Machine Learning for Graphs and Sequential Data

Sequential Data – Temporal Point Processes

lecturer: Prof. Dr. Stephan Günnemann
www.daml.in.tum.de

Summer Term 2024

Data Analytics and
Machine Learning

Roadmap

- Chapter: Temporal Data / Sequential Data
  1. Autoregressive Models
  2. Markov Chains
  3. Hidden Markov Models
  4. Neural Network Approaches
  5. **Temporal Point Processes**
     a) **Introduction**
     b) Selected TPP Models

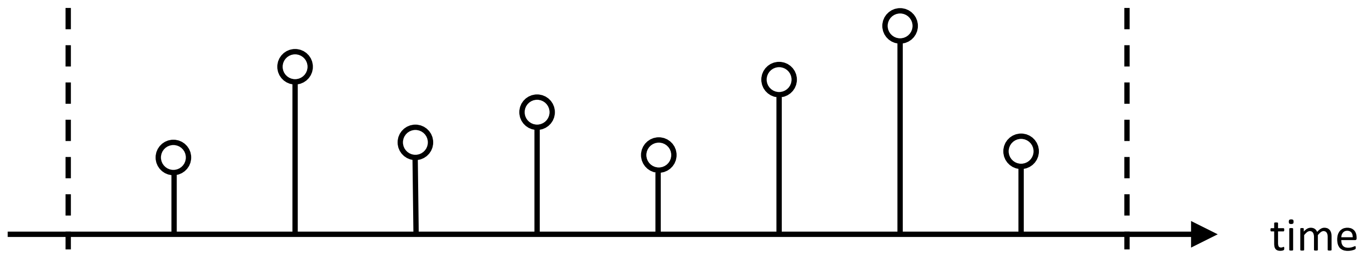
Event Data

- Our data consists of **discrete** events in **continuous** time, such as
  - Transaction times in finance
  - Messages on social media
  - Visits to hospitals in electronic health records

- Prediction tasks
  - When will the next event happen?
  - How many events will happen in the next hour/day/week?

Difference to Time Series

- Time series
  - Measurements (signal) collected at regular intervals



- (Asynchronous) event data
  - Irregular intervals
  - We care about the time of the occurrence

Temporal Point Processes

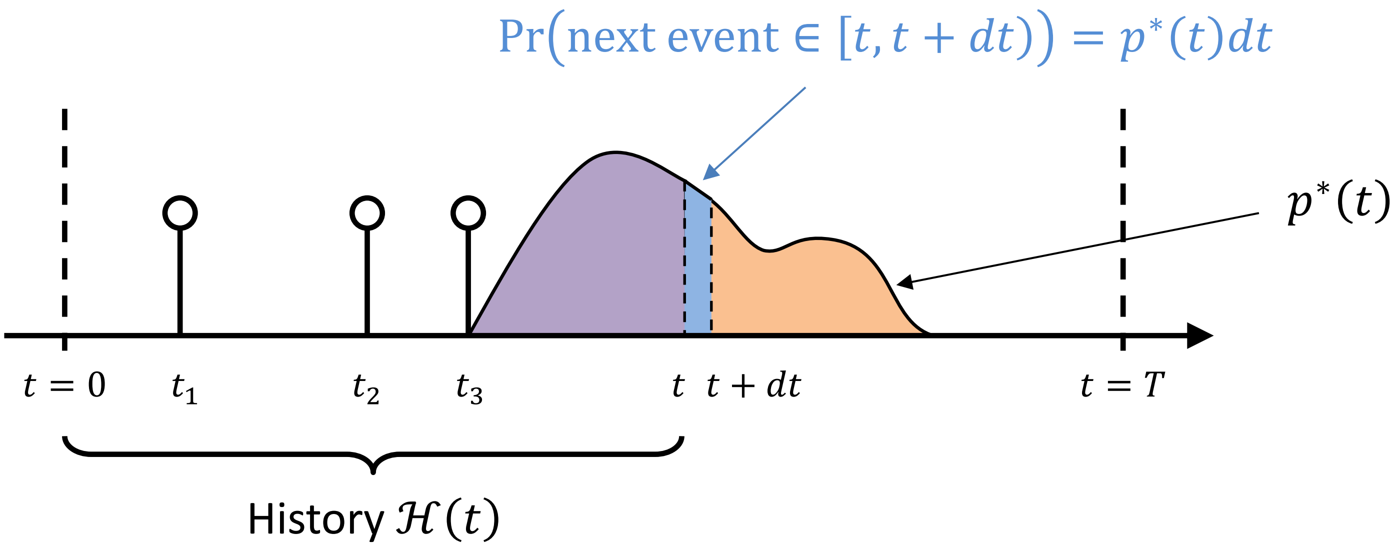
- Temporal Point Processes (TPP) are a class of probabilistic models that describe the distribution of **discrete** event sequences in **continuous** time

- TPP defines a generative model for variable-length sequences $\boldsymbol{t} = \{t_1, \ldots, t_N\}$ on the interval $[0, T]$
  - Both locations of the events t_i and their number N are random

- TPPs also provide a likelihood function $p(\{t_1, \ldots, t_N\})$

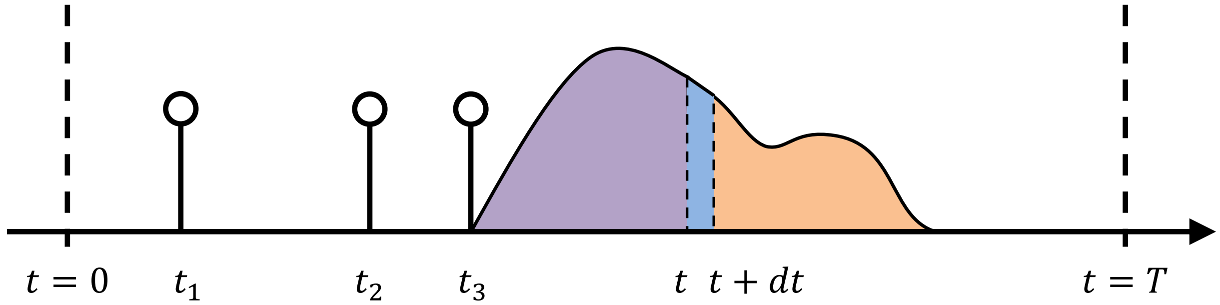
Modeling the Time of the Next Event

- We can model the distribution $p(\{t_1, \dots, t_N\})$ autoregressively
  - Predict the time of the <u>next</u> event t_i given the history $\mathcal{H}(t) = \{t_j < t\}$
  - Important: $\mathcal{H}(t)$ depends on the specific sample $\{t_1, \dots, t_N\}$!
  - We denote the conditional density as $p^*(t) := p(t|\mathcal{H}(t))$



$\text{next event} \in [t, t+dt) \quad \Leftrightarrow \quad \text{event in } [t, t+dt) \;\&\; \text{no event in } [t_3, t)$

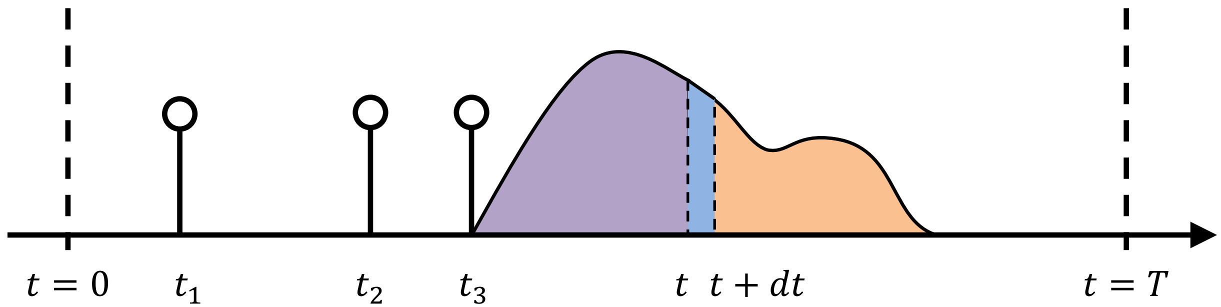
Alternative Ways to Model the Inter-event Time



- Cumulative distribution function (CDF)
  - $F^*(t) = \int_{t_{i-1}}^{t} p^*(u)du =$ Probability that the next event happens in $[t_{i-1}, t)$
  - t_{i-1} is the last event that happened before t
- Survival function
  - $S^*(t) = 1 - F^*(t) = \int_{t}^{\infty} p^*(u)du$
  - Probability that the next event doesn't happen before t
  - Probability that the next event happens after t

Conditional Intensity Function

- There exists another way to describe the conditional distribution



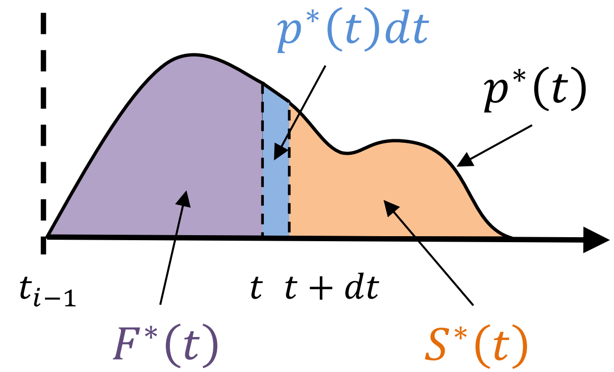
- Conditional intensity
  - $\lambda^*(t)dt =$ probability of event in $[t, t + dt)$ **given** no event in $[t_{i-1}, t)$

$$\lambda^*(t)dt = \frac{\Pr\big(\text{event in } [t, t + dt) \text{ \& no event in } [t_{i-1}, t)\big)}{\Pr(\text{no event in } [t_{i-1}, t))} = \frac{p^*(t)dt}{S^*(t)}$$

- Intuitive meaning of $\lambda^*(t)$: Expected # of events / unit of time
  - We will demonstrate this later

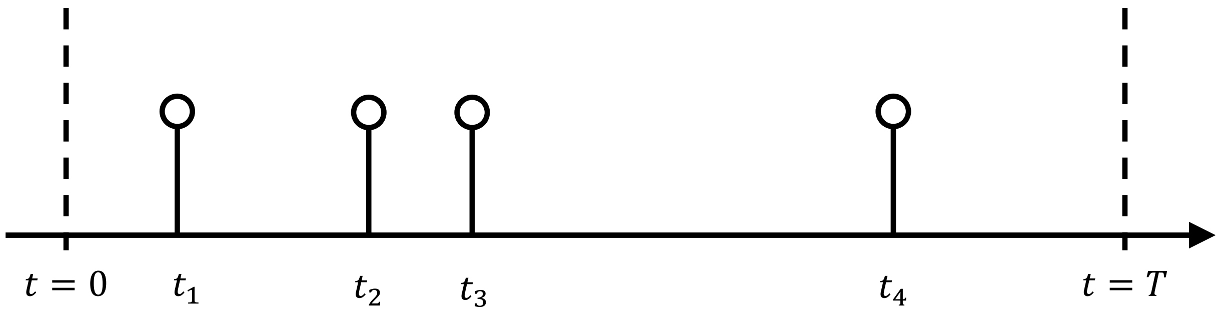
Relation between p^*, F^*, S^*, λ^*

$$
\begin{tikzcd}[column sep=huge, row sep=huge]
F^*(t) \arrow[r, shift left, "\frac{dF^*(t)}{dt}"] \arrow[d, shift left, "1 - F^*(t)"] & p^*(t) \arrow[l, shift left, "\int_{t_{i-1}}^{t} p^*(u)du"] \arrow[d, bend right] \\
S^*(t) \arrow[u, shift left, "1 - S^*(t)"] \arrow[r, bend left, "\frac{p^*(t)}{S^*(t)}"] & \lambda^*(t) \arrow[u, "\lambda^*(t)\exp\left(-\int_{t_{i-1}}^{t} \lambda^*(u)du\right)"'] \arrow[l, shift left, "\exp\left(-\int_{t_{i-1}}^{t} \lambda^*(u)du\right)"]
\end{tikzcd}
$$



Likelihood of an Entire Sequence

- How can we compute the likelihood of a realization $\{t_1, \dots, t_N\}$?



$$p(\{t_1, t_2, t_3, t_4\}) = p^*(t_1)\, p^*(t_2)\, p^*(t_3)\, p^*(t_4)\, S^*(T)$$
$$= \lambda^*(t_1)\, \lambda^*(t_2)\, \lambda^*(t_3)\, \lambda^*(t_4) \exp\left(-\int_0^T \lambda^*(u) du\right)$$

- Remember that the number of events can vary

Roadmap

- Chapter: Temporal Data / Sequential Data
  1. Autoregressive Models
  2. Markov Chains
  3. Hidden Markov Models
  4. Neural Network Approaches
  5. **Temporal Point Processes**
     a) Introduction
     b) Selected TPP Models

Models based on Conditional Intensity

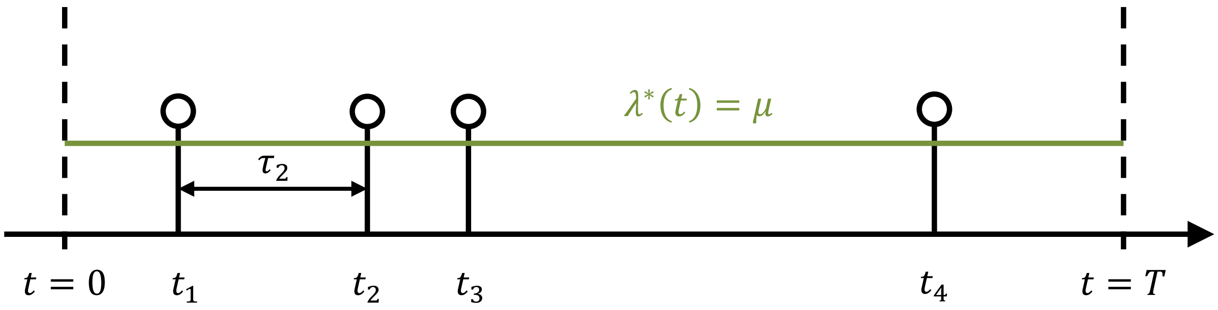
- Defining TPPs in terms of $\lambda^*(t)$ has several advantages

1. Easy to define TPPs with pre-defined behavior
   - Global trend, burstiness, repulsiveness
   - Intensity is more interpretable

2. Easy to combine different TPPs with different $\lambda^*(t)$'s

3. Efficient sampling

Homogeneous Poisson Process (HPP)



- Simplest possible model: constant intensity

$$\lambda^*(t) = \mu$$

- Inter-event times follow exponential distribution

$$p^*(t) = \mu \exp\left(-\int_{t_{i-1}}^{t} \mu \, du\right) = \mu \exp\big(-\mu \underbrace{(t - t_{i-1})}_{\text{inter-event time } \tau_i}\big)$$

Simulating an HPP

- We can simulate an HPP by generating the inter-event times

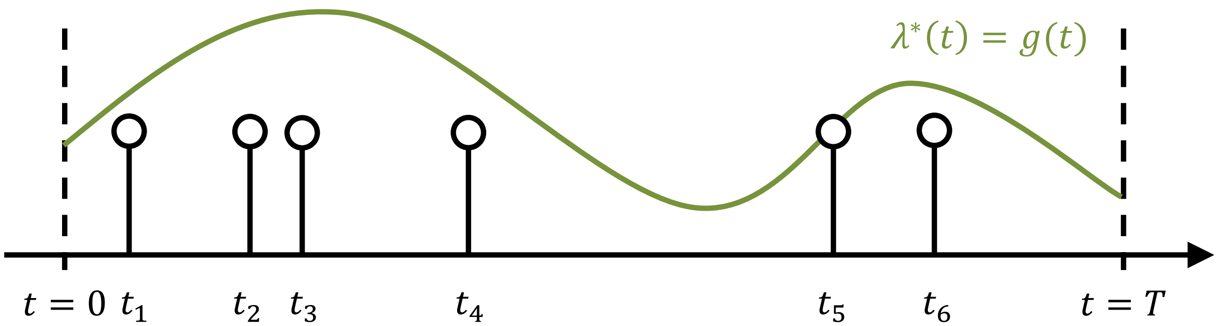
```
arrival_times = []
t = 0
while t < T:
    tau ~ Exponential(mu)
    t += tau
    if t < T:
        arrival_times.append(t)
```

- How to sample from the exponential distribution? – Inverse CDF transform

$$u = F(\tau) = 1 - \exp(-\mu\tau) \;\Rightarrow\; \tau = F^{-1}(u) = -\frac{1}{\mu}\log(1-u)$$

where $u \sim \mathrm{Uniform}(0,1)$ and F is the CDF of the exponential distribution

Inhomogeneous Poisson Process (IPP)



- The intensity changes over time

$$\lambda^*(t) = g(t) \geq 0$$

- Intensity is independent of the history

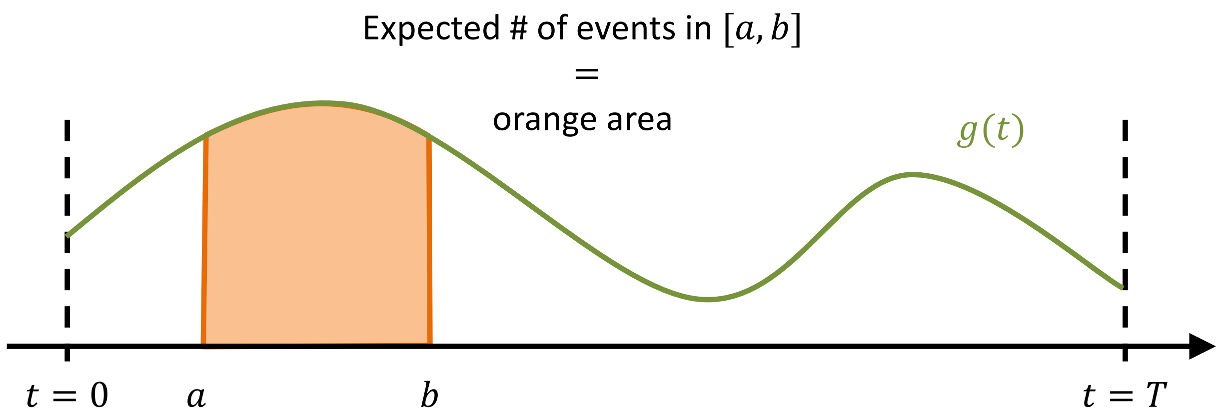
- Captures global trend
  - More events happen in the regions with higher intensity

Expected Number of Events

- Number of events in an interval $[a, b] \subseteq [0, T]$ follows Poisson distribution

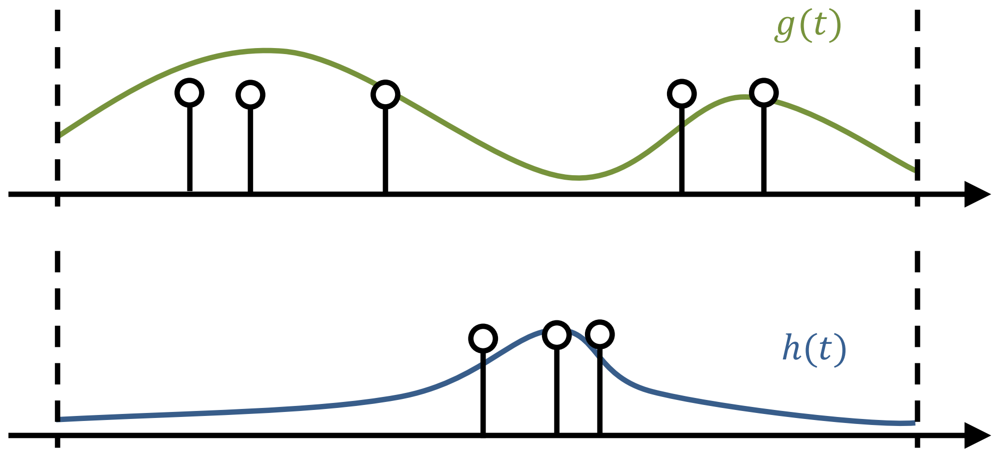
$$N([a, b]) \sim \text{Poisson}\left(\int_a^b g(t)dt\right)$$

- This means, the expected number of events inside $[a, b]$ is equal to the total intensity over this region

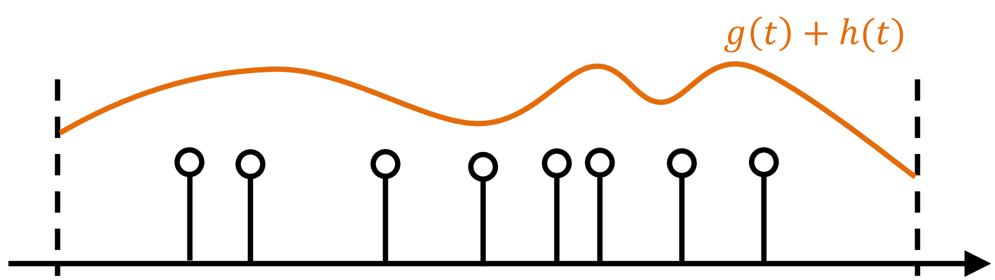


Superposition of IPPs

- Consider two IPPs with intensities $g(t)$ and $h(t)$



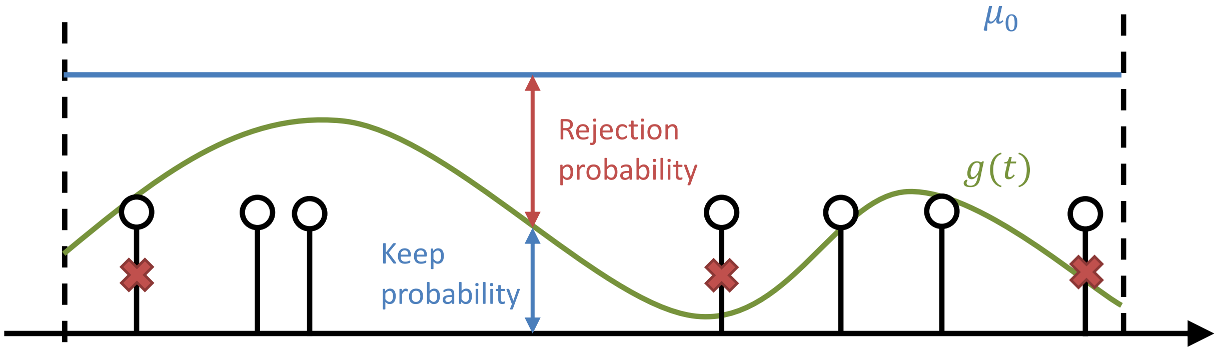
- Combination of the two IPPs is again an IPP with intensity $g(t) + h(t)$



- This also applies to a general $\lambda^*(t)$, but showing this is more involved

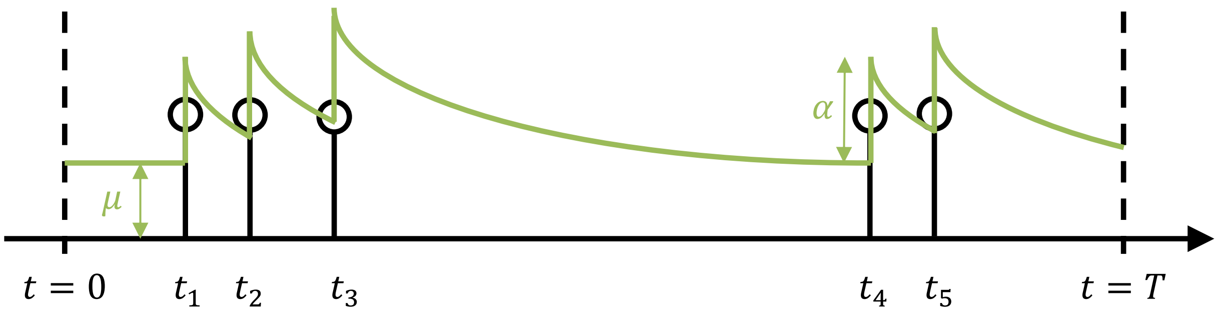
Simulating an IPP

- Simulating the inter-event times is hard (requires integration)
- Better alternative – thinning



1. Find an upper bound $\mu_0 \geq g(t)$ for all t
2. Simulate candidate events $\{t_1, t_2, \dots\}$ from a HPP with rate μ_0
3. Keep each t_i with probability $g(t_i)/\mu_0$

Hawkes Process



- Also known as self-exciting process

$$\lambda^*(t) = \mu + \alpha \sum_{t_j \in \mathcal{H}(t)} k_\omega(t - t_j)$$

  - Triggering kernel $k_\omega(t - t_i) = \exp\left(-\omega(t - t_i)\right)$
  - Parameters $\mu, \alpha, \omega \geq 0$
- Intensity depends on the history
- Clustered ("bursty") event occurrences

Parameter Estimation in TPPs

- Pick a parametric conditional intensity $\lambda_{\boldsymbol{\theta}}^*(t)$ (e.g. Hawkes, IPP)
- Maximize the log-likelihood of the observed sequences $\mathcal{D}_{\text{train}}$

$$\max_{\boldsymbol{\theta}} \sum_{\boldsymbol{t}=\{t_1,\dots,t_N\}\in\mathcal{D}_{\text{train}}} \log p_{\boldsymbol{\theta}}(\{t_1,\dots,t_N\})$$

  - The log-likelihood of a single sequence is

$$\log p_{\boldsymbol{\theta}}(\{t_1,\dots,t_N\}) = \sum_{i=1}^{N} \log \lambda^*(t_i) - \int_0^T \lambda^*(u)du$$

  - Remember, different sequences have different length N

- Lots of different optimization techniques possible
  - Simple models like HPP allow closed-form solutions
  - For Hawkes process we can use convex optimization methods
  - Always possible to use gradient descent (with modifications for constraints)

Conditional Intensity: Summary

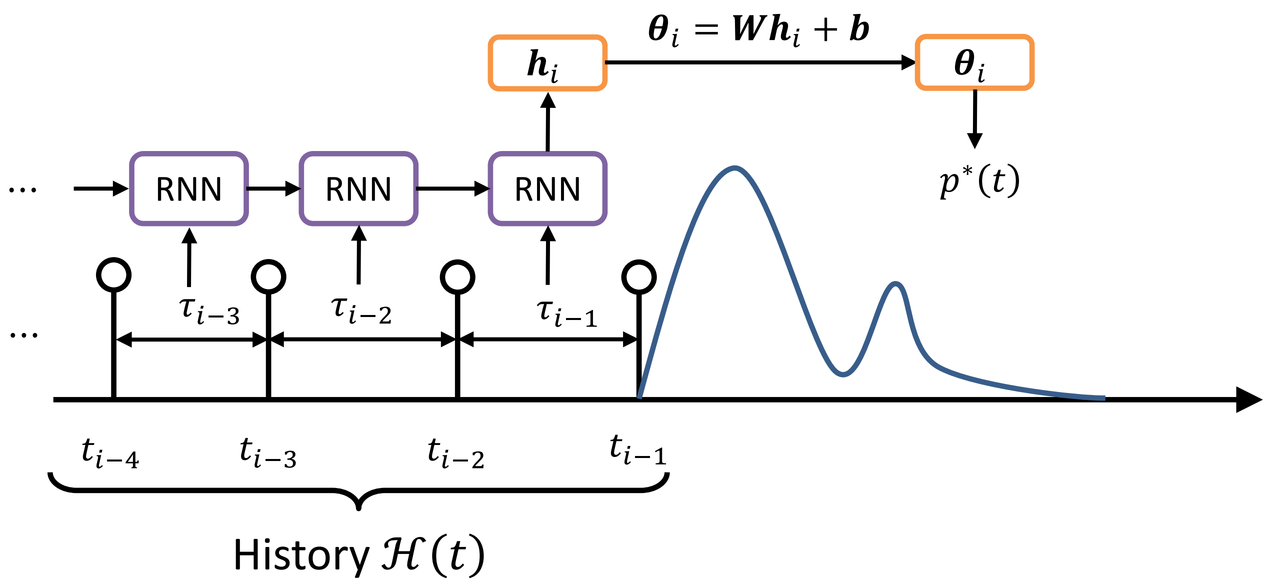
- Conditional intensity $\lambda^*(t)$ provides an alternative to the conditional density $p^*(t)$ when constructing TPPs

- Advantages
  - Easy to define models with simple behavior
  - Interpretable
  - Efficient sampling

- Limitations
  - Integration required to compute the log-likelihood – might be intractable
  - Not clear how to define flexible models with arbitrary dynamics

- We will define more flexible TPPs by going back to $p^*(t)$ and using RNNs

Modeling TPPs with RNNs



- Directly model the conditional distribution $p^*(t)$ using an RNN

1. Every time an event happens, we feed τ_i into the RNN
2. Use the hidden state $\boldsymbol{h}_i \in \mathbb{R}^D$ of the RNN as the history embedding
3. Use $\boldsymbol{h}_i$ to generate the parameters $\boldsymbol{\theta}_i$ of the distribution $p^*(t)$

$$p^*(t) = p(t|\mathcal{H}(t)) = p(t|\boldsymbol{h}_i)$$

How to Model $p^*(t)$?

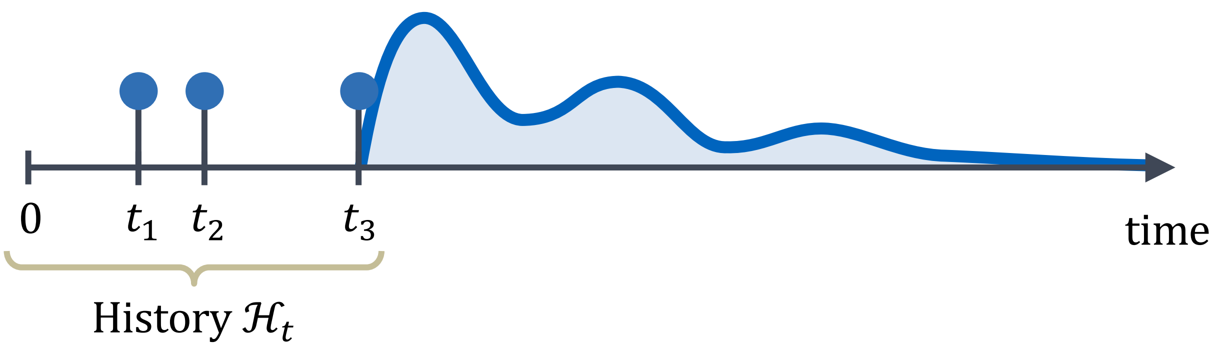
- The sequence of events must be increasing: $t_i > t_{i-1}$ for all i
- We can instead model the distribution of the inter-event times τ_i
  - It's sufficient to ensure that $\tau_i > 0$

- How to define a flexible and tractable $p^*(\tau_i)$?

- Simple nonnegative distribution
  - Exponential, Gamma, Weibull, Gompertz, …
- Mixture distribution
  - Take a convex combination of simple densities
- Normalizing flows
  - Use transformations like $\exp(x)$ or $\log(1 + \exp(x))$ to ensure non-negativity
  - Combine with other transformations (e.g., polynomials, NNs with positive weights) to add flexibility

Problems with Autoregressive TPP's

$$\lambda(t \mid \mathcal{H}_t) = \lim_{dt \to 0} \frac{\Pr(\text{next event} \in [t, t + dt) \mid \mathcal{H}_t)}{dt}$$

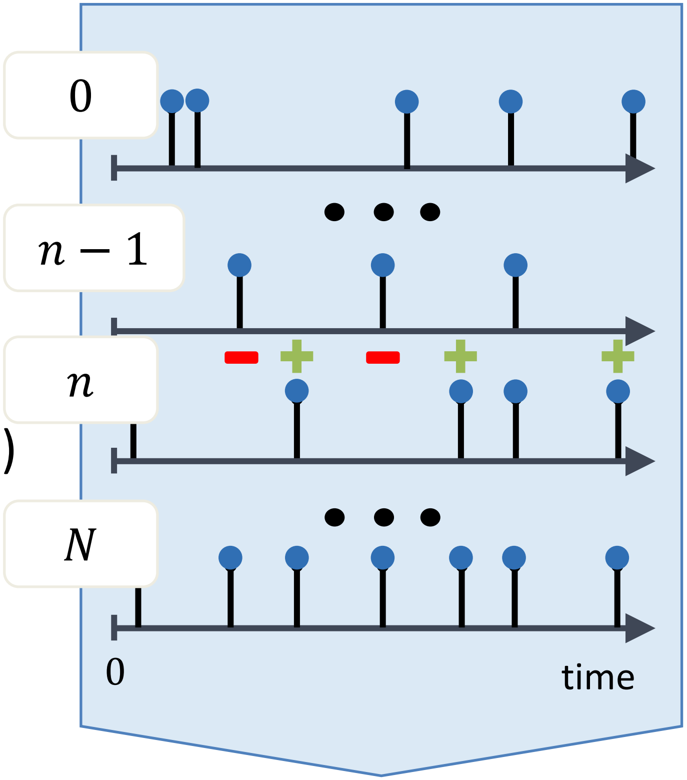


- ✗ Error accumulation
- ✗ Computational complexity
- ✗ Hard to model long-range interactions

Alternative: ADD and THIN – Diffusion for TPPs

Noising

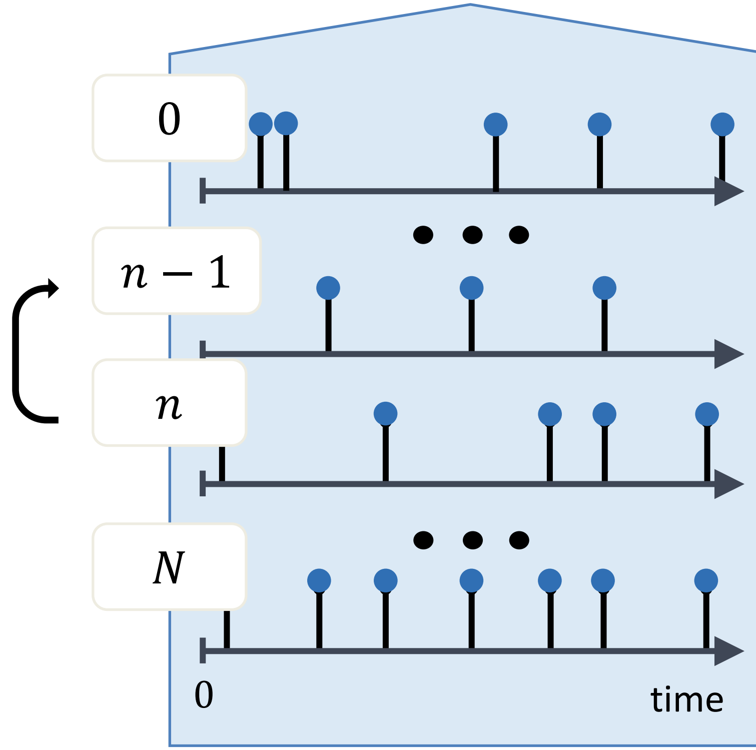
- Start: Data sample $\boldsymbol{t}^{(0)} \sim \lambda_0$
- For every diffusion step n:
  1. **Thin** $\boldsymbol{t}^{(n-1)}$
  2. **Add** events from HPP (superposition)
- End: Noised data $\boldsymbol{t}^{(N)} \sim \lambda_{HPP}$



Alternative: ADD and THIN – Diffusion for TPPs

Denoising

- Start: Noised data $\boldsymbol{t}^{(N)} \sim \lambda_{HPP}$
- For every diffusion step n:
  Sample $\boldsymbol{t}^{(n-1)} \sim \lambda_{n-1}(t \mid \boldsymbol{t}^{(0)}, \boldsymbol{t}^{(n)})$
- End: Data sample $\boldsymbol{t}^{(0)} \sim \lambda_0$



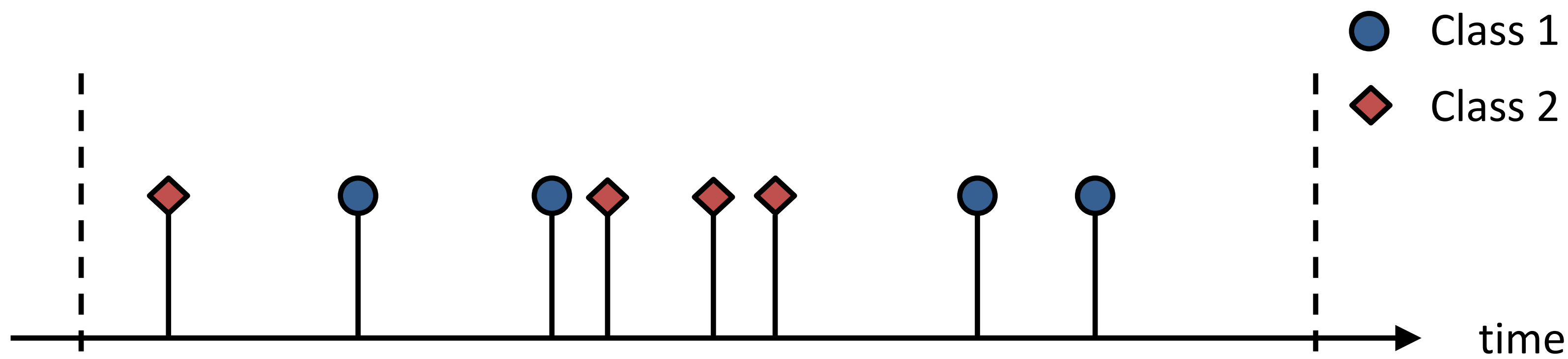
For further details see our paper:
https://arxiv.org/pdf/2311.01139

What we haven't covered

- Modeling TPPs with marks
  - https://www.research-collection.ethz.ch/handle/20.500.11850/151886
  - https://www.kdd.org/kdd2016/papers/files/rpp1081-duA.pdf
- More efficient sampling techniques
  - https://web.ics.purdue.edu/~pasupath/PAPERS/2011pasB.pdf
- Spatial and spatio-temporal point processes – modeling events in space
  - https://arxiv.org/abs/1708.02647

Marked Temporal Point Processes

- Most common type: categorical marks
  - Each event has an associated class (i.e., category, type)
  - Events of different classes may influence each other
  - E.g., activity of each use is represented by a different mark



- Continuous marks also possible
  - E.g., magnitude of the earthquake, amount of money spent by a customer

Questions – TPP

1. Is it possible to obtain the conditional intensity $\lambda^*(t)$ if you know only the survival function $S^*(t)$ and don't know the conditional PDF $p^*(t)$?
2. Would you use (a) Hawkes process or (b) inhomogeneous Poisson process to model the following event data?
   – Customers visiting a supermarket (event = customer enters the supermarket)
   – Messages sent by a single user on WhatsApp (event = message sent)
   – Taxi rides in a city (event = a trip starts)
3. What can you say about a TPP with the following conditional intensity function? What kind of behavior does it model?

$$\lambda^*(t) = \exp\left(t - \sum_{t_i \in \mathcal{H}(t)} 1\right)$$

Acknowledgments

- These slides are based on the ICML 2018 tutorial by Manuel Gomez Rodriguez & Isabel Valera (http://learning.mpi-sws.org/tpp-icml18/)

Recommended Reading

- Lecture notes on TPPs by De, Upadhyay and Gomez-Rodriguez
  - http://courses.mpi-sws.org/hcml-ws18/lectures/TPP.pdf
  - Except Section 3.4, 4
- Alternatively, lecture notes by Rasmussen
  - https://arxiv.org/abs/1806.00221
  - Except Sections 2.4, 3.2, 4.2, 5, 6
- Modeling TPPs with recurrent neural networks
  - https://arxiv.org/abs/1909.12127
  - https://www.kdd.org/kdd2016/papers/files/rpp1081-duA.pdf
- ADD and THIN: Diffusion for Temporal Point Processes
  - https://arxiv.org/pdf/2311.01139

Machine Learning for Graphs and Sequential Data

Sequential Data – Temporal Point Processes

lecturer: Prof. Dr. Stephan Günnemann
www.daml.in.tum.de

Summer Term 2024

Data Analytics and
Machine Learning