Stochastic Modeling of Infectious Disease with Temporal Logic Point Processes[*]

Mou Minghao[†]

November 25, 2022

Abstract

Contents

1 Preliminaries

1.1 Marked Temporal Point Process

Our brief introduction to *marked temporal point process (MTPP)* is based on the lecture notes [Ras18]. Marked temporal point process is a continuous-time stochastic process whose realization consists of a sequence of events $\mathcal{S}_i = (t_i, m_i)$, where $t_i \in \mathbb{R}^+$ stands for the *event time*, $m_i \in \mathcal{M}$ is the corresponding *marker* of the event, and $i \in \mathbb{Z}^+$ is the event index. We use $\mathcal{H}(t)$ to denote the past history of the process up to and inclue time t (Alternatively, it is written as $\mathcal{H}_t$. To be consistent with our later presentation, we adopt the convention $\mathcal{H}(t)$)

$$\mathcal{H}(t) := \{\mathcal{S}_i = (t_i, m_i)\}_{i=1}^{N}, t_N \leq t.$$

There are different strategies to specify a temporal point process. Two of the main strategies are *specifying the distribution of the interarrival time* $d_i = t_i - t_{i-1}$ and *specifying the conditional intensity function* $\lambda^*(t|\mathcal{H}(t))$.

Interarrival times. The length between two consecutive events is called the *interarrival time*. We can specify a temporal point process by defining the conditional density function $f(t_{n+1}|\mathcal{H}(t_n))$, which characterizes the distribution of the next event time given the past history up to the last event time t_n. By the *chain rule* of probability, we know that the joint probability distribution of all event times can be written as

$$f(\cdots, t_1, t_2, \cdots) = \prod_n f(t_n|\cdots, t_{n-2}, t_{n-1}) = \prod_n f(t_n|\mathcal{H}(t_{n-1})).$$

[*] This work is the course project for MAT3300: Mathematical Modeling at the Chinese University of Hong Kong, Shenzhen

[†] School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen

Below we give a simple example of a point process that is specified by giving the distribution of its interarrival time.

Example 1.1 (Renewal Process) *The process is defined by setting the interarrival times to be i.i.d random variables.*

$$f(t_n|\mathcal{H}(t_{n-1})) := g(t_n - t_{n-1}),$$

where $g : (0, \infty) \to [0, 1]$ is a density function. A special case of renewal process is the homogeneous Poisson process with intensity λ, which is a positive constant. In this case, the distribution g follows the exponential distribution with rate λ, i.e. $(t_n - t_{n-1}) \sim \exp(\lambda)$.

Conditional Intensity Function. Another method that is widely used to specify temporal point processes is by defining the *conditional intensity function*. One can easily see from example 1.1 that the interarrival time depends on the past only through the last event time. However, in real world, most of time the assumption fails to hold. Therefore, it turns out the specifying temporal point process by interarrival time is not a good choice. Instead the conditional intensity function would be a more intuitive and convenient way to show dynamic of the process that depends on the past. The conditional intensity function is defined by

$$\lambda^*(t) = \frac{f(t|\mathcal{H}(t_{t_n}))}{1 - F(t|\mathcal{H}(t_{t_n}))}, \tag{1}$$

where $f(t|\mathcal{H}(t_{t_n}))$ is the conditional intensity of the next event time and $F(t|\mathcal{H}(t_{t_n}))$ is the corresponding cumulative distribution function. As suggested by [DVJ+03], the superscript $*$ is used to remind us the conditional intensity depends on the history. Consider an infinitesimal time interval around t, say dt, then the conditional intensity function can be heuristically understood as the expected number of points falling in the small time interval, i.e.

$$\lambda^*(t) = \mathbb{E}[N([t, t + dt])|\mathcal{H}(t-)],$$

where $N(A)$ denotes the number of points falling in the region A and $\mathcal{H}(t-)$ denotes the history up to but not include t.

Different choices of conditional intensities can make the appearence of point processes very flexible. For example, we can alternatively characterize the (inhomogeneous) Poisson process by the number of points in disjoint regions to be *independent*. The specification can be done by defining the conditional intensity to be independent of the past, i.e. the conditional intensity is equal to the intensity of the Poisson process, $\lambda^*(t) = \lambda(t)$. Moreover, the conditional intensity can be chosed to describe triggering and inhibiting behaviors.

Example 1.2 (Hawkes Process[Haw71]) *The conditional intensity is defined to be*

$$\lambda^*(t) = \mu + \alpha \sum_{t_i < t} \exp(-(t - t_i)),$$

where μ and α are positive constants. The key point is, whenever a new event happens, it increases the conditional intensity by the exponential term. As one can imagine, the points from this process tend to cluster together. One remark is the exponential function can be replaced by other more complicated functions to capture complex dynamics as long as the chosen function is non-negative. For example, Neural networks [ME17] and transformers [ZJL+20] are used to specify the triggering term.

From (1), one can see that the conditional intensity is uniquely determined by the conditional density. One interesting result is the reverse also holds under very mild conditions on λ^*, which says that a point process can be uniquely defined by specifying the conditional intensity.

$$f(t|\mathcal{H}_{t_n}) = \lambda^*(t) \exp\left(-\int_{t_n}^{t} \lambda^*(s) ds\right), \tag{2}$$

$$F(t|\mathcal{H}_{t_n}) = 1 - \exp\left(-\int_{t_n}^{t} \lambda^*(s) ds\right). \tag{3}$$

Likelihood and Learning. Given a observation event sequence $\{t_1, t_2, \cdots, t_n\}$ on $[0, T)$ for some given $T > 0$, it can be shown that the likelihood function is given by

$$\mathcal{L} = \left(\prod_{i=1}^{n} \lambda^*(t_i)\right) \exp\left(-\int_0^T \lambda^*(s)ds\right). \tag{4}$$

Suppose the model parameters are unknown, the simplest method for model learning is to treat the negative log-likelihood

$$l(\boldsymbol{\theta}) = \sum_{i=1}^{n} \lambda^* \log \lambda^*(t_i) - \int_0^T \lambda^*(s)ds$$

as the loss funciton, then the estimate for the model parameter $\boldsymbol{\theta}^*$ is

$$\boldsymbol{\theta}* = arg\min l(\boldsymbol{\theta}).$$

Marked Case. Our previous discussions all focus on the *unmarked* case. For the marked case, we need to modify the conditional intensity a little to allow the existence of marks $m \in \mathcal{M}$. We can specify the distribution of the mark associated with the point t by its conditional density function $f^*(m|t) = f(m|t, \mathcal{H}(t-))$. Now, we can define the conditional intensity function for a marked point process as

$$\lambda^*(m, t) := \lambda^*(t) f^*(m|t). \tag{5}$$

In the marked case, given a realization of the process $\{(t_1, m_1), (t_2, m_2), \cdots, (t_n, m_n)\}$ on $[0, T) \times \mathcal{M}$, the likelihood function becomes

$$L = \left(\prod_{i=1}^{n} \lambda^*(t_i, m_i)\right) \exp\left(-\int_{\mathcal{M}} \int_0^T \lambda^*(s, u_i)dsdu\right). \tag{6}$$

The derivation is similar to the unmarked case. Again, for model learning, we can use MLE if the data are fully observable.

1.2 Gibbs Sampling

Gibbs sampling is a special case of the so-called *Metropolis-Hastings* algorithm. The basic idea of MH algorithm is at each step, we move from the current state $\boldsymbol{x}$ to a proposed state $\boldsymbol{x}'$ sampled from a proposal distribution $q(\boldsymbol{x}'|\boldsymbol{x})$. We accept the proposal with an acceptance probability $A(\boldsymbol{x}'|\boldsymbol{x})$. If $\boldsymbol{x}'$ is rejected, we stay at the original state $\boldsymbol{x}$. This procedure constructs a *Markov Chain* whose stationary distribution is the target distribution $p^*(\boldsymbol{x})$. However, although the Markov chain is guaranteed to become stationary in the long-run, it might take a sufficiently large amount of time to reach stationarity. Gibbs sampling solves this problem by leveraging the conditional independence property to automatically construct a good proposal which makes the acceptance rate to be 1 [Mur23].

Blocked Gibbs Sampling. Analogous to coordinate-wise gradient descent method, Gibbs sampling can be slow since it only updates one parameter each time. It can be more efficient if we can sample a group of variables together.

Metropolis-Within-Gibbs. In Gibbs sampling, every time we need to draw samples from the full conditional distribution. However, if the no conjugate prior is assumed. Then the form of the full conditional may not be regular and thus standard sampling methods fail. The idea of Metropolis-Within Gibbs is to apply the Metropolis-Hastings algorithm to sample from the full conditional. However, this is often very slow since one needs to wait for sufficiently long time for the Markov chain to become stationary.

2 Problem Statement and Formulation

Consider three processes: immigration process, infection process, and recovery process, denoted by $A(t)$, $I(t)$, and $R(t)$ respectively. We model all of them by *0-1 two-state temporal logic point processes*, which is a framework suitable to reflect the interdependence between the three processes. See Fig. 1 for an illustration.

2.1 Assumptions

1. All of the processes $A(t)$, $I(t)$, and $R(t)$ can only take values in $\{0, 1\}$.
2. Collections of prespecified logic rules $\mathcal{F}_A = \{f_{A_k}\}_k$, $\mathcal{F}_I = \{f_{I_k}\}_k$, and $\mathcal{F}_R = \{f_{R_k}\}_k$ are given for deducing the processes.
3. At one particular time point, only one event is allowed, i.e. there is NO overlapping events.

2.2 Fully Observable Unmarked Point Processes

We adopt the *temporal logic point processes* [LWZ+20] as the framework for our model. We need to model the dynamics of three interdependent processes in which the occurrence of events in one process will trigger or inhibit the occurrence of events in other processes. Naturally, one might model the processes are *Markov processes*. However, the Markov property is too idealistic and not possible to hold in real problems. Model the problem by temporal logic point processes allows us to take the whole past history into consideration, which is more likely to able to capture the true disease spread dynamics in the real world.

We first consider the simplest *unmarked* case and in later sections extend it to more complicated cases. Suppose the whole past history up to the time horizon $T > 0$ of the model is given as

$$\mathcal{H}(T) := \{A(t_{a_1}), A(t_{a_2}), \cdots, A(t_{a_p}), I(t_{i_1}), I(t_{i_2}), \cdots, I(t_{i_q}), R(t_{r_1}), R(t_{r_2}), \cdots, R(t_{r_s})\},$$

where $p, q, s \in \mathbb{N}$ are the number of events occurred in the processes $A(t)$, $I(t)$, and $R(t)$ up to time T respectively. Moreover, we have

$$0 < t_{a_1} < t_{a_2} < \cdots < t_{a_p} < T;\ 0 < t_{i_1} < t_{i_2} < \cdots < t_{i_q} < T;\ 0 < t_{r_1} < t_{r_2} < \cdots < t_{r_s} < T.$$

Suppose collections of logic rules for deducing events in each process are given

$$\mathcal{F}_A = \{f_{A_k}\}_{k=1}^{n_A}; \quad \mathcal{F}_I = \{f_{I_k}\}_{k=1}^{n_I}; \quad \mathcal{F}_R = \{f_{R_k}\}_{k=1}^{n_R},$$

where $n_A, n_I, n_R \in \mathbb{N}$ are the number of logic rules respectively. Note that for each first-order logic rule, we can rewrite it equivalently as its *clausal form* (e.g. $A \to B \Leftrightarrow \neg A \vee B$.)

$$f(A(t_a), I(t_i), R(t_r), t_a = \tau_a, t_i = \tau_i, t_r = \tau_r).$$

By taking the above stated clausal form, we can define the *formula effect* corresponding to each logic rule f With out loss of generality, assume that rule f is for deducing the process $I(t)$. Then the formula effect can be written as

$$\begin{aligned}&\delta_f(t | t_a = \tau_a, t_r = \tau_r)\\ &:= f(A(t_a), 1 - I(t_i), R(t_r), t_a = \tau_a, t_i = t, t_r = \tau_r) - f(A(t_a), I(t_i), R(t_r), t_a = \tau_a, t_i = t, t_r = \tau_r),\end{aligned} \tag{7}$$

where $1 - I(t_i)$ stands for the *counterfactual*, which takes the idea of counterfactual in causal inference. Therefore, the formula effect can be understood as the strength the rule puts to move the state to its counter state. If $\delta_f > 0$, the formula encourages the state to transit. On the other hand, it disencourages the state to transit. Note that $\delta_f(t | t_a = \tau_a, t_r = \tau_r) \in [-1, 1]$ if *softened temporal relations* are applied. Similarly, we can write down the formula effect if f is used for deducing processes $A(t)$ and $R(t)$ respectively

$$\begin{aligned}&\delta_f(t | t_i = \tau_i, t_r = \tau_r)\\ &:= f(1 - A(t_a), I(t_i), R(t_r), t_a = t, t_i = \tau_i, t_r = \tau_r) - f(A(t_a), I(t_i), R(t_r), t_a = t, t_i = t, t_r = \tau_r),\end{aligned} \tag{8}$$

$$\begin{aligned}&\delta_f(t | t_a = \tau_a, t_i = \tau_i)\\ &:= f(A(t_a), I(t_i), 1 - R(t_r), t_a = \tau_a, t_i = \tau_i, t_r = t) - f(A(t_a), I(t_i), R(t_r), t_a = \tau_a, t_i = \tau_i, t_r = t).\end{aligned} \tag{9}$$

Once we have obtained the formula effect of each rule $f \in \mathcal{F}_I$, we can define the *conditional transition* *intensity* $\lambda_I(t)$ for $I(t)$ from state 0 to 1, contributed by the collection of rules $\mathcal{F}_I$. For simplicity of notation, we have omitted the $*$ in the conditional intensity function if fortunately no confusion exists. But one should always remember the dependence on the past history of the conditional intensity.

$$\lambda_I(t) := \exp\left(\sum_{j=1}^{n_I} \omega_{f_{I_j}} \cdot \left(\sum_{\tau_a \in \mathcal{H}_A(t)} \sum_{\tau_r \in \mathcal{H}_R(t)} \delta_{f_{I_j}}(t|t_a = \tau_a, t_r = \tau_r)\right) + b_I(t)\right), \tag{10}$$

where $\omega_{f_{I_j}} \in [0,1]$ is the *rule weight* assigned to f_{I_j} that can be viewed as the confidence level of the rule and $b_I(t)$ is the *base term*, which captures the spontaneous transition dynamics of $I(t)$.

By definition of conditional intensity function, for the process $I(t)$, given a realization $\{I(t_{i_k})\}_{k=1}^q$ of the process up to time T, the *likelihood* $\mathcal{L}(\{I(t_{i_k})\})_{k=1}^q$ can be written as

$$\mathcal{L}(\{I(t_{i_k})\})_{k=1}^q = \prod_{j=1}^q \left(\lambda_I(t_{i_j})\right) \cdot \exp\left(-\int_0^T \lambda_I(s)ds\right). \tag{11}$$

If the realization of the processes $A(t)$ and $R(t)$ are $\{A(t_{a_k})\}_{k=1}^p$ and $\{R(t_{r_k})\}_{k=1}^s$ respectively, then follow the above-mentioned procedure, we can similarly derive the likelihoods for $A(t)$ and $R(t)$ as

$$\mathcal{L}(\{A(t_{i_k})\})_{k=1}^p = \prod_{j=1}^p \left(\lambda_A(t_{i_j})\right) \cdot \exp\left(-\int_0^T \lambda_A(s)ds\right). \tag{12}$$

and

$$\mathcal{L}(\{R(t_{i_k})\})_{k=1}^s = \prod_{j=1}^s \left(\lambda_R(t_{i_j})\right) \cdot \exp\left(-\int_0^T \lambda_R(s)ds\right). \tag{13}$$

Therefore, by *superposition principle*, we can compute the *complete likelihood* for the model

$$\mathcal{L}(\mathcal{H}(T)) = \mathcal{L}(\{A(t_{i_k})\})_{k=1}^p \cdot \mathcal{L}(\{I(t_{i_k})\})_{k=1}^q \cdot \mathcal{L}(\{R(t_{i_k})\})_{k=1}^s.$$

With our proposed conditional intensity function, predication for the occurrence time of the next event is given by

$$\begin{aligned} p(t|\mathcal{H}(t)) &= \lambda(t)\exp\left(\int_{t_j}^t \lambda(s)ds\right), \\ \mathbb{E}[t_{j+1}] &= \int_{t_j}^\infty t \cdot p(t|\mathcal{H}(t))dt, \\ \hat{k}_{j+1} &= arg \max_{i \in \{A,I,R\}} \frac{\lambda_I(t_{j+1})}{\lambda_i(t_{j+1})}, \end{aligned} \tag{14}$$

where $\lambda(t) := (\lambda_A + \lambda_I + \lambda_R)(t)$.

We summarize the model parameters, which contain the rule weights and the parameters parameterizing the base term functions, as $\boldsymbol{\theta}$. We treat our model parameters as unknown then the complete likelihood becomes a function of $\boldsymbol{\theta}$. We can use *data-driven techniques* to learn the model parameter by maximizing the likelihood.

2.3 Fully Observable Marked Point Processes

In the last section, we have not taken event marks into consideration. However, in real problems, it would be sensible to consider assigning each arrival in any one of the processes a mark that specifies the characteristics of the arrival (e.g. diseases type, age, gender). In this section, we would assume all the processes $A(t)$, $I(t)$, and $R(t)$ are *marked* temporal logic point processes with the mark space $\mathcal{M} := \mathcal{D} \times \mathcal{R}$, where $\mathcal{D} = \{\boldsymbol{d}^{(1)}, \boldsymbol{d}^{(2)}, ..., \boldsymbol{d}^{(m)}\}$ contains m possible disease types of interest and $\mathcal{R} \subset \mathbb{R}$ denotes range of age of the

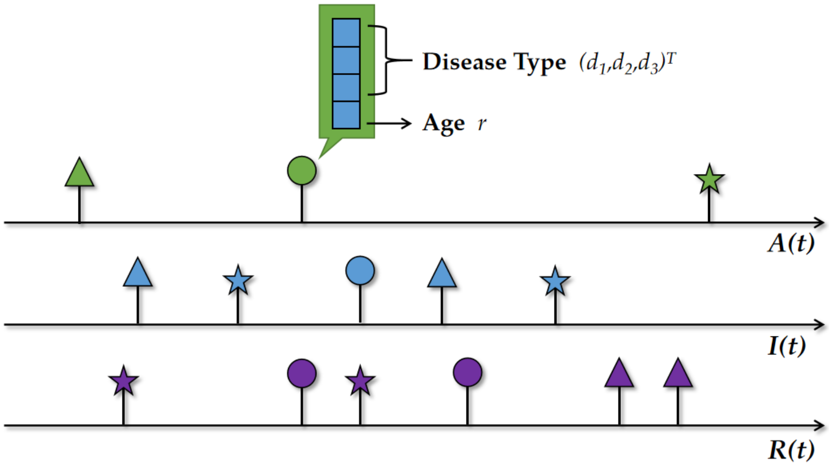

Figure 1: Illustration of the stochastic disease spread model. $A(t), I(t)$, and $R(t)$ denote the immigration process, infection process, and recovery process respectively. The shapes (circle, triangle, star) represent different marks. In this case, only three different disease types $d_1, d_2, d_3 \in \mathcal{D}$ are considered. $r \in \mathbb{R}$ denotes the age of the individual.

arrival person. Fig. 1 is a simple illustration of the model. To obtain the conditional intensity function for each process, we need to specify the distribution of marks $f^*(\boldsymbol{m}|t) := f^*(\boldsymbol{d}, r|t)$, where $d \in \mathcal{D}$ and $r \in \mathcal{R}$. For simplicity, we assume that the probabilities for diseases $\boldsymbol{p} := (p_1, p_2, \cdots, p_m) \in \Delta_m$ follows a Dirichlet distribution with parameters $(\alpha_1, \alpha_2, \cdots, \alpha_m)$ and r follows a normal distribution with mean μ and standard deviation σ, we assume a conjugate normal prior on $\mu \sim \mathcal{N}(\mu_0, \sigma^2/\kappa_0)$ and an inverse-gamma prior on $\sigma^2 \sim IG(v_0/2, v_0\sigma_0^2/2)$. We can set the prior sample size κ_0 and v_0 to be small to ensure that the priors are weakly-informative.

$$p(\boldsymbol{p}|\boldsymbol{\alpha}) = \frac{\Gamma(\sum_{i=1}^{m}(\alpha_i))}{\prod_{i=1}^{m}\Gamma(\alpha_i)} \prod_{i=1}^{m} p_i^{\alpha_i - 1} \propto p_1^{\alpha_1-1} p_2^{\alpha_2-1} \cdots p_m^{\alpha_m-1}. \tag{15}$$

$$p(\mu|\mu_0, \sigma, \kappa_0) = (2\pi\sigma^2/\kappa_0)^{-1/2} \cdot \exp\left(-\frac{\kappa_0}{2\sigma^2}(\mu - \mu_0)^2\right). \tag{16}$$

$$p(\sigma^2|v_0, \sigma_0^2) = \frac{(v_0\sigma_0^2/2)^{v_0/2}}{\Gamma(v_0/2)} \cdot (\sigma^2)^{-v_0/2-1} \cdot \exp\left(-\frac{v_0\sigma_0^2}{2\sigma^2}\right). \tag{17}$$

$$p(r|\mu, \sigma) = (2\pi\sigma^2)^{-1/2} \cdot \exp\left(\frac{1}{2\sigma^2}(r - \mu)^2\right) \propto (\sigma^2)^{-1/2} \cdot \exp\left(\frac{1}{2\sigma^2}(r - \mu)^2\right). \tag{18}$$

We can choose the prior parameters such that $\mathbb{P}(r \in [\mu - 2\sigma, \mu + 2\sigma]) = \mathbb{P}([0, 80]) \approx 0.95$. Therefore, we can ignore people with negative age and age larger than 80. We further assume statistical independence of d and r, then

$$\lambda^*(\boldsymbol{m}, t) = \lambda^*(t) f^*(\boldsymbol{m}|t) = \lambda^*(t) f^*(\boldsymbol{d}|t) f^*(r|t) = \lambda^*(t) \cdot \left(\prod_{i=1}^{m} p_i^{\mathbb{I}(\boldsymbol{d}=\boldsymbol{d}^{(i)})}\right) \cdot (2\pi\sigma^2)^{-1/2} \exp\left(\frac{1}{2\sigma^2}(r - \mu)^2\right), \tag{19}$$

where $\lambda^*(t)$ is given by (10) if we assume the logic rules for deducing $A(t)$, $I(t)$, and $R(t)$ remain unchanged.

We denote the model parameter by $\boldsymbol{\theta} := (\boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B})$, where $\boldsymbol{\omega}$ is a high-dimensional vector containing all the rule weights of a specific process and $\boldsymbol{B}$ is a vector containing all the base terms $b(t)$. We adopt the Bayesian framework and treat $\boldsymbol{\theta}$ as latent variables, the model learning problem can be equivalently viewed as an *inference problem*. Since exact inference for this system is intractable, we would instead choose to use

Markov Chain Monte Carlo (MCMC) to conduct approximate inference. We assume independent priors on $\boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R$, and $\boldsymbol{B}$.

$$\begin{aligned}
\boldsymbol{\omega}_A &\sim \mathit{Dirichlet}(\beta_A^{(1)}, \beta_A^{(2)}, \cdots, \beta_A^{(|\boldsymbol{\omega}_A|)}),\\
\boldsymbol{\omega}_I &\sim \mathit{Dirichlet}(\beta_I^{(1)}, \beta_I^{(2)}, \cdots, \beta_I^{(|\boldsymbol{\omega}_I|)}),\\
\boldsymbol{\omega}_R &\sim \mathit{Dirichlet}(\beta_R^{(1)}, \beta_R^{(2)}, \cdots, \beta_R^{(|\boldsymbol{\omega}_R|)}).
\end{aligned}$$

and

$$\boldsymbol{B} \sim \mathcal{N}(\theta_0, \tau_0^2).$$

The posterior distribution can be written as

$$\begin{aligned}
p(\boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2 | \boldsymbol{x}) &\propto p(\boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2) \cdot p(\boldsymbol{x} | \boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2)\\
&= p(\boldsymbol{\omega}_A) p(\boldsymbol{\omega}_I) p(\boldsymbol{\omega}_R) p(\boldsymbol{B}) p(\boldsymbol{p}) p(\mu|\sigma^2) p(\sigma^2) \cdot p(\boldsymbol{x} | \boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2),
\end{aligned} \tag{20}$$

where $\boldsymbol{x} := \{(t_1, m_1), (t_2, m_2), \cdots, (t_n, m_n)\}$ stands for the observation. The first distribution on the RHS can be explicitly computed if the independence of all the latent variables are assumed. The second term on the RHS also has closed-form expression since it is plainly the likelihood of $\boldsymbol{x}$.

$$p(\boldsymbol{x} | \boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2) = \left(\prod_{i=1}^{n} \lambda^*(t_i, m_i) \right) \exp\left(- \sum_{i=1}^{m} \int_{\mathbb{R}} \int_0^T \lambda^*(s, r, u_i) ds dr \right). \tag{21}$$

[Ras13] presents how to conduct Bayesian inference for fully observable Hawkes processes. Indeed, the procedure introduced can be extended to our case. The idea is to use *blocked Gibbs sampling* to sample from full conditionals sequentially based on the most up-to-date information. However, the prior defined here is not conjugate so that sampling from the full conditional is still not easy. Therefore, we use *Metropolis-Within-Gibbs* to sample from the full conditionals. If the reader is not familiar with MCMC methods, it is suggested to refer to section 1.2, where we provide a brief introduction to the methods we are going to use.

We demonstrate how the procedure for sampling from the full conditional of $\boldsymbol{\omega}_A$, the procedure for sampling from the full conditionals of $\boldsymbol{\omega}_I, \boldsymbol{\omega}_R$, and $\boldsymbol{B}$ can be done in a similar way. Set the proposal distribution for the Metropolis algorithm to be $q(\boldsymbol{\omega}'_A | \boldsymbol{\omega}_A)$ (the choice of the proposal q can be very flexible. For example, the proposal can be data-driven [ORBD05] or adaptive, which means the proposal can change its parameters based on the progress of the procedure [CMP08],[HST01]). Suppose we have proposed a new state $\boldsymbol{\omega}'_A$ from q. Immediately, we can write down the expression of the acceptation probability

$$A(\boldsymbol{\omega}'_A | \boldsymbol{\omega}_A) = \min(1, \alpha(\boldsymbol{\omega}'_A | \boldsymbol{\omega}_A)),$$

where

$$\begin{aligned}
\alpha(\boldsymbol{\omega}'_A | \boldsymbol{\omega}_A) &= \frac{p(\boldsymbol{\omega}'_A | \boldsymbol{x}, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2) q(\boldsymbol{\omega}_A | \boldsymbol{\omega}'_A)}{p(\boldsymbol{\omega}_A | \boldsymbol{x}, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2) q(\boldsymbol{\omega}'_A | \boldsymbol{\omega}_A)} \propto \frac{p(\boldsymbol{\omega}'_A) p(\boldsymbol{x} | \boldsymbol{\omega}'_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2) q(\boldsymbol{\omega}_A | \boldsymbol{\omega}'_A)}{p(\boldsymbol{\omega}_A) p(\boldsymbol{x} | \boldsymbol{\omega}_A, \boldsymbol{x}, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2) q(\boldsymbol{\omega}'_A | \boldsymbol{\omega}_A)}\\
&= \frac{p(\boldsymbol{\omega}'_A) q(\boldsymbol{\omega}_A | \boldsymbol{\omega}'_A)}{p(\boldsymbol{\omega}_A) q(\boldsymbol{\omega}'_A | \boldsymbol{\omega}_A)} \cdot \left(\prod_{i=1}^{n} \frac{\lambda'^*(t_i, m_i)}{\lambda^*(t_i, m_i)} \right) \exp\left(- \sum_{i=1}^{m} \left[\int_{\mathbb{R}} \int_0^T (\lambda'^*(s, r, u_i) - \lambda^*(s, r, u_i)) ds dr \right] \right),
\end{aligned} \tag{22}$$

where $\lambda'^*(t_i, m_i)$ is the same as $\lambda^*(t_i, m_i)$ except $\boldsymbol{\omega}_A$ is replaced by $\boldsymbol{\omega}'_A$.

The full conditional of $\boldsymbol{p}$ is easier to derive since we assume a Dirichlet conjugate prior. By conditional independence, given the observation $\boldsymbol{x}$, $\boldsymbol{p}$ is conditionally independent with $\boldsymbol{\omega}_A$, $\boldsymbol{\omega}_I$, $\boldsymbol{\omega}_R$, $\boldsymbol{B}$, μ and σ^2. Therefore, the full conditional distribution of $\boldsymbol{p}$ is

$$\begin{aligned}
p(\boldsymbol{p} | \boldsymbol{x}, \boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \mu, \sigma^2) = p(\boldsymbol{p} | \boldsymbol{x}) &= \frac{\Gamma(\sum_{i=1}^{m} \alpha_i + \sum_{k=1}^{n} \mathbb{I}(d_k = d^{(i)}) - 1)}{\prod_{i=1}^{m} \Gamma(\alpha_i + \sum_{k=1}^{n} \mathbb{I}(d_k = d^{(i)}) - 1)} \prod_{i=1}^{m} p_i^{\alpha_i + \sum_{k=1}^{n} \mathbb{I}(d_k = d^{(i)}) - 1}\\
&\propto \prod_{i=1}^{m} p_i^{\alpha_i + \sum_{k=1}^{n} \mathbb{I}(d_k = d^{(i)}) - 1}.
\end{aligned} \tag{23}$$

That is to say

$$\boldsymbol{p}|\boldsymbol{x},\boldsymbol{\omega}_A,\boldsymbol{\omega}_I,\boldsymbol{\omega}_R,\boldsymbol{B},\mu,\sigma^2 \sim Dirichlet(\alpha_1+\sum_{k=1}^{n}\mathbb{I}(d_k=d^{(1)}),\alpha_2+\sum_{k=1}^{n}\mathbb{I}(d_k=d^{(2)}),\cdots,\alpha_m+\sum_{k=1}^{n}\mathbb{I}(d_k=d^{(m)})).$$

Similarly, the posterior for μ and σ^2 can be derived as follows

$$\begin{aligned} &p(\mu,\sigma^2|\boldsymbol{x},\boldsymbol{\omega}_A,\boldsymbol{\omega}_I,\boldsymbol{\omega}_R,\boldsymbol{B},\boldsymbol{p}) = p(\mu,\sigma^2|\boldsymbol{x}) = p(\mu|\boldsymbol{x},\sigma^2)p(\sigma^2|\boldsymbol{x}) \\ &\propto (\sigma^2)^{-1/2}\exp\left(-\frac{\kappa_n}{2\sigma^2}(\mu-\mu_n)\right)\cdot(\sigma^2)^{-v_n/2-1}\cdot\exp\left(-\frac{v_n\sigma_n^2}{2\sigma^2}\right), \end{aligned} \tag{24}$$

where $\kappa_n=\kappa_0+n$, $v_n=v_0+n$, $\mu_n=(\kappa_0\mu_0+n\bar{r})/(\kappa_n)$, $\sigma_n^2=(v_0\sigma_0^2+\sum_{i=1}^{n}(r_i-\bar{r})^2+n\kappa_0/\kappa_n\cdot(\bar{r}-\mu_0)^2)/v_n$, and $\bar{r}=\sum_{i=1}^{n}r_i/n$.

Since we have derived the full conditional distributions for all the latent variables, we can adopt the *blocked Gibbs sampling* algorithm to sequentially sample $\boldsymbol{x},\boldsymbol{\omega}_A,\boldsymbol{\omega}_I,\boldsymbol{\omega}_R,\boldsymbol{B}$, and $\boldsymbol{p}$. The pseudo-code is presented in algorithm 1.

Algorithm 1 Blocked Gibbs Sampling for Fully Observable Marked Temporal Logic Point Processes

Input: Observation $\boldsymbol{x}$; initial guess $\boldsymbol{\omega}_A^{(0)},\boldsymbol{\omega}_I^{(0)},\boldsymbol{\omega}_R^{(0)},\boldsymbol{B}^{(0)}$; number of iterations S

Output: Posterior distribution $p(\boldsymbol{\omega}_A,\boldsymbol{\omega}_I,\boldsymbol{\omega}_R,\boldsymbol{B},\boldsymbol{p}|\boldsymbol{x})$

```
For $i$ from 1 up to $S$ do:
    $\boldsymbol{p}^{(i)} \leftarrow p(\boldsymbol{p}|\boldsymbol{x})$    # Sample from Dirichlet distribution
    $\sigma^{2(i)} \leftarrow p(\sigma^2|\boldsymbol{x})$    # Sample from inverse-gamma distribution
    $\mu^{(i)} \leftarrow p(\mu|\boldsymbol{x},\sigma^2)$    # Sample from normal distribution
    $\boldsymbol{\omega}_A^{(i)} \leftarrow p(\boldsymbol{\omega}_A|\boldsymbol{x},\boldsymbol{\omega}_I^{(i-1)},\boldsymbol{\omega}_R^{(i-1)},\boldsymbol{B}^{(i-1)},\boldsymbol{p}^{(i)},\mu^{(i)},\sigma^{2(i)})$    # Metropolis-Within-Gibbs
    $\boldsymbol{\omega}_I^{(i)} \leftarrow p(\boldsymbol{\omega}_I|\boldsymbol{x},\boldsymbol{\omega}_A^{(i)},\boldsymbol{\omega}_R^{(i-1)},\boldsymbol{B}^{(i-1)},\boldsymbol{p}^{(i)},\mu^{(i)},\sigma^{2(i)})$    # Metropolis-Within-Gibbs
    $\boldsymbol{\omega}_R^{(i)} \leftarrow p(\boldsymbol{\omega}_R|\boldsymbol{x},\boldsymbol{\omega}_A^{(i)},\boldsymbol{\omega}_I^{(i)},\boldsymbol{B}^{(i-1)},\boldsymbol{p}^{(i)},\mu^{(i)},\sigma^{2(i)})$    # Metropolis-Within-Gibbs
    $\boldsymbol{B}^{(i)} \leftarrow p(\boldsymbol{B}|\boldsymbol{x},\boldsymbol{\omega}_A^{(i)},\boldsymbol{\omega}_I^{(i)},\boldsymbol{\omega}_R^{(i)},\boldsymbol{p}^{(i)},\mu^{(i)},\sigma^{2(i)})$    # Metropolis-Within-Gibbs
    Save $(\boldsymbol{\omega}_A^{(i)},\boldsymbol{\omega}_I^{(i)},\boldsymbol{\omega}_R^{(i)},\boldsymbol{B}^{(i)},\boldsymbol{p}^{(i)},\mu^{(i)},\sigma^{2(i)})$
End For
```

2.4 Marked Point Processes with Missing Events

Previously in section 2.3, we consider fully observable marked point processes model in which we assume all the arrival times in all processes are recorded and the associated marks are also recorded. However, this assumption is highly improbable in the real world. For pandemics like COVID-19, it is really hard to record precisely the time when an infectious individual move into another region, when an susceptible individual gets infected, and when one patient gets recovered. Even though we know the arrival times, it is still not likely that we will also know the disease type and the person's age. In real problems, many phenomenons are caused by some other *unobserved* events. Imputing missing data is very important and useful in a range of real application problems such as data analysis of social media[SRdS+19], health care [PA12], and recommendation system [MZRS11].

Latent Markers. It is very likely that we have the record for all the arrivals but the markers (e.g. disease type and age) are missing. This is reasonable in reality because one city can record the times of the arrivals of external immigrants but it is impossible to know their ages and if someone carries viruses. Moreover, it is also possible that the records show some people get infected at certain time points but we have no idea what are their ages and disease types before they are diagnosed by doctors. In our treatment, we can model this type of missing data as *latent markers*. There might be some arrivals in the processes $A(t), I(t)$, and $R(t)$ that do not carry the information of markers, see Fig. 2 for an illustration.

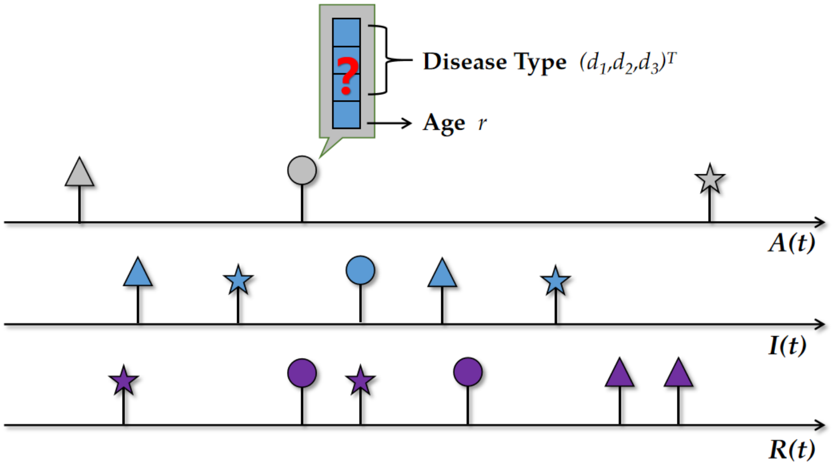

Figure 2: The case of missing markers. The gray color denotes that the arrival is observed but the marker is unobserved.

The missing data can be easily imputed in our model. Note that we have assumed the conditional density function for the marker given time t is

$$f^*(\boldsymbol{m}|t) = f^*(\boldsymbol{d}|t)f^*(r|t) = \left(\prod_{i=1}^{m} p_i^{\mathbb{I}(d=d^{(i)})}\right) \cdot (2\pi\sigma^2)^{-1/2} \exp\left(\frac{1}{2\sigma^2}(r-\mu)^2\right).$$

Since the conditional density functions for disease type $\boldsymbol{d} \in \mathcal{D}$ and age $r \in \mathcal{R}$ do not depend on the current time t, the time of the arrival that has missing marker does not matter. Because there are fully observed data in hand, we can update our belief on the distributions of disease type $f^*(\boldsymbol{d}|t)$ and age $f^*(r|t)$. Suppose the fully observed data is given by

$$\mathcal{S} := \{(\boldsymbol{d}_1, r_1), (\boldsymbol{d}_2, r_2), \cdots, (\boldsymbol{d}_n, r_n)\},$$

where we omit the arrival times and only consider the marker information of those fully observed events. Then, the posterior distributions for disease type and age are

$$\begin{aligned} &\boldsymbol{d}|t, \mathcal{S}, \boldsymbol{p} \sim Multinomial(1, \boldsymbol{p}), \\ &\boldsymbol{p}|\mathcal{S} \sim Dirichlet(\alpha_1 + \sum_{k=1}^{n} \mathbb{I}(\boldsymbol{d}_k = \boldsymbol{d}^{(1)}), \alpha_2 + \sum_{k=1}^{n} \mathbb{I}(\boldsymbol{d}_k = \boldsymbol{d}^{(2)}), \cdots, \alpha_m + \sum_{k=1}^{n} \mathbb{I}(\boldsymbol{d}_k = \boldsymbol{d}^{(m)})). \end{aligned}$$

and

$$r|\mu,\sigma \sim \mathcal{N}(\mu,\sigma^2), \quad \mu|\sigma \sim \mathcal{N}(\mu_n, \sigma^2/\kappa_n); \quad \sigma^2 \sim IG(v_n/2, v_n\sigma_n^2/2),$$

where

$$\kappa_n = \kappa_0 + n, \quad v_n = v_0 + n; \quad \mu_n = (\kappa_0\mu_0 + n\bar{r})/\kappa_n, \quad \sigma_n^2 = (v_0\sigma_0^2 + \sum_{i=1}^{n}(r_i - \bar{r})^2 + \kappa_0 n/\kappa_n(\bar{r} - \mu_0)^2)/v_n$$

and $\bar{r} = 1/n \cdot \sum_{i=1}^{n} r_i$ denotes the sample mean of r. Therefore, to impute the missing markers, we only need to construct some estimators (e.g. MAP, mean, median) based on the posterior distributions. Indeed, if there are significant difference between different processes, it would be better to obtain the posterior distributions only based on the observed data from that specific process.

Latent Processes. This is one of the most interesting and also the most challenging problems we want to deal with. For example, in the disease spread case, during some time interval, it is possible that the we lose the records of those people who are infected, i.e. the whole $I(t)$ process is latent within the time interval $[0, T)$. In this case, since the data is incomplete, we cannot learn the model simply by maximizing the log-likelihood $\log p_\theta(\boldsymbol{x})$. The unique challenge of latent temporal logic point processes is that the latent variables take the form of a temporal logic point process. Since the number of latent events is undetermined. The Monte-Carlo method would be suitable for simulating events with unknown numbers [LWB17]. For latent variable problem, we need to jointly perform inference for hidden variables and learn the model parameters. Actually, this can be done through first imputing the missing events condition on the observed data.

$$\boldsymbol{x}_{mis} \sim p(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs}, \boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2),$$

then we successfully complete the dataset so we can continue to use the Gibbs sampler to iteratively sample from full conditionals, until the simulated Markov chain becomes stationary, the samples drawn by Gibbs sampler is exactly samples from the true posterior distribution $p(\boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2|\boldsymbol{x})$. The pseudo-code is shown in algorithm 2.

Algorithm 2 Blocked Gibbs Sampling for Marked Temporal Logic Point Processes with Missing Data

```
Input: Incomplete data $\boldsymbol{x}_{obs}$; initial guess $\boldsymbol{\omega}_A^{(0)}, \boldsymbol{\omega}_I^{(0)}, \boldsymbol{\omega}_R^{(0)}, \boldsymbol{B}^{(0)}$,; number of iterations $S$
Output: Posterior distribution $p(\boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}|\boldsymbol{x})$
$\boldsymbol{p} \leftarrow Dirichlet(\boldsymbol{\alpha})$
$\sigma^2 \sim IG(v_0/2, v_0\sigma_0^2/2)$
$\mu \sim \mathcal{N}(\mu_0, \sigma^2/\kappa_0)$
For $i$ from 1 up to $S$ do:
    $\boldsymbol{x}_{mis} \leftarrow p(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs}, \boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2)$    # The most difficult part, see algorithm 3
    $\boldsymbol{x} \leftarrow [\boldsymbol{x}_{obs}, \boldsymbol{x}_{mis}]$
    $\boldsymbol{p}^{(i)} \leftarrow p(\boldsymbol{p}|\boldsymbol{x})$    # Sample from Dirichlet distribution
    $\sigma^{2(i)} \leftarrow p(\sigma^2|\boldsymbol{x})$    # Sample from inverse-gamma distribution
    $\mu^{(i)} \leftarrow p(\mu|\boldsymbol{x}, \sigma^2)$    # Sample from normal distribution
    $\boldsymbol{\omega}_A^{(i)} \leftarrow p(\boldsymbol{\omega}_A|\boldsymbol{x}, \boldsymbol{\omega}_I^{(i-1)}, \boldsymbol{\omega}_R^{(i-1)}, \boldsymbol{B}^{(i-1)}, \boldsymbol{p}^{(i)}, \mu^{(i)}, \sigma^{2(i)})$    # Metropolis-Within-Gibbs
    $\boldsymbol{\omega}_I^{(i)} \leftarrow p(\boldsymbol{\omega}_I|\boldsymbol{x}, \boldsymbol{\omega}_A^{(i)}, \boldsymbol{\omega}_R^{(i-1)}, \boldsymbol{B}^{(i-1)}, \boldsymbol{p}^{(i)}, \mu^{(i)}, \sigma^{2(i)})$    # Metropolis-Within-Gibbs
    $\boldsymbol{\omega}_R^{(i)} \leftarrow p(\boldsymbol{\omega}_R|\boldsymbol{x}, \boldsymbol{\omega}_A^{(i)}, \boldsymbol{\omega}_I^{(i)}, \boldsymbol{B}^{(i-1)}, \boldsymbol{p}^{(i)}, \mu^{(i)}, \sigma^{2(i)})$    # Metropolis-Within-Gibbs
    $\boldsymbol{B}^{(i)} \leftarrow p(\boldsymbol{B}|\boldsymbol{x}, \boldsymbol{\omega}_A^{(i)}, \boldsymbol{\omega}_I^{(i)}, \boldsymbol{\omega}_R^{(i)}, \boldsymbol{p}^{(i)}, \mu^{(i)}, \sigma^{2(i)})$    # Metropolis-Within-Gibbs
    Save $(\boldsymbol{\omega}_A^{(i)}, \boldsymbol{\omega}_I^{(i)}, \boldsymbol{\omega}_R^{(i)}, \boldsymbol{B}^{(i)}, \boldsymbol{p}^{(i)}, \mu^{(i)}, \sigma^{2(i)})$
End For
```

However, the posterior predictive distribution $p(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs}, \boldsymbol{\omega}_A, \boldsymbol{\omega}_I, \boldsymbol{\omega}_R, \boldsymbol{B}, \boldsymbol{p}, \mu, \sigma^2)$ does not have a regular form and we even do not know in what space the distribution is defined since $\boldsymbol{x}_{mis}$ do not have a fixed dimension. Since exactly computing the distribution is intractable, we resort the approximation inference algorithm. There are a number of different approximate inference techiniques like *Variational Inference* ([BKM17],[HBWP13], [RGB14]), Markov Chain Monte Carlo [BGJM11], and Sequential Monte Carlo ([DFG01], [NLRB18]). We will utilize the idea of variational inference and sequential Monte Carlo to construct and train an approximate posterior distribution.

We proposal to train a variational posterior $q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})$ to approximate the ground truth posterior. We discretize the time interval into fine scaffolds $\{[s_{i-1}, s_i)\}_{i=1}^M$ where $s_0 = 0, s_M = T$ and $s_{i-1} < s_i, \forall i = 1, 2, ..., M$. We choose $\max_{i \in \{1,2,...,M\}}(s_i - s_{i-1})$ to be small enough so that inside each interval, there is at most one transition. (i.e., we discretize the infection process.) According to chain rule we have

$$q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs}) = q_\psi(\{z_m\}_{m=1}^M|\boldsymbol{x}_{obs}) = q_\psi(z_1|\boldsymbol{x}_{obs})q_\psi(z_2|z_1,\boldsymbol{x}_{obs})\cdots q_\psi(z_M|z_{1:M-1},\boldsymbol{x}_{obs}),$$

where z_m stands for an infection arrival (note that there can also be no event) in the time interval $[s_{m-1}, s_m)$. We can use two LSTMs to encode the information in the observation $\boldsymbol{x}_{obs}$ and the information contained in $z_{1:i}$ [MQE19], say, $f_1(\boldsymbol{x}_{obs}) = \boldsymbol{h}^o$ and $f_2(z_{1:i}) = \boldsymbol{h}_i^I$. Once we have a new sample z_{i+1}, we update the encoding by $\boldsymbol{h}_{i+1}^I := f_2(\boldsymbol{h}_i^I, z_{i+1})$. Finally, both of the encodings are mapped to a Bernoulli distribution, that is, they are mapped to a probability $p \in [0,1]$. Then, We can train the LSTMs by minimizing the KL-divergence between the true posterior and the proposal

$$\min_\psi \mathcal{D}_{KL}(q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})||p(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs},\boldsymbol{\omega}_A,\boldsymbol{\omega}_I,\boldsymbol{\omega}_R,\boldsymbol{B},\boldsymbol{p},\mu,\sigma^2)),$$

which is equivalent to maximizing the Evidence Lower Bound (ELBO)

$$\max_\psi \mathbb{E}_{q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_{obs},\boldsymbol{x}_{mis})\right] + \mathbb{H}[q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})],$$

the second term on the RHS denotes the entropy of the variational posterior. The objective is almost VAE-like [KW13] except in our treatment the model parameter is clamped. If we assume the variational posterior is reparameterizable (it can always be made to be the case by choosing the variational posterior to be flexible enough), i.e., there exists a differentiable mapping $g(\epsilon;\psi) : \epsilon \mapsto \boldsymbol{x}_{mis}$ and $\epsilon \sim s(\epsilon)$ has a tractable distribution, then the objective can be rewritten as

$$\max_\psi \left(\mathbb{E}_{s(\epsilon)}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_{obs}, g(\epsilon;\psi))\right] + \mathbb{H}\left[q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})\right]\right).$$

Since we can draw samples from the proposal by sampling from $s(\epsilon)$, say, we have obtained L samples from the variational posterior

$$\{\boldsymbol{x}_{mis}^{(l)}\}_{l=1}^L \sim q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs}).$$

then we have a Monte-Carlo estimate of the ELBO.

$$\mathcal{L}(\boldsymbol{\theta},\psi,\boldsymbol{x}_{obs}) \approx \frac{1}{L}\sum_{l=1}^L \left(\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_{mis}^{(l)},\boldsymbol{x}_{obs}) - \log q_\psi(\boldsymbol{x}_{mis}^{(l)}|\boldsymbol{x}_{obs})\right).$$

or,

$$\mathcal{L}(\boldsymbol{\theta},\psi,\boldsymbol{x}) \approx \frac{1}{L}\sum_{l=1}^L \left(\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_{mis}^{(l)},\boldsymbol{x}_{obs})\right) + \mathbb{H}\left[q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})\right],$$

since we can explicitly compute the entropy.

$$\begin{aligned}\mathbb{H}\left[q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})\right] &= -\int \log q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})\cdot q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})d\boldsymbol{x}_{mis} \\ &= -\int \sum_{i=1}^M \log q_\psi(z_i|\boldsymbol{h}^o,\boldsymbol{h}_{i-1}^I)\cdot\prod_{i=1}^M q_\psi(z_i|\boldsymbol{h}^o,\boldsymbol{h}_{i-1}^I)dz_{1:M} \\ &= -\sum_{i=1}^M \int \log q_\psi(z_i|\boldsymbol{h}^o,\boldsymbol{h}_{i-1}^I)\cdot q_\psi(z_i|\boldsymbol{h}^o,\boldsymbol{h}_{i-1}^I)dz_i \\ &= \sum_{i=1}^M \mathbb{H}[q_\psi(z_i|\boldsymbol{h}^o,\boldsymbol{h}_{i-1}^I)].\end{aligned} \tag{25}$$

Differentiate the RHS w.r.t. ψ, we can perform gradient ascent to jointly optimize the variational posterior. The pseudo code is as follows:

Algorithm 3 Train Variational Posterior

```
Input: Dataset $\{\boldsymbol{x}^n_{obs}\}_{n=1}^N$; initial guess $\psi^{(0)} \leftarrow$ RANDOM(); batch size $R$
Output: Variational Posterior $q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})$
While $\psi$ does not converge do:
    $\{\boldsymbol{x}^{(r)}_{obs}\}_{r=1}^R \leftarrow$ Random minibatch of $R$ action realizations
    $\boldsymbol{\epsilon} \leftarrow$ Random samples from the noise distribution $p(\boldsymbol{\epsilon})$
    $\boldsymbol{g} \leftarrow \nabla_\psi \frac{N}{R} \sum_{r=1}^R \mathcal{L}(\boldsymbol{\theta}, \psi, \boldsymbol{x}^{(r)}_{obs})$
    $\psi \leftarrow$ GRADIENTDESCENT($\boldsymbol{g}$)
```

Gumbel-Max Trick. Note that when we construct a Monte-Carlo estimator for the ELBO objective, we involve a *sample* operation in the computation graph $\boldsymbol{x}_{mis} \sim q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})$, which impedes the gradient to propagate backward. Recall that

$$q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs}) = q_\psi(\{z_m\}_{m=1}^M|\boldsymbol{x}_{obs}) = q_\psi(z_1|\boldsymbol{x}_{obs})q_\psi(z_2|z_1, \boldsymbol{x}_{obs}) \cdots q_\psi(z_M|z_{1:M-1}, \boldsymbol{x}_{obs}),$$

where each $q_\psi(z_{i+1}|z_{1:i}, \boldsymbol{x}_{obs})$ is the probability mass function of a discrete random variable. We can use the *Gumbel-Max Trick* [JGP16], which is a kind of *reparameterization* trick, to remove the stochastic node out of the computation graph and thus we can train the variational posterior end-to-end. Suppose for a fixed choice of j, the sample space of z_j is

$$\left\{ \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \cdots, \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \right\} \subset \Delta^I,$$

where I is the number of latent processes. The corresponding probabilities are

$$\{\pi_0, \pi_1, \pi_2, ..., \pi_I\}, \quad \sum_{k=0}^I \pi_k = 1.$$

each of the probability π_j depends on the variational parameter ψ. Let $g_0, g_1, g_2, ..., g_I \sim_{i.i.d.}$ *Gumbel*$(0, 1)$, then

$$\boldsymbol{y} \sim \text{Gumbel-Softmax Distribution}, \quad y_j := \frac{1}{\tau} softmax(\log \pi_0 + g_0, \log \pi_1 + g_1, ..., \log \pi_I + g_I), j = 1, 2, ..., I,$$

where τ is the temperature parameter. As $\tau \to 0^+$, the distribution of $\boldsymbol{y}$ approaches the distribution of z_j. Thus, $q_\psi(z_j|\boldsymbol{h}^a, \boldsymbol{h}^m_{j-1})$ can be reparameterized as

$$q_\psi(z_j|\boldsymbol{h}^o, \boldsymbol{h}^I_{j-1}) = q(f_j(\boldsymbol{g}; \psi, \tau)|\boldsymbol{h}^o, \boldsymbol{h}^I_{j-1}).$$

Since f is a differentiable mapping w.r.t. to ψ, we can take the gradient

$$\begin{aligned} &\nabla_\psi \left(\mathbb{E}_{q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})} \left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_{mis}, \boldsymbol{x}_{obs})\right] + \mathbb{H}\left[q_\psi(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs})\right]\right) \\ &= \nabla_\psi \left(\mathbb{E}_{\boldsymbol{g}_1, \boldsymbol{g}_2, ..., \boldsymbol{g}_M \sim s(\boldsymbol{g})} \left[\log p_{\boldsymbol{\theta}} \left(\{f_j(\boldsymbol{g}_j; \psi, \tau)\}_{j=1}^M, \boldsymbol{x}_{obs}\right)\right] + \nabla_\psi \mathbb{H}\left[q\left(\{f_j(\boldsymbol{g}_j; \psi, \tau)\}_{j=1}^M\right)\right]\right) \\ &= \mathbb{E}_{\boldsymbol{g}_1, \boldsymbol{g}_2, ..., \boldsymbol{g}_M \sim s(\boldsymbol{g})} \left[\sum_{j=1}^M \frac{\partial \log p_{\boldsymbol{\theta}}(\{f_j(\boldsymbol{g}_j; \psi, \tau)\}_{j=1}^M, \boldsymbol{x}_{obs})}{\partial f_j(\boldsymbol{g}_j; \psi, \tau)} \nabla_\psi f_j(\boldsymbol{g}_j; \psi, \tau)\right] + \nabla_\psi \mathbb{H}\left[q\left(\{f_j(\boldsymbol{g}_j; \psi, \tau)\}_{j=1}^M\right)\right]. \end{aligned}$$

Note that the term in red has a closed-form expression. Therefore, we can sample from the distribution $s(\boldsymbol{g})$ to get a Monte-Carlo estimate of the gradient w.r.t. ψ of the objective function.

3 Experiments

References

[BGJM11] Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of markov chain monte carlo*. CRC press, 2011.

[BKM17] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

[CMP08] Bo Cai, Renate Meyer, and François Perron. Metropolis–hastings algorithms with adaptive proposals. *Statistics and Computing*, 18(4):421–433, 2008.

[DFG01] Arnaud Doucet, Nando de Freitas, and Neil Gordon. An introduction to sequential monte carlo methods. In *Sequential Monte Carlo methods in practice*, pages 3–14. Springer, 2001.

[DVJ$^+$03] Daryl J Daley, David Vere-Jones, et al. *An introduction to the theory of point processes: volume I: elementary theory and methods*. Springer, 2003.

[Haw71] Alan G Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971.

[HBWP13] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 2013.

[HST01] Heikki Haario, Eero Saksman, and Johanna Tamminen. An adaptive metropolis algorithm. *Bernoulli*, pages 223–242, 2001.

[JGP16] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[KW13] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[LWB17] Scott W Linderman, Yixin Wang, and David M Blei. Bayesian inference for latent hawkes processes. *Advances in Neural Information Processing Systems*, 2017.

[LWZ$^+$20] Shuang Li, Lu Wang, Ruizhi Zhang, Xiaofu Chang, Xuqin Liu, Yao Xie, Yuan Qi, and Le Song. Temporal logic point processes. In *International Conference on Machine Learning*, pages 5990–6000. PMLR, 2020.

[ME17] Hongyuan Mei and Jason M Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. *Advances in neural information processing systems*, 30, 2017.

[MQE19] Hongyuan Mei, Guanghui Qin, and Jason Eisner. Imputing missing events in continuous-time event streams. In *International Conference on Machine Learning*, pages 4475–4485. PMLR, 2019.

[Mur23] Kevin P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023.

[MZRS11] Benjamin M Marlin, Richard S Zemel, Sam T Roweis, and Malcolm Slaney. Recommender systems: missing data and statistical model estimation. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[NLRB18] Christian Naesseth, Scott Linderman, Rajesh Ranganath, and David Blei. Variational sequential monte carlo. In *International conference on artificial intelligence and statistics*, pages 968–977. PMLR, 2018.

[ORBD05] Sang Min Oh, James M Rehg, Tucker Balch, and Frank Dellaert. Data-driven mcmc for learning and inference in switching linear dynamic systems. Georgia Institute of Technology, 2005.

[PA12] Kay I Penny and Ian Atkinson. Approaches for dealing with missing data in health care studies. *Journal of clinical nursing*, 21(19pt20):2722–2729, 2012.

[Ras13] Jakob Gulddahl Rasmussen. Bayesian inference for hawkes processes. *Methodology and Computing in Applied Probability*, 15(3):623–642, 2013.

[Ras18] Jakob Gulddahl Rasmussen. Lecture notes: Temporal point processes and the conditional intensity function. *arXiv preprint arXiv:1806.00221*, 2018.

[RGB14] Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial intelligence and statistics*, pages 814–822. PMLR, 2014.

[SRdS$^+$19] Koustuv Saha, Manikanta D Reddy, Vedant das Swain, Julie M Gregg, Ted Grover, Suwen Lin, Gonzalo J Martinez, Stephen M Mattingly, Shayan Mirjafari, Raghu Mulukutla, et al. Imputing missing social media data stream in multisensor studies of human behavior. In *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 178–184. IEEE, 2019.

[ZJL$^+$20] Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. Transformer hawkes process. In *International conference on machine learning*, pages 11692–11702. PMLR, 2020.