Machine Learning for Graphs and Sequential Data Exercise Sheet 03

Temporal Point Processes

Problem 1: Consider a temporal point process, where all the inter-event times $\tau_i = t_i - t_{i-1}$ are sampled i.i.d. from the distribution with the survival function

$$S(\tau) = \exp\Big(-(e^{b\tau} - 1)\Big)$$

with a parameter $b > 0$.

a) Write down the closed-form expression for the conditional intensity function $\lambda^*(t)$ of this TPP. Simplify as far as you can.

First, let's find the intensity function for the inter-event time distribution. We denote this intensity as $h(\tau)$ and use the definition

$$h(\tau) = \frac{p(\tau)}{S(\tau)}$$

That means, we first need to find $p(\tau)$. We recall from the lecture the following two facts

$$F(\tau) = 1 - S(\tau) \qquad\qquad p(\tau) = \frac{d}{d\tau} F(\tau)$$

By combining them, we can conclude that

$$\begin{aligned} p(\tau) &= -\frac{d}{d\tau} S(\tau) \\ &= -\frac{d}{d\tau} \exp\Big(-(e^{b\tau} - 1)\Big) \\ &= b \exp(b\tau) \exp\Big(-(e^{b\tau} - 1)\Big) \end{aligned}$$

Now, we use the definition of intensity

$$\begin{aligned} h(\tau) &= \frac{p(\tau)}{S(\tau)} \\ &= \frac{b \exp(b\tau) \exp\big(-(e^{b\tau} - 1)\big)}{\exp\left(-(e^{b\tau} - 1)\right)} \\ &= b \exp(b\tau) \end{aligned}$$

We can now define the overall conditional intensity $\lambda^*(t)$ of the entire TPP as

$$\lambda^*(t) = b \exp(b(t - t_{i-1}))$$

where t_{i-1} is the last event that happened before t.

b) Write down the closed-form expression for the log-likelihood of a sequence $\{t_1, ..., t_N\}$ generated from this TPP on the interval $[0, T]$. Simplify as far as you can.

Let's start by writing the general expression for the likelihood of a TPP

$$\begin{aligned} p(\{t_1, ..., t_N\}) &= \left(\prod_{i=1}^{N} p^*(t_i) \right) S^*(T) \\ &= \left(\prod_{i=1}^{N} \lambda^*(t_i) S^*(t_i) \right) S^*(T) \end{aligned}$$

Now apply the logarithm

$$\log p(\{t_1, ..., t_N\}) = \left(\sum_{i=1}^{N} \log \lambda^*(t_i) + \log S^*(t_i) \right) + \log S^*(T)$$

We can compute the log-intensity and log-survival as

$$\log \lambda^*(t_i) = \log b + b(t_i - t_{i-1}) \qquad \log S^*(t_i) = 1 - e^{b(t_i - t_{i-1})}$$

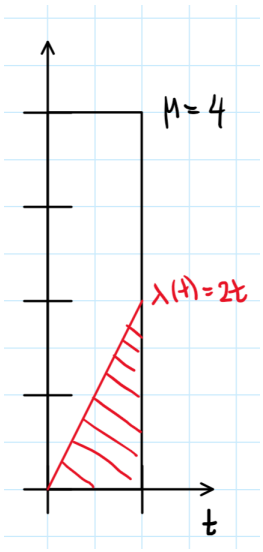
Putting everything together we obtain

$$\begin{aligned} p(\{t_1, ..., t_N\}) &= \left(\sum_{i=1}^{N} \log b + b(t_i - t_{i-1}) + 1 - e^{b(t_i - t_{i-1})} \right) + 1 - e^{b(T - t_N)} \\ &= N \log b + N + 1 + b t_N - \left(\sum_{i=1}^{N+1} e^{b(t_i - t_{i-1})} \right) \end{aligned}$$

where we denoted $t_0 = 0$ and $t_{N+1} = T$.

Problem 2: Consider an inhomogeneous Poisson process (IPP) on $[0, 1]$ with the intensity function $\lambda(t) = 2t$. We simulate a sample from this IPP using thinning. For this, we first simulate a *homogeneous* Poisson process (HPP) with intensity $\mu = 4$ and apply the thinning procedure described in the lecture. What is the expected number of events from the HPP that will be rejected when using this procedure?

Let's plot the the intensity function $\lambda(t)$ of the IPP and the intensity of the HPP μ



Recall from the lecture, that the expected number of events generated from the IPP is equal to the area under $\lambda(t)$ (red shaded area in the figure). We can easily compute that it's equal to $\int_0^1 2t dt = 1$.

Using the same reasoning, we can conclude that the expected number of events generated from the HPP is equal to 4, since HPP can be seen as an IPP with constant intensity.

By combining these two facts we conclude that on average 3 events will be discarded (which happens to correspond to the area of the unshaded region).

Problem 3: Consider an inhomogeneous Poisson process on $[0, 4]$ with the intensity function $\lambda(t) = \beta t$, where $\beta > 0$ is a parameter that has to be estimated. You have observed a single sequence $\{1, 2.1, 3.3, 3.8\}$ generated from this IPP. What is the maximum likelihood estimate of the parameter β?

We need to solve the following optimization problem

$$\max_{\beta} \log p(\{t_1, ..., t_N\}|\beta)$$

where the log-likelihood is computed as

$$\begin{aligned}
\log p(\{t_1, ..., t_N\}|\beta) &= \log \left[\left(\prod_{i=1}^{N} \lambda(t_i) \right) \exp \left(- \int_0^T \lambda(u) du \right) \right] \\
&= \sum_{i=1}^{N} \log \lambda(t_i) - \int_0^T \lambda(u) du \\
&= \sum_{i=1}^{N} \log(\beta t_i) - \int_0^T \beta u du \\
&= N \log \beta - \frac{T^2}{2} \beta + \underbrace{\sum_{i=1}^{N} \log(t_i)}_{\text{constant in } \beta}
\end{aligned}$$

Since this is a sum of concave functions, we can compute the derivative, set it to zero and solve for β in order to find the maximum

$$\begin{aligned}
\frac{d}{d\beta} \log p(\{t_1, ..., t_N\}|\beta) &= \frac{N}{\beta} - \frac{T^2}{2} \stackrel{!}{=} 0 \\
\frac{T^2}{2} \beta &\stackrel{!}{=} N \\
\beta &= \frac{2N}{T^2} \\
\beta &= \frac{1}{2}
\end{aligned}$$

Problem 4: Consider a *neural* temporal point process where the conditional intensity function is defined with a neural network. In particular, for a time point t_i, we represent the history $\{t_1, t_2, \ldots, t_{i-1}\}$ with a fixed-sized vector $\boldsymbol{h}_i \in \mathbb{R}^d$. The conditional intensity function $\lambda^*(t)$ is defined as a function of $\boldsymbol{h}_i$. We will use the transformer architecture (see previous lecture). We propose the following implementation.

Given the full sequence $\{t_1, t_2, \ldots, t_n\}$, we calculate all $\{\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_n\}$ in parallel. We first calculate vectors $\boldsymbol{q}_i, \boldsymbol{k}_i, \boldsymbol{v}_i \in \mathbb{R}^d$ as a function of t_i. We stack these vectors into matrices $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} \in \mathbb{R}^{n \times d}$. The output of the transformer is: $\boldsymbol{H} = \text{softmax}(\boldsymbol{Q}\boldsymbol{K}^T)\boldsymbol{V}$, then $\boldsymbol{h}_i$ is the ith row of $\boldsymbol{H}$.

Identify the errors in this implementation compared to the original definition of $\boldsymbol{h}_i$. Propose a solution.

The vector $\boldsymbol{h}_i$ was originally defined as a representation of all the time points up until t_i, excluding t_i. In the proposed implementation, $\boldsymbol{h}_i$ is a function of all the points, including those from *the future*. This breaks the assumed autoregressive property of the model which means the usual likelihood definition does not hold.

To fix this, consider the sequence consisting of two elements $\{t_1, t_2\}$. What we would like to output is two vectors, $\boldsymbol{h}_1$ that does not depend on either t_1 or t_2, and $\boldsymbol{h}_2$ that depends on t_1 only. We modify

the initial sequence by shifting the whole sequence by one element to the right and pad with the zero value from the left: $\{0, t_1\}$. Now we only need to ensure that the future elements do not influence the past. The interaction between different elements $i \neq j$ happens when we multiply the attention matrix softmax($\boldsymbol{QK}^T$) with the values $\boldsymbol{V}$. That is, the final vector $\boldsymbol{h}_i$ is obtained by multiplying the ith row of the attention matrix with elements of $\boldsymbol{V}$. We can see this as a weighted sum of vectors $\boldsymbol{v}_j \in \boldsymbol{V}$. To prevent the future influencing the past, the weight values for $j > i$ should be zero. Thus, the ith row of the attention matrix should have zeros on all the places $j > i$. This corresponds to having a lower-triangular matrix. What we do in the end is mask the $\boldsymbol{QK}^T$ matrix such that the values above the diagonal are set to $-\infty$. The softmax operation will give zeros on these places and each row will still sum up to 1.

To summarize, we shift the whole sequence by one place to the right and impose the lower triangular structure on the attention matrix using infinity masking. Another approach is to impose the lower triangular structure where the diagonal is also zeroed-out.