Learning with Temporal Point Processes

Manuel Gomez Rodriguez
MPI for Software Systems

Isabel Valera
MPI for Intelligent Systems

Slides/references: http://learning.mpi-sws.org/tpp-icml18

ICML TUTORIAL, JULY 2018
Many discrete events in continuous time

Online actions

Financial trading

Disease dynamics

Mobility dynamics
Variety of processes behind these events

Events are (noisy) observations of a variety of complex dynamic processes...

- Stock trading
- Flu spreading
- Article creation in Wikipedia
- News spread in Twitter
- Reviews and sales in Amazon
- Ride-sharing requests
- A user’s reputation in Quora

...in a wide range of temporal scales.
Example I: Information propagation

S → D means D follows S

They can have an impact in the off-line world

Friggeri et al., 2014
Example II: Knowledge creation

Barack Obama: Revision history

03:41, 28 November 2016  Hanoi (talk | contribs)  . . (301,156 bytes) [+18] . .
03:32, 28 November 2016  Xin Deui (talk | contribs)  . . (301,087 bytes) [-68] . .
00:57, 28 November 2016  SporkBox (talk | contribs)  m . . (301,156 bytes) [-37]
07:03, 27 November 2016  Saph12l1 (talk | contribs)  . . (301,192 bytes) [+25]

03:21, 20 September 2016

is a Kenyan politician

possible vandalism by MLM2016

is an American politician

What are the pros and cons of living in Australia?

Question

Answer

Upvote 150

M Sharma, Lived in Australia as Migrant, Student, Worker, Business Owner & Family Man

Upvote
Example III: Human learning

1st year computer science student

- Introduction to programming
- Discrete math
- Project presentation

- For/do-while loops
- Set theory
- Logic
- Private functions
- If ... else
- Graph Theory
- Inheritance
- Powerpoint vs. Keynote
- Export pptx to pdf
- Logic
- Plot library
- Set theory
- PP templates
- Geometry
- Class destructor
- Graph Theory
Aren’t these event traces just time series?

The framework of temporal point processes provides a *native representation*.

What about aggregating events in epochs?

- What if no event in one epoch?
- How long is each epoch?
- How to aggregate events per epoch?
- What about time-related queries?
# Outline of the Seminar

## Temporal Point Processes (TPPs):

### Intro
1. Intensity function
2. Basic building blocks
3. Superposition
4. Marks and SDEs with jumps

## Models & Inference
1. Modeling event sequences
2. Clustering event sequences
3. Capturing complex dynamics
4. Causal reasoning on event sequences

## RL & Control
1. Marked TPPs: a new setting
2. Stochastic optimal control
3. Reinforcement learning

Slides/references: learning.mpi-sws.org/tpp-icml18
Temporal Point Processes (TPPs): Introduction

1. Intensity function
2. Basic building blocks
3. Superposition
4. Marks and SDEs with jumps
Temporal point processes

Temporal point process:
A random process whose realization consists of discrete events localized in time $\mathcal{H} = \{t_i\}$

Formally:

$$N(t) = \int_0^t dN(s)$$

$$dN(t) = \sum_{t_i \in \mathcal{H}} \delta(t - t_i) \, dt$$
Model time as a random variable

Prob. between \([t, t+dt)\]

\[ f^*(t) \, dt \]

density

\[ f^*(t) := f(t|\mathcal{H}(t)) \]

Likelihood of a timeline:

\[ f^*(t_1) \, f^*(t_2) \, f^*(t_3) \, f^*(t) \, S^*(T) \]
Problems of density parametrization (I)

It is difficult for model design and interpretability:

1. Densities need to integrate to 1 (i.e., partition function)
2. Difficult to combine timelines
Intensity function

Intensity:
Probability between \([t, t+dt)\) but not before \(t\)

\[
\lambda^*(t)dt = \frac{f^*(t)dt}{S^*(t)} \geq 0 \quad \Rightarrow \quad \lambda^*(t)dt = \mathbb{E}[dN(t)|\mathcal{H}(t)]
\]

Observation: \(\lambda^*(t)\) It is a rate = # of events / unit of time
Advantages of intensity parametrization (I)

Suitable for model design and interpretable:

1. Intensities only need to be nonnegative
2. Easy to combine timelines
Relation between \( f^* \), \( F^* \), \( S^* \), \( \lambda^* \)

Central quantity we will use!
Representation: Temporal Point Processes

1. Intensity function
2. Basic building blocks
3. Superposition
4. Marks and SDEs with jumps
Poisson process

Intensity of a Poisson process

\[ \lambda^*(t) = \mu \]

Observations:

1. Intensity independent of history
2. Uniformly random occurrence
3. Time interval follows exponential distribution
Fitting & sampling from a Poisson

Fitting by maximum likelihood:

\[ \mu^* = \arg \max_{\mu} \sum \log \mu - \mu T = \frac{3}{T} \]

Sampling using inversion sampling:

\[ t \sim \mu \exp(-\mu(t - t_3)) \quad \Rightarrow \quad t = -\frac{1}{\mu} \log(1 - u) + t_3 \]
Inhomogeneous Poisson process

Intensity of an inhomogeneous Poisson process

\[ \lambda^*(t) = g(t) \geq 0 \quad \text{(Independent of history)} \]

Example:

\[ \lambda^*(t) = \sum_j \alpha_j k(t - t_j) \]
Fitting & sampling from inhomogeneous Poisson

\[ \lambda^*(t) = g(t) \geq 0 \]

Fitting by maximum likelihood:
\[
\text{maximize} \quad \sum_{i=1}^{n} \log g(t_i) - \int_0^T g(\tau) \, d\tau
\]

Sampling using thinning (reject. sampling) + inverse sampling:

1. Sample \( t \) from Poisson process with intensity \( \mu \) using inverse sampling
2. Generate \( u_2 \sim Uniform(0, 1) \)
3. Keep the sample if \( u_2 \leq g(t) / \mu \)

Keep sample with prob. \( g(t) / \mu \)
Terminating (or survival) process

Intensity of a terminating (or survival) process

\[ \lambda^*(t) = g^*(t)(1 - N(t)) \geq 0 \]

Observations:
1. Limited number of occurrences
Self-exciting (or Hawkes) process

Intensity of self-exciting (or Hawkes) process:

\[
\lambda^*(t) = \mu + \alpha \sum_{t_i \in \mathcal{H}(t)} \kappa_\omega(t - t_i)
\]

\[
= \mu + \alpha \kappa_\omega(t) \ast dN(t)
\]

Observations:
1. Clustered (or bursty) occurrence of events
2. Intensity is stochastic and history dependent
Fitting a Hawkes process from a recorded timeline

The max. likelihood is jointly convex in $\mu$ and $\alpha$.

Sampling using thinning (reject. sampling) + inverse sampling:

Key idea: the maximum of the intensity $\lambda_0$ changes over time.
Building blocks to represent different dynamic processes:

Poisson processes:
\[ \lambda^*(t) = \lambda \]

Inhomogeneous Poisson processes:
\[ \lambda(t) = g(t)(1 - N(t)) \]

Self-exciting point processes:
\[ \lambda^*(t) = \mu + \alpha \sum_{t_i \in \mathcal{H}(t)} \kappa_{\omega}(t - t_i) \]

We know **how to fit** them and **how to sample** from them.
Representation:
Temporal Point Processes

1. Intensity function
2. Basic building blocks
3. Superposition
4. Marks and SDEs with jumps
Mutually exciting process

Clustered occurrence affected by neighbors

\[
\lambda^*(t) = \mu + \alpha \sum_{t_i \in \mathcal{H}_b(t)} \kappa_\omega(t - t_i) + \beta \sum_{t_i \in \mathcal{H}_c(t)} \kappa_\omega(t - t_i)
\]
Mutually exciting terminating process

Clustered occurrence affected by neighbors

\[ \lambda^*(t) = (1 - N(t)) \left( g(t) + \beta \sum_{t_i \in \mathcal{H}_c(t)} \kappa_\omega(t - t_i) \right) \]
Representation: Temporal Point Processes

1. Intensity function
2. Basic building blocks
3. Superposition
4. Marks and SDEs with jumps
Marked temporal point processes

Marked temporal point process:
A random process whose realization consists of discrete *marked* events localized in time

\[ N(t) \in \{0\} \cup \mathbb{Z}^+ \]

\[ x(t) \]

\[ y(t) \]

History, \( \mathcal{H}(t) \)

\[ t_1 \quad t_2 \quad t_3 \quad t \]

\[ t = T \]

\[ t_1 \quad t_2 \quad t_3 \quad t \]

\[ t_1 \quad t_2 \quad t_3 \quad t \]
Independent identically distributed marks

Distribution for the marks:

\[ x^*(t_i) \sim p(x) \]

Observations:

1. Marks independent of the temporal dynamics
2. Independent identically distributed (I.I.D.)
Dependent marks: SDEs with jumps

Marks given by stochastic differential equation with jumps:

\[ x(t + dt) - x(t) = dx(t) = f(x(t), t)dt + h(x(t), t)dN(t) \]

Observations:
1. Marks dependent of the temporal dynamics
2. Defined for all values of t
Dependent marks: distribution + SDE with jumps

Distribution for the marks:

\[ x^\star(t_i) \sim p(x^\star \mid x(t)) \quad \Rightarrow \quad dx(t) = f(x(t), t)dt + h(x(t), t)dN(t) \]

Observations:
1. Marks dependent on the temporal dynamics
2. Distribution represents additional source of uncertainty
Mutually exciting + marks

Marks affected by neighbors

\[ dx(t) = f(x(t), t)dt + g(x(t), t)dM(t) \]

- **Drift**: \( f(x(t), t)dt \)
- **Neighbor influence**: \( g(x(t), t)dM(t) \)
Marked TPPs as stochastic dynamical systems

Example: Susceptible-Infected-Susceptible (SIS)

\[ X_i(t) = 0 \quad \text{Susceptible} \]
\[ X_i(t) = 1 \quad \text{Infected} \]
\[ X_i(t) = 0 \quad \text{Susceptible} \]

\[ \mathbb{E}[dY_i(t)] = \lambda Y_i(t)dt \]

Infection rate

\[ \lambda Y_i(t)dt = (1 - X_i(t))\beta \sum_{j \in \mathcal{N}(i)} X_j(t)dt \]

If friends are infected, higher infection rate

\[ \mathbb{E}[dW_i(t)] = \lambda W_i(t)dt \]

Recovery rate

\[ d\lambda W_i(t) = \delta dY_i(t) - \lambda W_i(t)dW_i(t) + \rho dN_i(t) \]

Self-recovery rate when node gets infected

If node recovers, rate to zero

Rate increases if node gets treated

SDE with jumps

It gets infected

It recovers

Node is susceptible

If friends are infected, higher infection rate
Outline of the Seminar

**TEMPORAL POINT PROCESSES (TPPs): INTRO**
1. Intensity function
2. Basic building blocks
3. Superposition
4. Marks and SDEs with jumps

**MODELS & INFERENCE**
1. Modeling event sequences
2. Clustering event sequences
3. Capturing complex dynamics
4. Causal reasoning on event sequences

**RL & CONTROL**
1. Marked TPPs: a new setting
2. Stochastic optimal control
3. Reinforcement learning

Slides/references: learning.mpi-sws.org/tpp-icml18