desired dynamic network $\mathcal{A} = \{A_1, A_2, ..., A_t\}$.

3 MODEL FITTING

We now describe an inference procedure to fit the model to an observed $\mathcal{A}$. Since seasonal processes are latent in data, the task is to estimate a posterior distribution on each $m_t^{(a,b)}$ and $s_t^{(a,b)}$ for each pair of node types (a,b). Kalman filters [5] are an appropriate tool for this task, but requires transforming the generative model into a state-space model (SSM). A SSM is a time series model with *hidden* and *observed* variables [7]; here we define $\mathbf{x}_t$ as hidden and w_t as observed variables, respectively. We can notice the bias and seasonal offsets from before were hidden, while the adjacency matrix is observed, foreshadowing the structure to be defined. A SSM creates observations at time t by two linear models: An **observation model** $w_t = \mathbf{h}\mathbf{x}_t + \epsilon_t$ and a **transition model** $\mathbf{x}_t = \mathbf{G}\mathbf{x}_{t-1} + \Delta_t$. Here, observations w_t are generated by a transformation of the output (defined by $\mathbf{h}$) of the underlying transition model. The transition model describes transformations within a hidden state space where transitions from time $t-1$ to time t are defined by the matrix $\mathbf{G}$. Observations and transitions are to be subject to time dependent random noise, which are modeled by Gaussian distributions $\epsilon_t \sim \mathcal{N}(0, R_t)$ and $\Delta_t \sim \mathcal{N}(0, \mathbf{Q}_t)$ With R_t and $\mathbf{Q}_t$ controlling the amount of observation and transition noise, respectively. Assuming parameters $\theta_t = \{\mathbf{h}, \mathbf{G}, R_t, \mathbf{Q}_t\}$ are known, a Kalman Filter can be used to derive the exact posterior $Pr(\mathbf{x}_t | \theta_t, w_1, w_2, ..., w_t)$, i.e., the probability of the hidden state value at time t given the observations up to and including time t [7].

Now we transform the model specification into a state space to define the transition model $\mathbf{x}_t = \mathbf{G}\mathbf{x}_{t-1} + \Delta_t$. As we are assuming edges of different vertex types (a,b) are independent of each other, we will formulate the inference in terms of a pairing (a,b). The

full inference is completed by repeating the process for all pairs (a,b). First we transform Equations 1 and 2 to define the hidden state variable $\mathbf{x}_t$ and state transition $\mathbf{G}$. The hidden state will be composed of the bias and vector of seasonal offsets as a $d \times 1$ seasonal state vector for a period of length d:

$$\mathbf{x}_t^{(a,b)} = \begin{bmatrix} m_t^{(a,b)} & s_t^{(a,b)} & s_{t-1}^{(a,b)} & \dots & s_{t-d+2}^{(a,b)} \end{bmatrix}^T \tag{5}$$

Note that all the seasonal offsets from $\mathbf{s}^{(a,b)}$ are maintained in the state for a given t, with the dth seasonal offset implicitly defined based on the zero-sum constraint. Now to perform the state transition from time $t-1$ to time t we define a $d \times d$ matrix $\mathbf{G}$:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & -1 & -1 & \dots & -1 & -1 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix} \tag{6}$$

In $\mathbf{G}$, we see that multiplication of the first row of $\mathbf{G}$ by $\mathbf{x}_t^{(a,b)}$ yields Equation 1 without noise being added, as only the bias term is updated. Multiplication of the second row of $\mathbf{G}$ by $\mathbf{x}_t^{(a,b)}$ will update a *single* seasonal offset as shown in Equation 2. The remaining rows of $\mathbf{G}$ serve to permute the remaining seasonal offsets, such that each offset $s_i^{(a,b)}$ is updated after a full period of d time steps. Each time step will update the most current seasonal offset, and shift the remaining offsets right one index in the state vector. Next we define the $d \times 1$ noise vector $\Delta_t = [\delta_{m_t^{(a,b)}}, \delta_{s_t^{(a,b)}}, 0, \dots, 0]^T$ where the first element is the bias noise in Equation 1, the second element is seasonal noise in Equation 2, and the remaining elements are all 0 as there is no additional noise for permuting the seasonal offsets. These noise values are sampled from a zero mean Gaussian with $d \times d$ covariance matrix $\mathbf{Q} = \mathtt{diag}[q_m^{(a,b)}, q_s^{(a,b)}, 0, \dots, 0]$. Assuming the bias noise $\delta_{m_t^{(a,b)}}$ and seasonal offset noise $\delta_{s_t^{(a,b)}}$ are independent, the off-diagonal elements of $\mathbf{Q}$ are zero. The remaining elements are all zero, reflecting the lack of noise for the permutation operations. We can assume $\mathbf{Q}$ is stationary, and drop the dependence on t. This complete formulation of the transition model is not new, and has been completed by other researchers such as in [1].

Our next task is to transform Equations 3 and 4 into the observation model $w_t = \mathbf{h}\mathbf{x}_t + \epsilon_t$. To do this, we will need to define some additional variables, and take advantage of a result of the central limit theorem for a large number of vertices with types (a,b) to create an approximate Gaussian transformation. First we need a count of the number of *possible edges* in block (a,b), so if there are $|a|$ nodes of type a and $|b|$ nodes of type b then define:

$$n^{(a,b)} = \begin{cases} a = b & \frac{|a|(|a|-1)}{2} \\ a \neq b & |a||b| \end{cases} \tag{7}$$

Also define the random variable $p_t^{(a,b)} \sim Binomial(E_t^{(a,b)}, n^{(a,b)})$ as the number of *formed edges* in block (a,b) at time t, where $E_t^{(a,b)}$ is the expected edge density as determined by Equation 3. $p_t^{(a,b)}$ is simply a more mathematically convenient way to define the edge generation process from Equation 4 and does not change the

overall model. Having a binomial random variable, for large enough $n^{(a,b)}E_t^{(a,b)}$ we can apply the central limit theorem to approximate the distribution of $p_t^{(a,b)}$ as Gaussian:

$$p_t^{(a,b)} = n^{(a,b)}E_t^{(a,b)} + \omega_{p_t^{(a,b)}} \tag{8}$$

where $\omega_{p_t^{(a,b)}} \sim \mathcal{N}(0, n^{(a,b)}E_t^{(a,b)}(1 - E_t^{(a,b)}))$ is *observation noise* that is time dependent on $E_t^{(a,b)}$. This represents variation in the repeated binary decision process of whether pairs of complex system components will interact. For example, in a geospatial context, a seasonal process may dictate that people travel from home to work at 8am, yet people individually decide the precise time they leave for work. In this example, locations of the geospace are system components (vertices), and movement between these locations are interactions (edges). In the SDSBM, each system component will have a type, such that each individual home is a vertex and all these homes can have the same type of 'residence'. The sum of individual departure time decisions creates variation captured in $\omega_{p_t^{(a,b)}}$.

Now that we have successfully transformed our edge sampling procedure from Equation 4 to an approximately Gaussian formulation in Equation 8, we can return to defining observation model parameters w_t and $\mathbf{h}$. We will define the observed variable $w_t^{(a,b)}$ as the number of formed edges $p_t^{(a,b)}$. To create $\mathbf{h}$, define a transformation which takes as input a seasonal state vector $\mathbf{x}_t^{(a,b)}$ and produces as output the number of formed edges $w_t^{(a,b)}$. By combining the operations of Equations 3 and 8 we define:

$$\mathbf{h} = \begin{bmatrix} n^{(a,b)} & n^{(a,b)} & 0 & \dots & 0 \end{bmatrix} \tag{9}$$

Examining $w_t^{(a,b)} = \mathbf{h}\mathbf{x}_t^{(a,b)}$ closer, we can see this multiplication both sums the bias $m_t^{(a,b)}$ and first seasonal offset $s_t^{(a,b)}$ following Equation 3, and multiplies by $n^{(a,b)}$ to match the repeated Bernoulli trials from Equation 8. Finally, we model the variance of the measurement noise ϵ_t by summing measurement noise $\epsilon_{E_t^{(a,b)}}$ and observation noise $\omega_{p_t^{(a,b)}}$. This definition allowing for more flexible modeling of many complex systems. ϵ_t is sampled from a zero mean Gaussian distribution with time dependent variance $R_t^{(a,b)} = n^{(a,b)}E_t^{(a,b)}(1 - E_t^{(a,b)}) + (n^{(a,b)})^2 r^{(a,b)}$.

We have now transformed the generative procedure to the suitable SSM to allow easy inference via the Kalman Filter. Given an initial Gaussian belief state $Pr(\mathbf{x}_0^{(a,b)})$ with mean $\mathbf{x}_0^{(a,b)}$ and variance $\Sigma_0^{(a,b)}$, all subsequent belief states will be Gaussian as well. The closed-form updates for the posterior distribution $Pr(\mathbf{x}_t^{(a,b)}|\theta_t^{(a,b)}, w_1^{(a,b)}, w_2^{(a,b)}, ..., w_t^{(a,b)})$ are not defined in this paper due to space constraints but are available in [7].

To estimate the unknown noise parameters of the SSM and Kalman Filter $\theta_t = \{R_t, \mathbf{Q}\}$ from data, we derive an expectation-maximization routine that iteratively converges to locally optimal point estimates. The update equations for $\mathbf{Q}$ can be found in other research such as [3]. The updates for R_t is a new formulation, which deviates from the conventional EM routine for Kalman Filters, due to our separation of noise parameter $R_t^{(a,b)}$ into a combination of $E_t^{(a,b)}$, $n^{(a,b)}$, and $r^{(a,b)}$. $n^{(a,b)}$ is assumed fixed and $E_t^{(a,b)}$ can
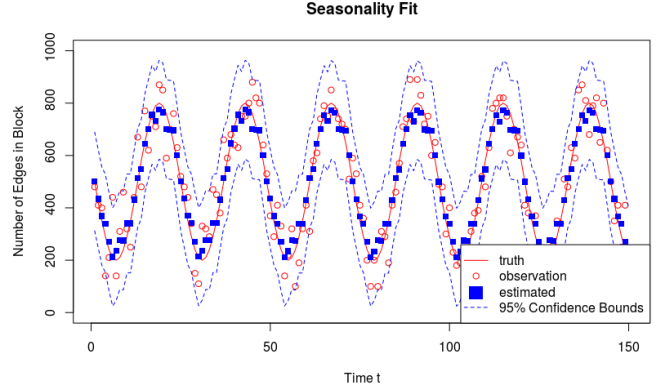

Figure 2: Comparison of the inferred seasonality (posterior distribution) compared to actual seasonality governing connections between nodes of type x and y. The underlying seasonality follows a simple sine wave that is accurately recovered from noisy observations.
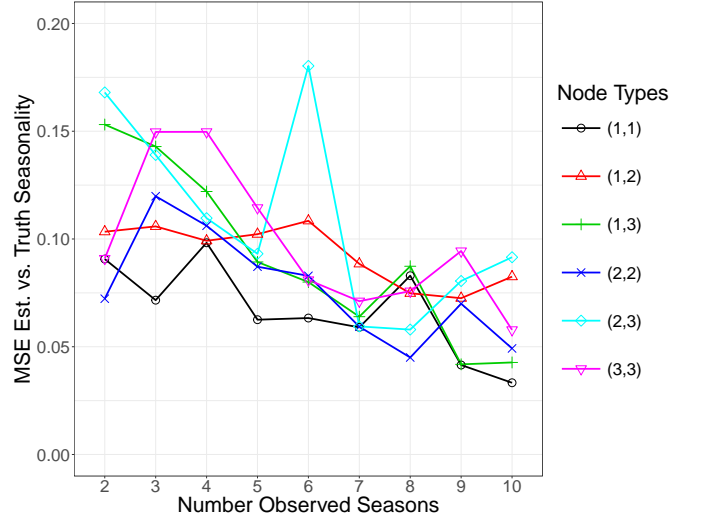

Figure 3: The MSE between the estimated seasonal state vectors to the true seasonal state vectors as a function of the number of seasonal periods observed. The error decreases linearly with observation length.

be estimated using the prediction step of the Kalman Filter as described in [11]. To estimate $r^{(a,b)}$, we set up an optimization routine, which maximizes the log-likelihood of the complete joint distribution $Pr(\mathbf{x}_1^{(a,b)}, ..., \mathbf{x}_t^{(a,b)}, w_1^{(a,b)}, ..., w_t^{(a,b)})$. This optimization is completed each iteration of EM, until a locally optimal estimate for both $\mathbf{Q}$ and R_t is found.

4 RESULTS

To demonstrate the fidelity of the SDSBM, we perform several experiments on a synthetic dynamic network $\mathcal{A} = \{A_1, A_2, ..., A_T\}$ generated following the process defined in Section 2. The network is defined with $k = 3$ vertex types, creating 6 unique pairs (a, b). There are approximately 32 nodes of each type, creating
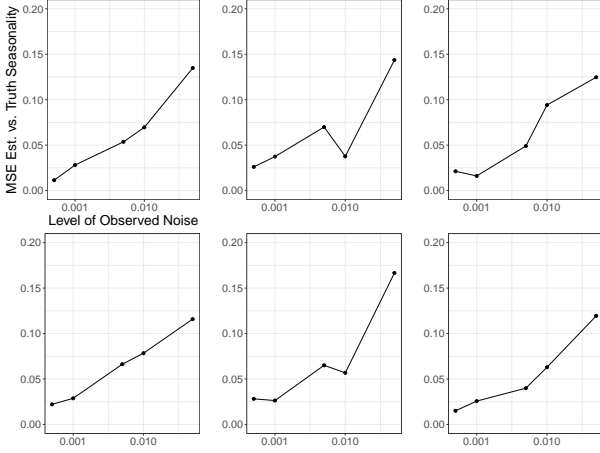
Figure 4: The MSE between the estimated seasonal state vectors to true seasonal state vectors as we increase the amount of observation noise r_t for each pair of node types. The x-axis is log scale while the y-axis is continuous. The errors appear to increase exponentially with the level of observation noise.

the number of possibles edges for each (a, b) as $n^{(a,b)} = 1000$. All pairings (a, b) have the same underlying true seasonality. We start with an initial bias $m_0^{(a,b)}$ and initial vector of seasonal off-sets $\mathbf{s}^{(a,b)} = (s_1^{(a,b)}, s_2^{(a,b)}, ..., s_d^{(a,b)})$. We set $m_0^{(a,b)} = 0.5$, and the seasonal offsets as samples from a sine wave with amplitude 0.3. This will create expected block densities $E_t^{(a,b)} \in [.2, .8]$, where noisy samples may cover the full range $[0, 1]$. For example, in some of the experiments, we set the length of seasonality as 8, creating $\mathbf{s}^{(a,b)} = (.2, .3, .2, 0, -.2, -.3, -.2, 0)$. To determine the noise level of the system, we set bias variance $q_m^{(a,b)} = 10^{-8}$ and seasonal offset variance $q_s^{(a,b)} = 10^{-8}$. These are set very low to reflect the expectation of near constant seasonality. If we set the number of possible edges of types (a, b) as $n^{(a,b)} = 1000$, we define a 'medium' amount of noise at $r^{(a,b)} = 5.5 * 10^{-3}$, which will result in a standard deviation of the number of formed edges of 75. Repeating the operations in Equations 1-4 T times will form the desired synthetic dynamic network $\mathcal{A}$.

To infer the underlying seasonal processes, thus fitting the model to $\mathcal{A}$, the user needs to provide values for seven hyperparameters. We assume the user knows both the node labels and desired length of seasonality d. Initial guesses need to be provided for the remaining hyperparameters of initial state mean $\mathbf{x}_0^{(a,b)}$, state variance $\Sigma_0^{(a,b)}$, measurement variance $r^{(a,b)}$, bias variance $q_m^{(a,b)}$, and seasonal offset variance $q_s^{(a,b)}$. These can be defaulted to initial values such as 1, without having a significant consequence on the fidelity performance. A more formal exploration of the sensitivity of the model to these hyperparameters is left as future work. We present the results of three experiments. First we simply infer the hidden seasonal state vectors $\mathbf{x}_t^{(x,y)}$ for a specific pairing (x, y) and for all t, to qualitatively demonstrate the goodness of fit of the algorithm. A visual of the seasonal state vector transformed to expected edge counts is given in Figure 2. The underlying seasonality is estimated

well, as the estimates shown as blue squares, closely follow the true seasonality of the solid red line. In the second experiment, we look to demonstrate the model's ability to recover the seasonal state vector with differing numbers of observations as controlled by T, the length of the dynamic network. We start with $T = 2d$ and increase the number of observed periods until $T = 10d$. To assess performance we calculated the mean-squared error (MSE) between the true $\mathbf{x}_t^{(a,b)}$ and estimated $\hat{\mathbf{x}}_t^{(a,b)}$ seasonal vectors for all t. In Figure 3 a negative linear dependence between MSE and number of observations is demonstrated in all pairs (a, b). The model becomes increasingly accurate for larger datasets. In the third experiment, we again evaluate the MSE between truth and estimated seasonality, but this time we vary the amount of measurement noise $r^{(a,b)}$. We start with a low $r^{(a,b)} = 5 * 10^{-4}$ and increasing a half order of magnitude (e.g., $5 * 10^{-4}, 1 * 10^{-3}, 5 * 10^{-3}$, etc...) until a high level of noise at $5 * 10^{-2}$. In Figure 4 we see the exponential dependence on the noise level. The performance of this model on real data may thus be sensitive to this measurement noise value.

5 CONCLUSION

This paper proposed a new statistical model for dynamic networks, leveraging the benefits of structural time series and stochastic block models. The generative specification and inference procedure are defined. We demonstrate the capabilities of the model on a synthetic dataset, showing some properties of the model such as the negative linear dependence to the number of observed seasonal periods.

ACKNOWLEDGMENTS

REFERENCES

[1] James W Davis and Mark A Keck. 2005. Modeling Behavior Trends and Detecting Abnormal Events using Seasonal Kalman Filters. *People* 15 (2005), 20.
[2] Bruno Gonçalves and José J Ramasco. 2008. Human dynamics revealed through Web analytics. *Physical Review E* 78, 2 (2008), 026123.
[3] Andrew C Harvey and Simon Peters. 1990. Estimation procedures for structural time series models. *Journal of Forecasting* 9, 2 (1990), 89–108.
[4] Qirong Ho, Le Song, and Eric P Xing. 2011. Evolving cluster mixed-membership blockmodel for time-varying networks. (2011).
[5] Rudolph Emil Kalman et al. 1960. A new approach to linear filtering and prediction problems. *Journal of basic Engineering* 82, 1 (1960), 35–45.
[6] Xiaolei Li, Zhenhui Li, Jiawei Han, and Jae-Gil Lee. 2009. Temporal outlier detection in vehicle traffic data. In *Data Engineering, 2009. ICDE'09. IEEE 25th International Conference on*. IEEE, 1319–1322.
[7] Kevin P Murphy. 2012. *Machine learning: a probabilistic perspective*. MIT press.
[8] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. 2010. Limits of predictability in human mobility. *Science* 327, 5968 (2010), 1018–1021.
[9] Eric P Xing, Wenjie Fu, Le Song, et al. 2010. A state-space mixed membership blockmodel for dynamic network tomography. *The Annals of Applied Statistics* 4, 2 (2010), 535–566.
[10] Kevin Xu. 2015. Stochastic Block Transition Models for Dynamic Networks.. In *AISTATS*.
[11] Kevin S Xu and Alfred O Hero III. 2013. Dynamic stochastic blockmodels: Statistical models for time-evolving networks. In *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*. Springer, 201–210.
[12] Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. 2011. Detecting communities and their evolutions in dynamic social networks—a Bayesian approach. *Machine Learning* 82, 2 (2011), 157–189. https://doi.org/10.1007/s10994-010-5214-7
[13] Wucherl Yoo and Alex Sim. 2016. Time-Series Forecast Modeling on High-Bandwidth Network Measurements. *Journal of Grid Computing* 14, 3 (2016), 463–476.