

# Stochastic Processes



**Week 04 (Version 2.0)**

**Poisson Processes**

**Point Process**

Hamid R. Rabiee

Fall 2022

# Outline of Week 04 Lectures

- Poisson Process
- Point Process

# Recall: Binomial Distribution and its relation to Poisson Distribution

Binomial Distribution:  $X \sim B(n, p)$

probability of exactly  $k$  success in  $n$  trials:

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

$$B(n, p) \xrightarrow[\substack{n \rightarrow \infty \\ np \text{ remains constant}}]{} \text{Poisson}(np)$$

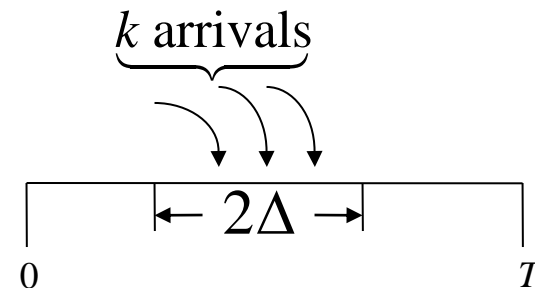
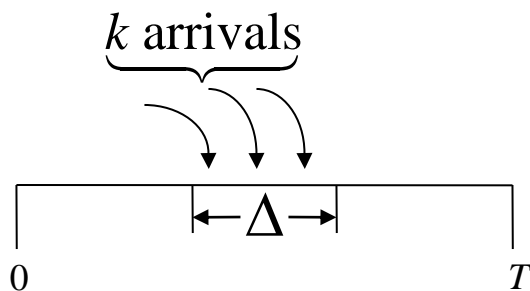
# Poisson Processes

- Recall: Binomial and Poisson distributions:  
Both distributions can be used to model the number of occurrences of some event.
- Recall: **Poisson arrivals** are the limiting behavior of **Binomial random variables**. (Refer to Poisson approximation of Binomial random variables in your text book):

$$P \left\{ \begin{array}{l} "k \text{ arrivals occur in an} \\ \text{interval of duration } \Delta" \end{array} \right\} = e^{-\lambda} \frac{\lambda^k}{k!}, \quad k = 0, 1, 2, \dots$$

Where:

$$\lambda = np = \mu T \cdot \frac{\Delta}{T} = \mu \Delta$$



# Poisson Processes

It follows that:

$$P \left\{ \begin{array}{l} "k \text{ arrivals occur in an} \\ \text{interval of duration } 2\Delta" \end{array} \right\} = e^{-2\lambda} \frac{(2\lambda)^k}{k!}, \quad k = 0, 1, 2, \dots,$$

since in that case:

$$np_1 = \mu T \cdot \frac{2\Delta}{T} = 2\mu\Delta = 2\lambda.$$

# Poisson Processes

- **Poisson arrivals** over an interval form a Poisson random variable whose parameter depends on the duration of that interval.
- Moreover because of the Bernoulli nature of the underlying basic random arrivals, **events over nonoverlapping intervals are independent.**
- We shall use these **two key observations** to define a Poisson process formally.

# Poisson Processes

**Definition:**  $X(t) = n(0, t)$  represents a Poisson process if:

(i) the number of arrivals  $n(t_1, t_2)$  in an interval  $(t_1, t_2)$  of length  $t = t_2 - t_1$  is a Poisson random variable with parameter  $\lambda t$ .

Thus:

$$P\{n(t_1, t_2) = k\} = e^{-\lambda t} \frac{(\lambda t)^k}{k!} \quad k = 0, 1, 2, \dots, t = t_2 - t_1$$

And:

# Poisson Processes

(ii) If the intervals  $(t_1, t_2)$  and  $(t_3, t_4)$  are nonoverlapping, then the random variables  $n(t_1, t_2)$  and  $n(t_3, t_4)$  are independent.

Since  $n(0, t) \sim P(\lambda t)$  we have:

$$E[X(t)] = E[n(0, t)] = \lambda t$$

And:

$$E[X^2(t)] = E[n^2(0, t)] = \lambda t + \lambda^2 t^2$$



# Poisson Processes

To determine the autocorrelation function  $R_{xx}(t_1, t_2)$  let  $t_2 > t_1$  then from (ii) above  $n(0, t_1)$  and  $n(t_1, t_2)$  are **independent Poisson random variables** with parameters  $\lambda t_1$  and  $\lambda(t_2 - t_1)$  respectively.

Thus:

$$E[n(0, t_1)n(t_1, t_2)] = E[n(0, t_1)]E[n(t_1, t_2)] = \lambda^2 t_1 (t_2 - t_1)$$

But:

$$n(t_1, t_2) = n(0, t_2) - n(0, t_1) = X(t_2) - X(t_1)$$

And:

$$E[X(t_1)\{X(t_2) - X(t_1)\}] = R_{xx}(t_1, t_2) - E[X^2(t_1)]$$

We obtain:

$$R_{xx}(t_1, t_2) = \lambda^2 t_1 (t_2 - t_1) + E[X^2(t_1)] = \lambda t_1 + \lambda^2 t_1 t_2$$
$$t_2 \geq t_1$$

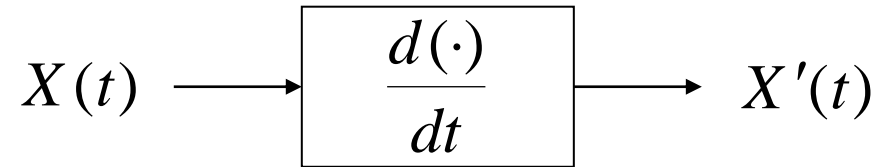
Similarly:

$$R_{xx}(t_1, t_2) = \lambda t_2 + \lambda^2 t_1 t_2$$

Thus:

$$R_{xx}(t_1, t_2) = \lambda^2 t_1 t_2 + \lambda \min(t_1, t_2)$$

## Example:



(Derivative as a LTI system)

Then:

$$\mu_{x'}(t) = \frac{d\mu_x(t)}{dt} = \frac{d\lambda t}{dt} = \lambda, \quad \text{a constant}$$

And:

$$\begin{aligned} R_{xx'}(t_1, t_2) &= \frac{\partial R_{xx}(t_1, t_2)}{\partial t_2} = \begin{cases} \lambda^2 t_1 & t_1 \leq t_2 \\ \lambda^2 t_1 + \lambda & t_1 > t_2 \end{cases} \\ &= \lambda^2 t_1 + \lambda U(t_1 - t_2) \end{aligned}$$

And:

$$R_{xx'}(t_1, t_2) = \frac{\partial R_{xx'}(t_1, t_2)}{\partial t_1} = \lambda^2 + \lambda \delta(t_1 - t_2).$$

# Poisson Processes

Notice that:

- The Poisson process  $X(t)$  *does not* represent a wide sense stationary process.
- Although  $X(t)$  *does not* represent a **wide sense stationary process**, its derivative  $X'(t)$  *does* represent a **wide sense stationary process**.

# Poisson Processes

Since  $X'(t)$  is a **wide sense stationary process**; nonstationary inputs to linear systems *can* lead to **wide sense stationary** outputs, an interesting observation.

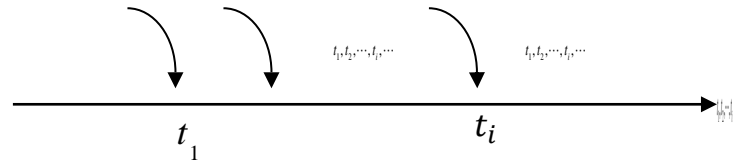
- **Sum of Poisson Processes:**

If  $X_1(t)$  and  $X_2(t)$  represent two **independent Poisson processes**, then their sum  $X_1(t) + X_2(t)$  is also a **Poisson process** with parameter  $(\lambda_1 + \lambda_2)t$ . (Follows from the definition of the Poisson process in (i) and (ii)).

# Poisson Processes

## Random selection of Poisson Points:

Let  $t_1, t_2, \dots, t_i, \dots$  represent random arrival points associated with a Poisson process  $X(t)$  with parameter  $\lambda t$ , and associated with each arrival point, define an independent Bernoulli random variable  $N_i$ , where:



$$P(N_i = 1) = p, \quad P(N_i = 0) = q = 1 - p.$$

# Poisson Processes

Define the processes:

$$Y(t) = \sum_{i=1}^{X(t)} N_i \quad ; \quad Z(t) = \sum_{i=1}^{X(t)} (1 - N_i) = X(t) - Y(t)$$

We claim that both  $Y(t)$  and  $Z(t)$  are **independent Poisson processes** with parameters  $\lambda pt$  and  $\lambda qt$ , respectively, where  $q = 1 - p$ .  
When  $X(t)$  is a Poisson process with parameter  $\lambda t$ .

# Poisson Processes

**Proof:**

$$Y(t) = \sum_{n=k}^{\infty} P\{Y(t) = k \mid X(t) = n\} P\{X(t) = n\}.$$

But given  $X(t) = n$ , we have  $Y(t) = \sum_{i=1}^n N_i \sim B(n, p)$  so that:

$$P\{Y(t) = k \mid X(t) = n\} = \binom{n}{k} p^k q^{n-k}, \quad 0 \leq k \leq n,$$

And:

$$P\{X(t) = n\} = e^{-\lambda t} \frac{(\lambda t)^n}{n!}.$$

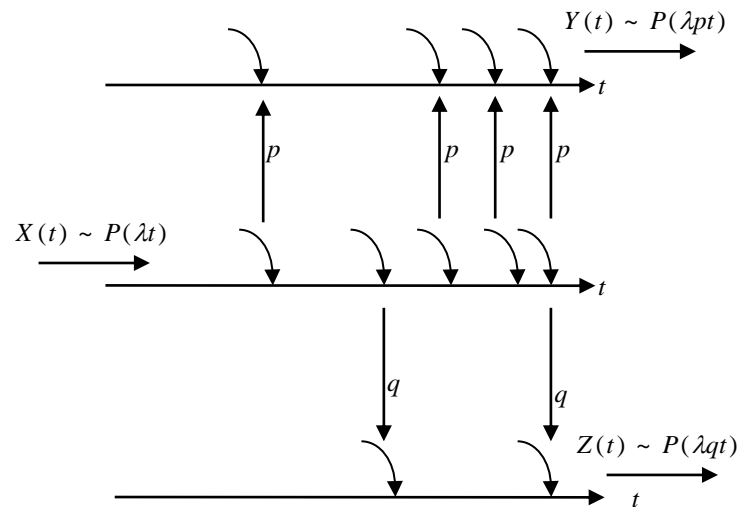


$$\begin{aligned}
P\{Y(t) = k\} &= e^{-\lambda t} \sum_{n=k}^{\infty} \frac{n!}{(n-k)!k!} p^k q^{n-k} \frac{(\lambda t)^n}{n!} = \frac{p^k e^{-\lambda t}}{k!} (\lambda t)^k \underbrace{\sum_{n=k}^{\infty} \frac{(q\lambda t)^{n-k}}{(n-k)!}}_{e^{q\lambda t}} \\
&= (\lambda p t)^k \frac{e^{-(1-q)\lambda t}}{k!} = e^{-\lambda p t} \frac{(\lambda p t)^k}{k!}, \quad k = 0, 1, 2, \dots \\
&\sim P(\lambda p t).
\end{aligned}$$

More generally:

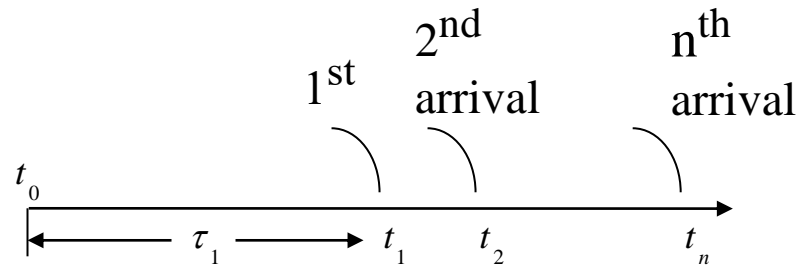
$$\begin{aligned}
P\{Y(t) = k, Z(t) = m\} &= P\{Y(t) = k, X(t) - Y(t) = m\} \\
&= P\{Y(t) = k, X(t) = k + m\} \\
&= P\{Y(t) = k \mid X(t) = k + m\} P\{X(t) = k + m\} \\
&= \binom{k+m}{k} p^k q^m \cdot e^{-\lambda t} \frac{(\lambda t)^{k+m}}{(k+m)!} = e^{-\lambda p t} \underbrace{\frac{(\lambda p t)^k}{k!}}_{P(Y(t)=k)} e^{-\lambda q t} \underbrace{\frac{(\lambda q t)^m}{m!}}_{P(Z(t)=m)} \\
&= P\{Y(t) = k\} P\{Z(t) = m\},
\end{aligned}$$

Notice that  $Y(t)$  and  $Z(t)$  are generated as a result of **random Bernoulli selections** from the **original Poisson process**  $X(t)$ , where each arrival gets tossed over to either  $Y(t)$  with probability  $p$  or to  $Z(t)$  with probability  $q$ . Each such **sub-arrival** stream is also a **Poisson process**. Thus random selection of Poisson points preserve the Poisson nature of the resulting processes. However, deterministic selection from a Poisson process destroys the Poisson property for the resulting processes.



# Inter-arrival Distribution for Poisson Processes

Let  $\tau_1$  denote the time interval (delay) to the first arrival from *any* fixed point  $t_0$ . To determine the probability distribution of the random variable  $\tau_1$ , we argue as follows: Observe that the **event** " $\tau_1 > t$ " is the same as " $n(t_0, t_0+t) = 0$ ", or the **complement event** " $\tau_1 \leq t$ " is the same as the event " $n(t_0, t_0+t) > 0$ ".



# Inter-arrival Distribution for Poisson Processes

Hence the **distribution function** of  $\tau_1$  is given by:

$$\begin{aligned} F_{\tau_1}(t) &\triangleq P\{\tau_1 \leq t\} = P\{X(t) > 0\} = P\{n(t_0, t_0 + t) > 0\} \\ &= 1 - P\{n(t_0, t_0 + t) = 0\} = 1 - e^{-\lambda t} \end{aligned}$$

Its derivative gives **the probability density function** for  $\tau_1$  to be:

$$f_{\tau_1}(t) = \frac{dF_{\tau_1}(t)}{dt} = \lambda e^{-\lambda t}, \quad t \geq 0$$

i.e.  $\tau_1$  is an exponential random variable with parameter  $\lambda$  so that:  $E(\tau_1) = 1/\lambda$ .

# Inter-arrival Distribution for Poisson Processes

Similarly, let  $t_n$  represent the  $n^{\text{th}}$  random arrival point for a Poisson process. Then:

$$\begin{aligned} F_{t_n}(t) &= P\{t_n \leq t\} = P\{X(t) \geq n\} \\ &= 1 - P\{X(t) < n\} = 1 - \sum_{k=0}^{n-1} \frac{(\lambda t)^k}{k!} e^{-\lambda t} \end{aligned}$$

and hence:

$$\begin{aligned} f_{t_n}(x) &= \frac{dF_{t_n}(x)}{dx} = -\sum_{k=1}^{n-1} \frac{\lambda(\lambda x)^{k-1}}{(k-1)!} e^{-\lambda x} + \sum_{k=0}^{n-1} \frac{\lambda(\lambda x)^k}{k!} e^{-\lambda x} \\ &= \frac{\lambda^n x^{n-1}}{(n-1)!} e^{-\lambda x}, \quad x \geq 0 \end{aligned}$$

# Inter-arrival Distribution for Poisson Processes

which represents a Gamma density function. i.e., the **waiting time** to the  $n^{\text{th}}$  **Poisson arrival** has a **Gamma distribution**.

Moreover:

$$t_n = \sum_{i=1}^n \tau_i$$

where  $\tau_i$  is the random inter-arrival duration between the  $(i - 1)^{\text{th}}$  and  $i^{\text{th}}$  events. Notice that  $\tau_i$  s are **independent, identically distributed random variables**. Hence using their characteristic functions, it follows that all inter-arrival durations of a Poisson process are independent exponential random variables with common parameter  $\lambda$ .

i.e.,

$$f_{\tau_i}(t) = \lambda e^{-\lambda t}, \quad t \geq 0.$$

# Inter-arrival Distribution for Poisson Processes

Alternatively, we have  $\tau_1$  is an exponential random variable. By repeating that argument after shifting  $t_0$  to the new point  $t_1$ , we conclude that  $\tau_2$  is an exponential random variable. Thus the sequence  $\tau_1, \tau_2, \dots, \tau_n, \dots$  are **independent exponential random variables** with common p.d.f.

Thus if we systematically tag every  $m^{\text{th}}$  outcome of a Poisson process  $X(t)$  with parameter  $\lambda t$  to generate a new process  $e(t)$ , then the inter-arrival time between any two events of  $e(t)$  is a **Gamma random variable**.

# Inter-arrival Distribution for Poisson Processes

Notice that:

$$E[e(t)] = m / \lambda, \text{ and if } \lambda = m\mu, \text{ then } E[e(t)] = 1 / \mu.$$

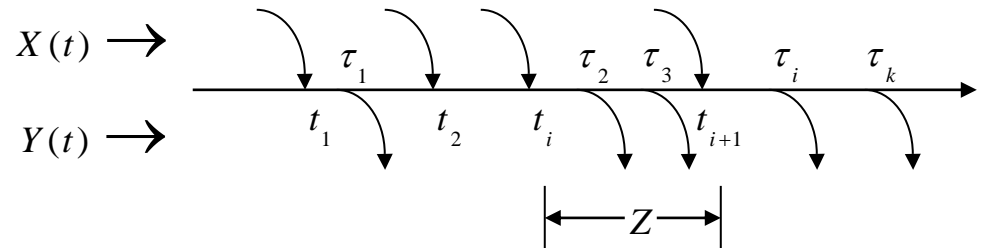
The inter-arrival time of  $e(t)$  in that case represents an **Erlang- $m$  random variable**, and  $e(t)$  is an **Erlang- $m$  process**.

In summary, if Poisson arrivals are randomly redirected to form new queues, then each such queue generates a new Poisson process.



# Poisson Departures between Exponential Inter-arrivals

Let  $X(t) \sim P(\lambda t)$  and  $Y(t) \sim P(\mu t)$  represent two independent Poisson processes called *arrival* and *departure* processes.



Let  $Z$  represent the random interval between *any* two successive arrivals of  $X(t)$ .  $Z$  has an exponential distribution with parameter  $\lambda$ . Let  $N$  represent the number of “departures” of  $Y(t)$  between *any* two successive arrivals of  $X(t)$ . Then from the Poisson nature of the departures we have:

$$P\{N = k \mid Z = t\} = e^{-\mu t} \frac{(\mu t)^k}{k!}.$$

# Poisson Departures between Exponential Inter-arrivals

$$\begin{aligned} P\{N = k\} &= \int_0^{\infty} P\{N = k \mid Z = t\} f_Z(t) dt \\ &= \int_0^{\infty} e^{-\mu t} \frac{(\mu t)^k}{k!} \lambda e^{-\lambda t} dt \\ &= \frac{\lambda}{k!} \int_0^{\infty} (\mu t)^k e^{-(\lambda+\mu)t} dt \\ &= \frac{\lambda}{\lambda+\mu} \left( \frac{\mu}{\lambda+\mu} \right)^k \underbrace{\frac{1}{k!} \int_0^{\infty} x^k e^{-x} dx}_{k!} \\ &= \left( \frac{\lambda}{\lambda+\mu} \right) \left( \frac{\mu}{\lambda+\mu} \right)^k, \quad k = 0, 1, 2, \dots \end{aligned}$$

# Poisson Departures between Exponential Inter-arrivals

The random variable  $N$  has a **geometric distribution**. Thus if customers come in and get out according to two independent Poisson processes at a counter, then the number of arrivals between any two departures has a geometric distribution. Similarly the number of departures between *any* two arrivals also represents another geometric distribution.

## Example

Suppose there are 2 Poisson processes with  $\lambda_1 = 1, \lambda_2 = 2$ .

Find the probability that 2<sup>nd</sup> arrival of first process occurs before 3<sup>rd</sup> arrival of the second process.

### Solution:

Consider the superposition of these two Poisson processes. It is still a Poisson process with  $\lambda = 1 + 2 = 3$ . Also each event of the resulting process is from first process with probability  $\frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{1}{3}$  and otherwise with probability  $\frac{2}{3}$ . So for the 2<sup>nd</sup> arrival of first process to occur before 3<sup>rd</sup> arrival of the second process, we need the first 4 occurrences to cover at least 2 occurrences of the first process:

$$\sum_{k=2}^4 \binom{4}{k} \left(\frac{1}{3}\right)^k \left(\frac{2}{3}\right)^{4-k}$$

## Example: Coupon Collecting

Suppose a cereal manufacturer inserts a sample of one type of coupon randomly into each cereal box. Suppose there are  $n$  such distinct types of coupons. One interesting question is that how many boxes of cereal should one buy on the average in order to collect at least one coupon of each kind?

## Example: Coupon Collecting

We shall reformulate the above problem in terms of Poisson processes. Let  $X_1(t), X_2(t), \dots, X_n(t)$  represent  $n$  *independent* identically distributed Poisson processes with common parameter  $\lambda t$ . Let  $t_{i1}, t_{i2}, \dots$  represent the first, second, ... random arrival instants of the process  $X_i(t)$ ,  $i = 1, 2, \dots, n$ . They will correspond to the first, second, ... appearance of the  $i^{\text{th}}$  type coupon in the above problem. Let:

$$X(t) \triangleq \sum_{i=1}^n X_i(t),$$

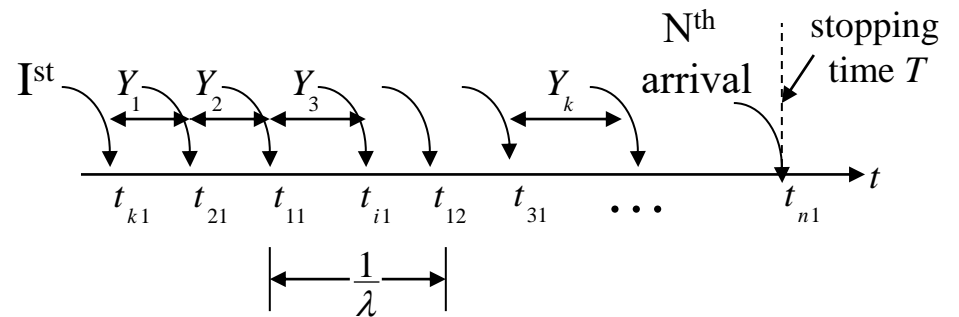
so that the sum  $X(t)$  is also a Poisson process with parameter  $\mu t$ , where

$$\mu = n\lambda.$$

# Example: Coupon Collecting

$1/\lambda$  represents: The average inter-arrival duration between any two arrivals of  $X_i(t), i = 1, 2, \dots, n$ , whereas:

$1/\mu$  represents the average inter-arrival time for the combined sum process  $X(t)$ .



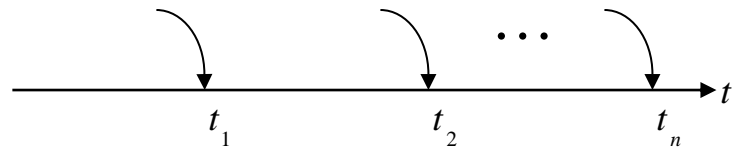
# Bulk Arrivals and Compound Poisson Processes

In an ordinary Poisson process  $X(t)$ , only one event occurs at any arrival instant. Instead suppose a random number of events  $C_i$  occur simultaneously as a cluster at every arrival instant of a Poisson process. If  $X(t)$  represents the total number of all occurrences in the interval  $(0, t)$ , then  $X(t)$  represents a **compound Poisson process**, or a **bulk arrival process**.

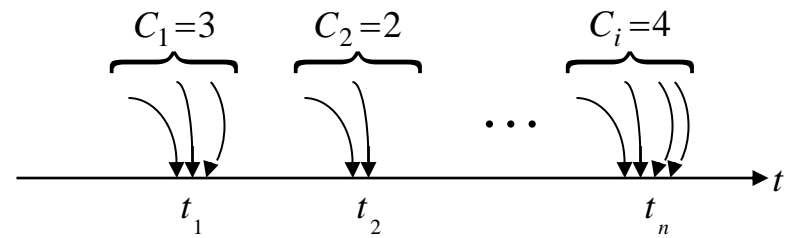


# Bulk Arrivals and Compound Poisson Processes

Inventory orders, arrivals at an airport queue, tickets purchased for a show, etc. follow this process (when things happen, they happen in a bulk, or a bunch of items are involved.)



(a) Poisson Process



(b) Compound Poisson Process

Let:

$$p_k = P\{C_i = k\}, \quad k = 0, 1, 2, \dots$$

represent the common probability mass function for the occurrence in any cluster  $C_i$ . Then the compound process  $X(t)$  satisfies:

$$X(t) = \sum_{i=1}^{N(t)} C_i,$$

where  $N(t)$  represents an ordinary Poisson process with parameter  $\lambda$ .  
Let:

$$P(z) = E\{z^{C_i}\} = \sum_{k=0}^{\infty} p_k z^k$$

represent the moment generating function associated with the cluster Statistics. Then the moment generating function of the compound Poisson process  $X(t)$  is given by:

$$\begin{aligned}
 \phi_x(z) &= \sum_{n=0}^{\infty} z^n P\{X(t) = n\} = E\{z^{X(t)}\} \\
 &= E\{E[z^{X(t)} \mid N(t) = k]\} = E[E\{z^{\sum_{i=1}^k C_i} \mid N(t) = k\}] \\
 &= \sum_{k=0}^{\infty} (E\{z^{C_i}\})^k P\{N(t) = k\} \\
 &= \sum_{k=0}^{\infty} P^k(z) e^{-\lambda t} \frac{(\lambda t)^k}{k!} = e^{-\lambda t(1-P(z))}
 \end{aligned}$$

If we let:

$$P^k(z) \triangleq \left( \sum_{n=0}^{\infty} p_n z^n \right)^k = \sum_{n=0}^{\infty} p_n^{(k)} z^n$$

where  $\{p_n^{(k)}\}$  represents the  $k$  fold convolution of the sequence  $\{p_n\}$  with itself, we obtain:

$$P\{X(t) = n\} = \sum_{k=0}^{\infty} e^{-\lambda t} \frac{(\lambda t)^k}{k!} p_n^{(k)}$$

The above, represents the probability that there are  $n$  arrivals in the interval  $(0, t)$  for a compound Poisson process  $X(t)$ .

We can rewrite  $\phi_X(z)$  also as:

$$\phi_X(z) = e^{-\lambda_1 t(1-z)} e^{-\lambda_2 t(1-z^2)} \dots e^{-\lambda_k t(1-z^k)} \dots$$

where  $\lambda_k = p_k \lambda$ , which shows that the compound Poisson process can be expressed as the sum of integer-scaled independent Poisson processes  $m_1(t), m_2(t), \dots$ . Thus:

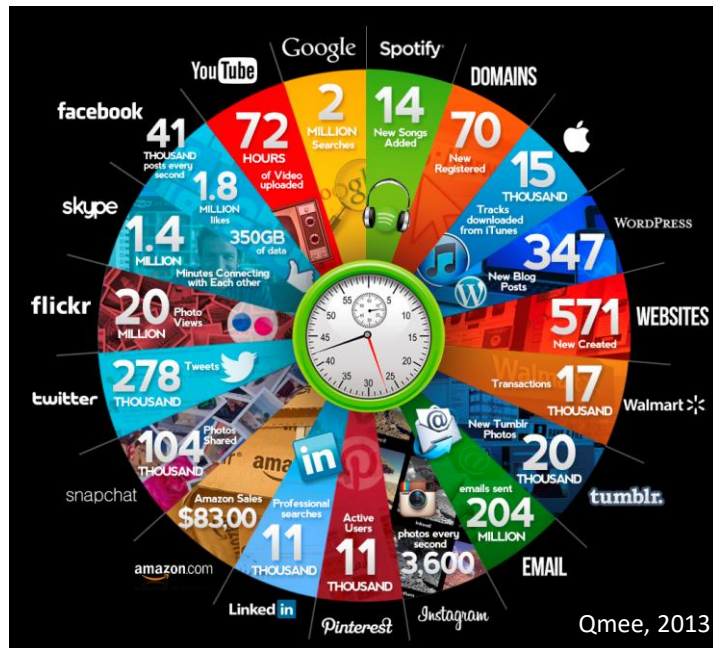
$$X(t) = \sum_{k=1}^{\infty} k m_k(t).$$

More generally, every linear combination of independent Poisson processes represents a compound Poisson process.

# Outline of Week 04 Lectures

- Poisson Process
- Point Process

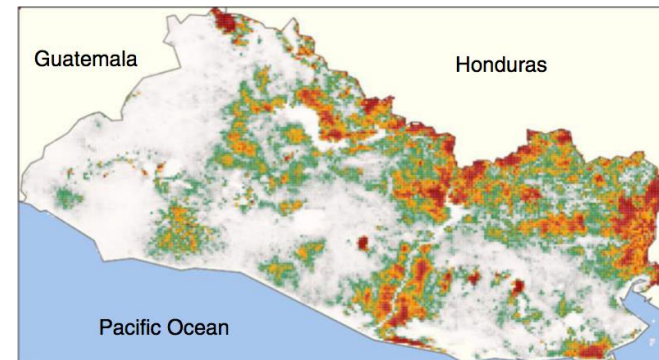
# Many discrete *events* in continuous time



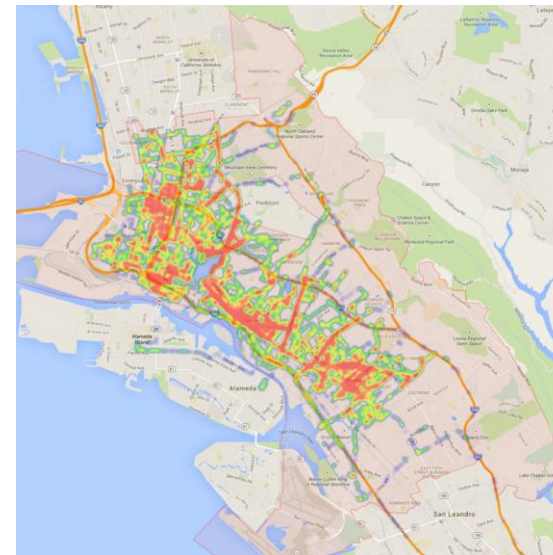
**Online actions**



**Financial trading**



**Disease dynamics**



**Mobility dynamics**

Variety of processes behind these events

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

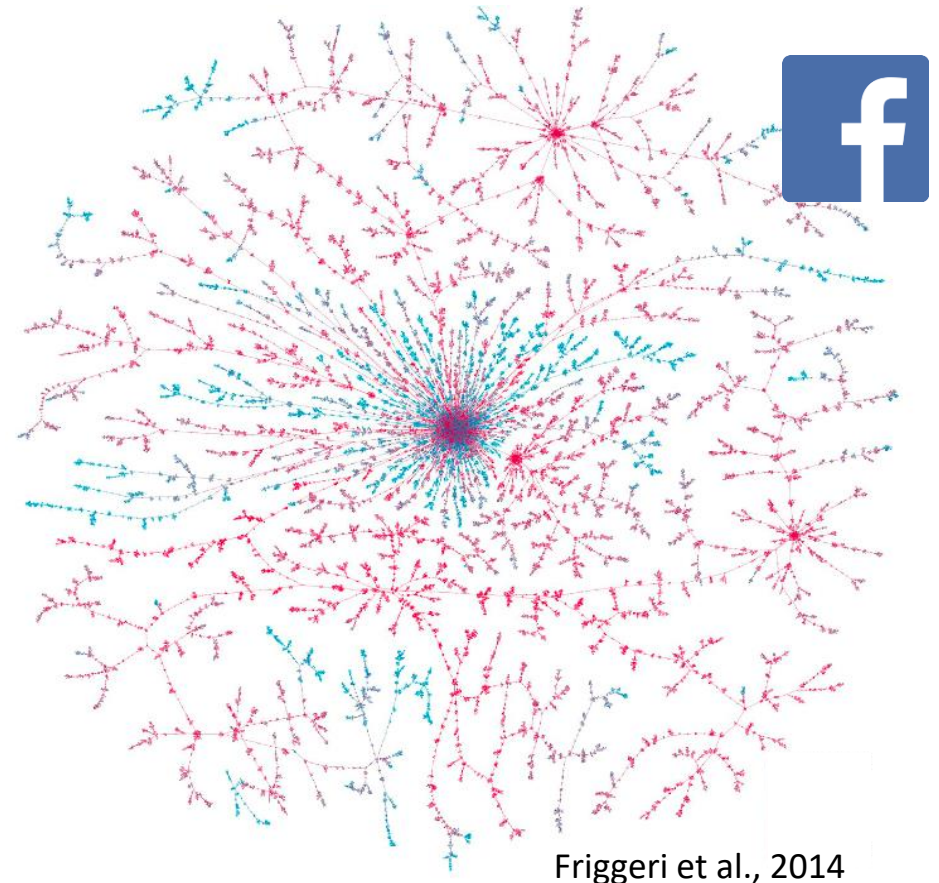
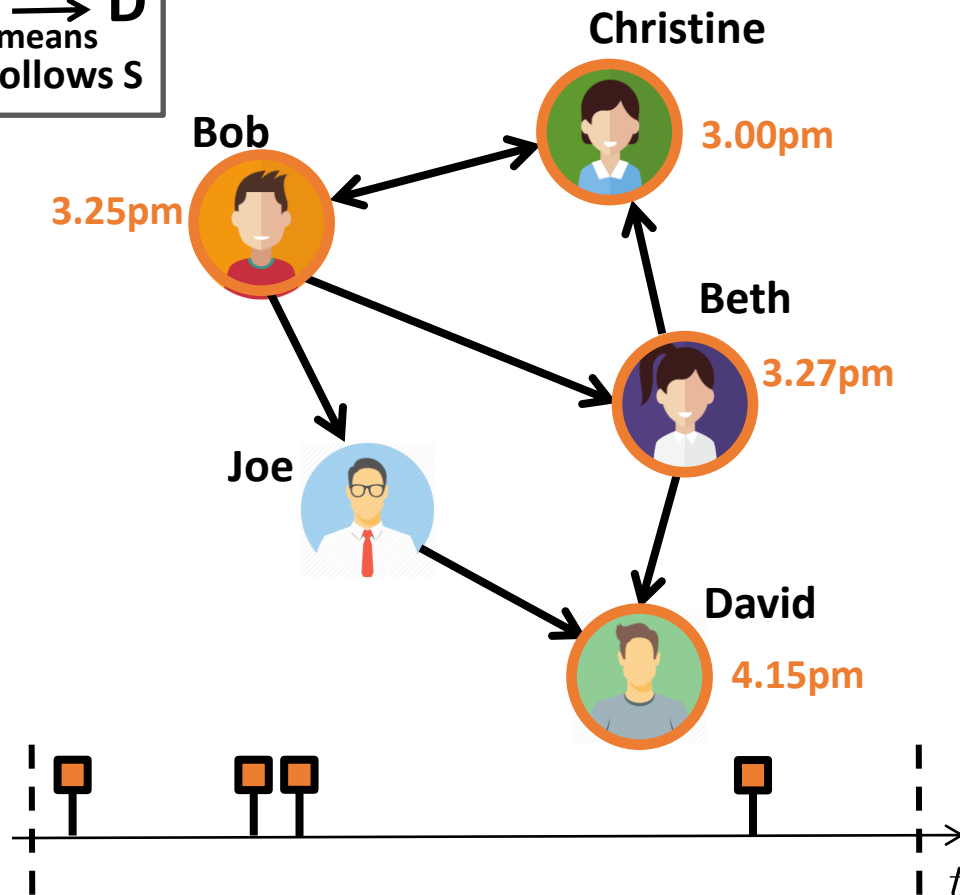


...in a wide range of temporal scales.



# Example I: Information propagation

$S \rightarrow D$   
means  
D follows S



**They can have an impact  
in the off-line world**

**theguardian**

Click and elect: how fake news helped Donald Trump win a real election



### Barack Obama

From Wikipedia, the free encyclopedia

*"Barack" and "Obama" redirect here. For his father, see Barack Obama Sr. For other uses of "Barack", see Barack (disambiguation) (disambiguation).*

**Barack Hussein Obama II** (), current President of the United States. He was president of the Harvard Law School, a civil rights attorney and taught at the United States House of Representatives.

### Barack Obama: Revision history

- 03:41, 28 November 2016 Ranze (talk | contribs) .. (301,105 bytes) (+18) .. (E)
- 03:32, 28 November 2016 Xin Deui (talk | contribs) .. (301,087 bytes) (-68) .. (E)
- 00:57, 28 November 2016 SporkBot (talk | contribs) m .. (301,155 bytes) (-37) .. (E)
- 07:03, 27 November 2016 Saiph121 (talk | contribs) .. (301,192 bytes) (+25) .. (E)

03:21, 20 September 2016

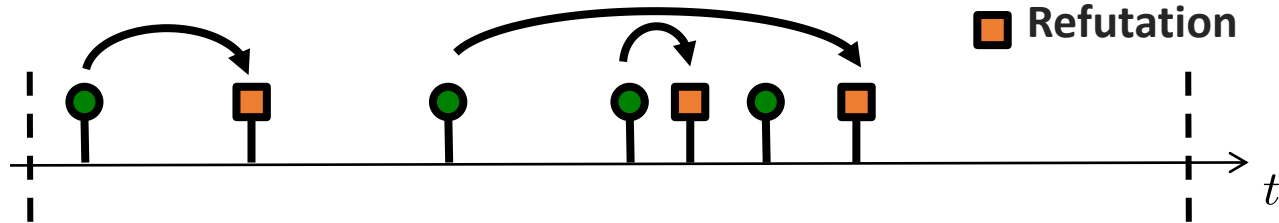
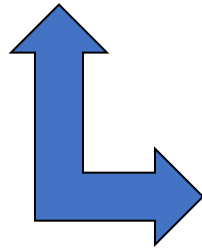
is a **Kenyan** politician



possible vandalism by **MLM2016**

is an American politician

- Addition
- Refutation



Moving to Australia Working in Australia Study abroad in Australia +4

### What are the pros and cons of living in Australia?

Answer Request Follow 109 Comment Share 9 Downvote

I have studied, worked and lived in Australia as an Internat...  
employee, business owner and a citizen.

Upvote | 150

I have experienced this country in all the ways possible, you...  
However, I firmly believe that there are definitely more pros...  
Australia but still I have mentioned below a few... challenges and...  
benefits.



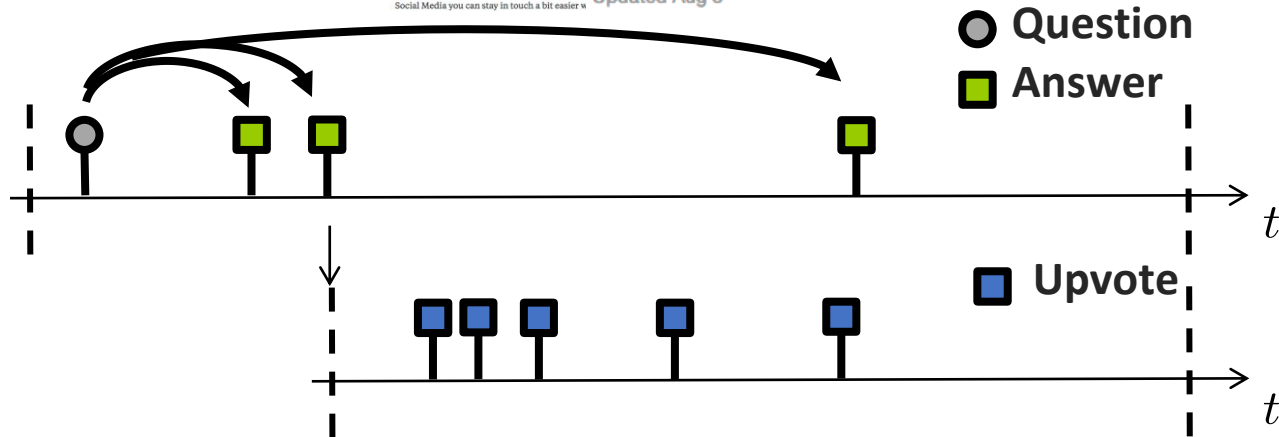
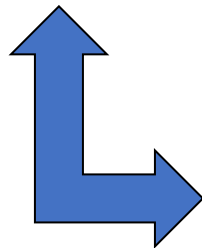
**M Sharma**, Lived in Australia as Migrant, Student, Worker, Business Owner & Family Man

Updated Aug 3

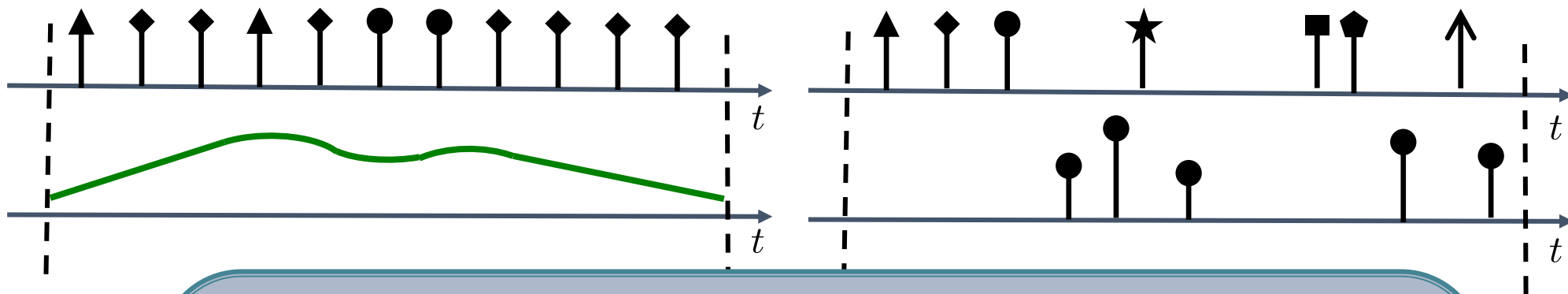
Hope it helps! :)

Possible Challenges

- Language problem for those who don't speak English
- Not having your family and friends around could be a challenge
- Society is more and more connected and thanks to Social Media you can stay in touch a bit easier



# Aren't these event traces just time series?



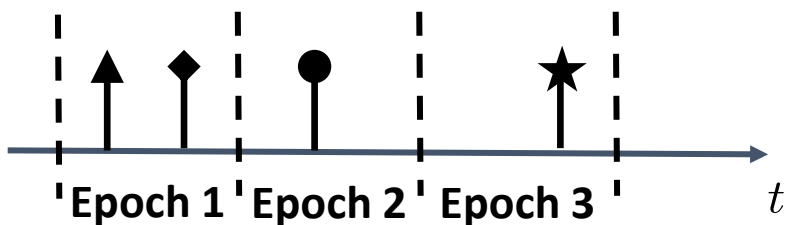
Disc

Wh

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

och?

What about time-related queries?



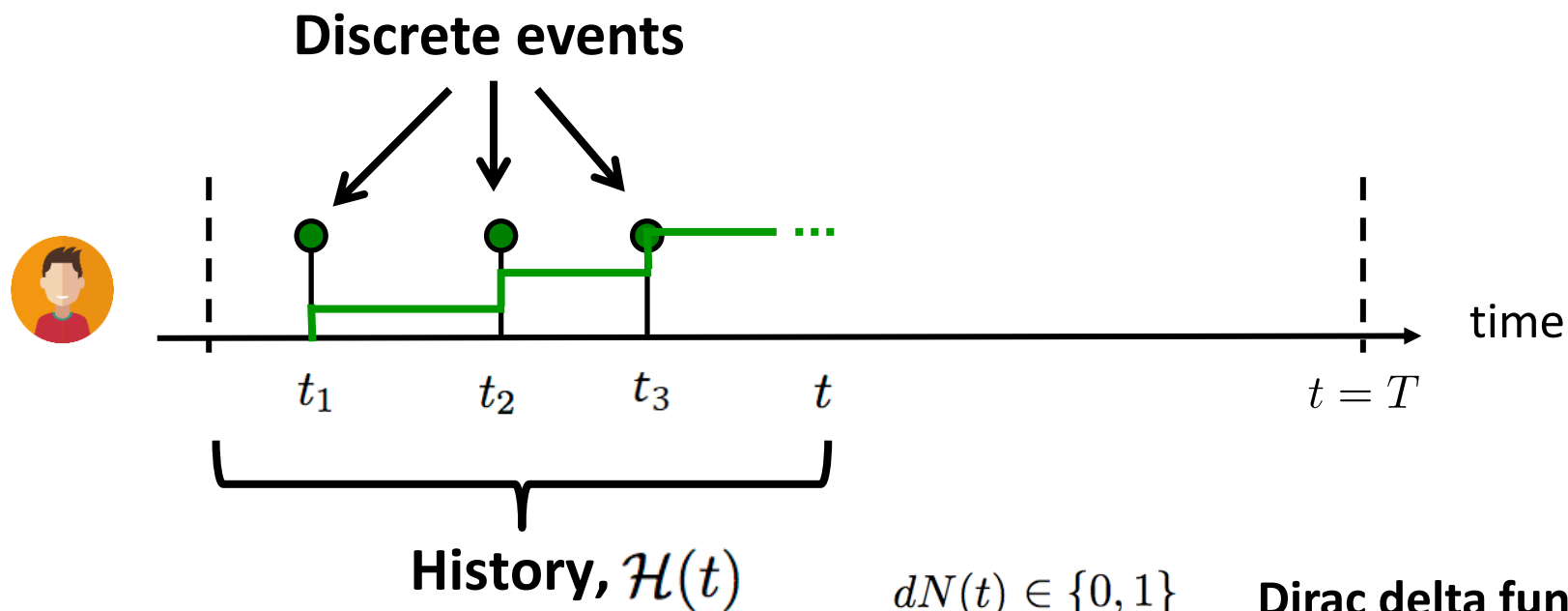
# Temporal Point Processes (TPPs): Introduction

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

# Temporal point processes

## Temporal point process:

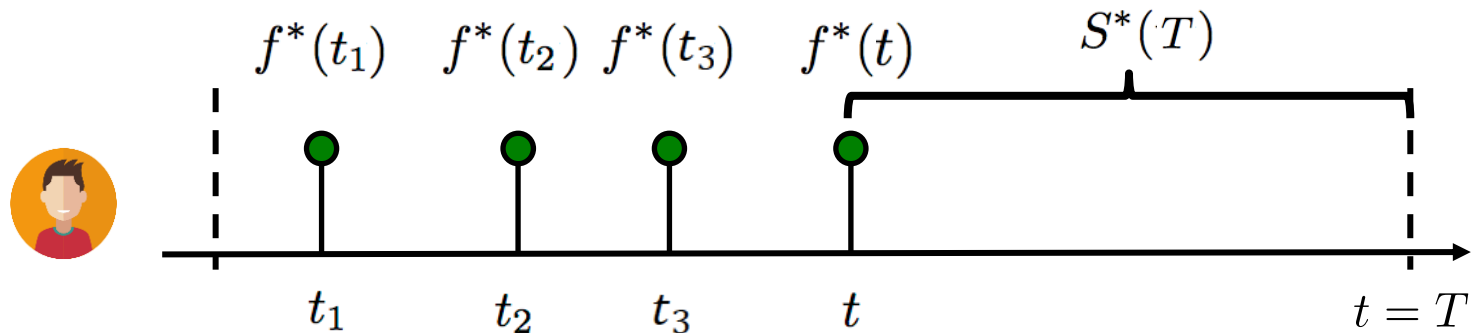
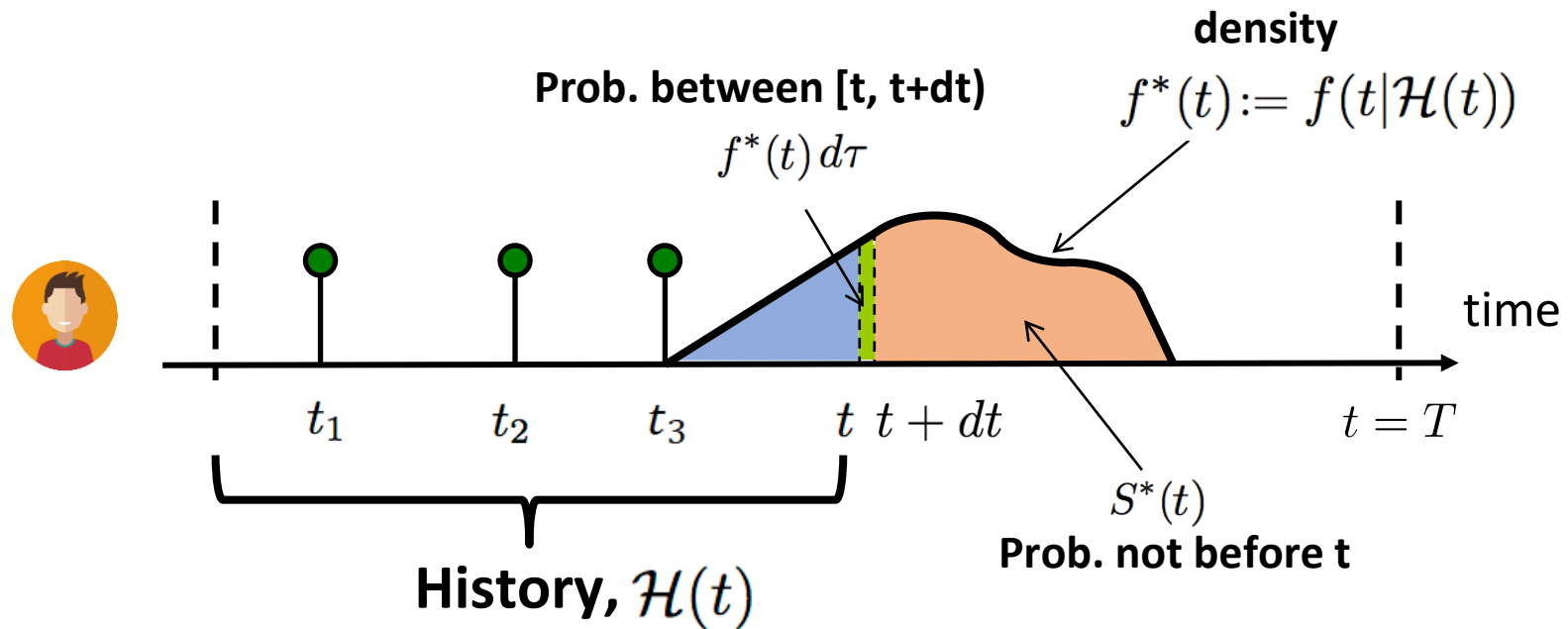
A random process whose realization consists of discrete events localized in time  $\mathcal{H} = \{t_i\}$



**Formally:**  $N(t) = \int_0^t dN(s) \Rightarrow dN(t) = \sum_{t_i \in \mathcal{H}} \delta(t - t_i) dt$

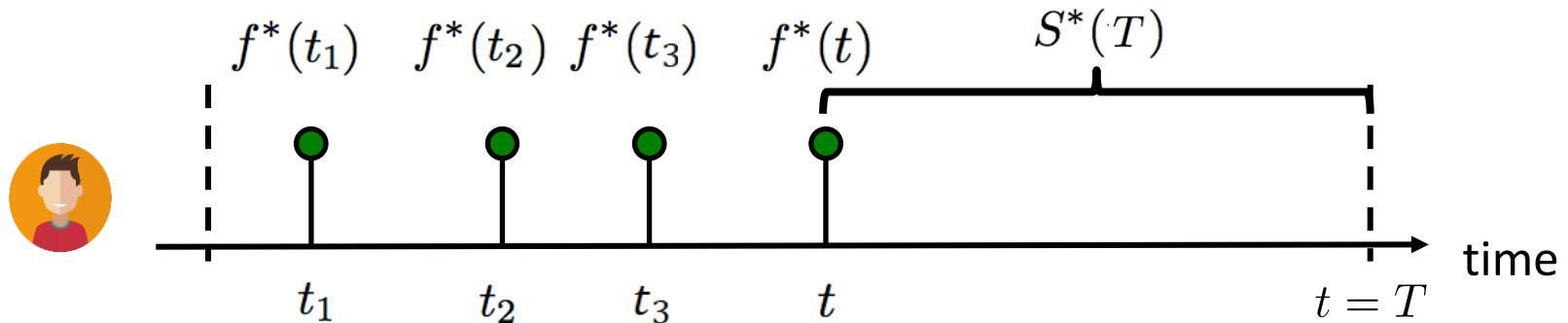
$dN(t) \in \{0, 1\}$       Dirac delta function

# Model time as a random variable



Likelihood of a timeline:  $f^*(t_1) f^*(t_2) f^*(t_3) f^*(t) S^*(T)$

# Problems of density parametrization (I)

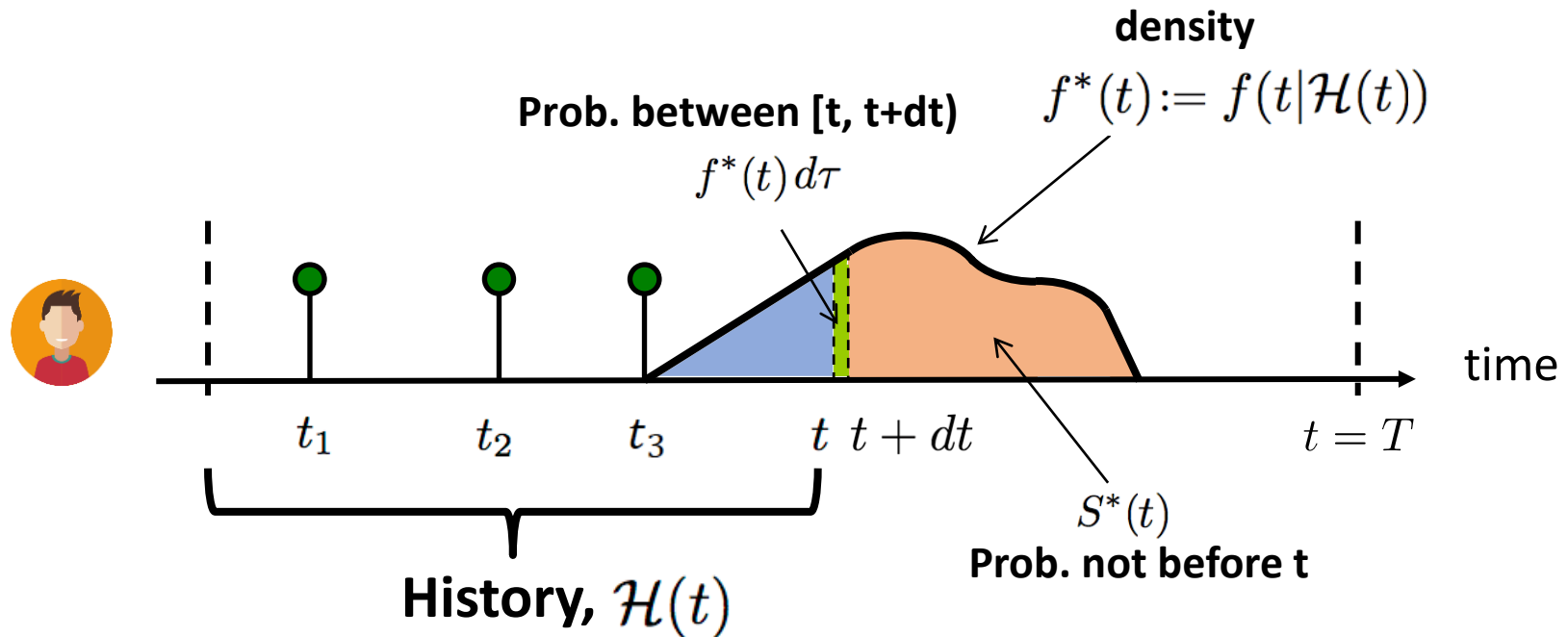


$$\begin{array}{cccccc}
 f^*(t_1) & f^*(t_2) & f^*(t_3) & f^*(t) & S^*(T) & \\
 \nearrow & \nearrow & \uparrow & \nearrow & \nwarrow & \\
 \frac{\exp\langle w, \psi^*(t_1) \rangle}{Z} & & \frac{\exp\langle w, \psi^*(t_3) \rangle}{Z} & & & \\
 & \frac{\exp\langle w, \psi^*(t_2) \rangle}{Z} & & \frac{\exp\langle w, \psi^*(t) \rangle}{Z} & & 1 - \int_t^T \frac{\exp\langle w, \psi^*(\tau) \rangle}{Z} d\tau
 \end{array}$$

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:**

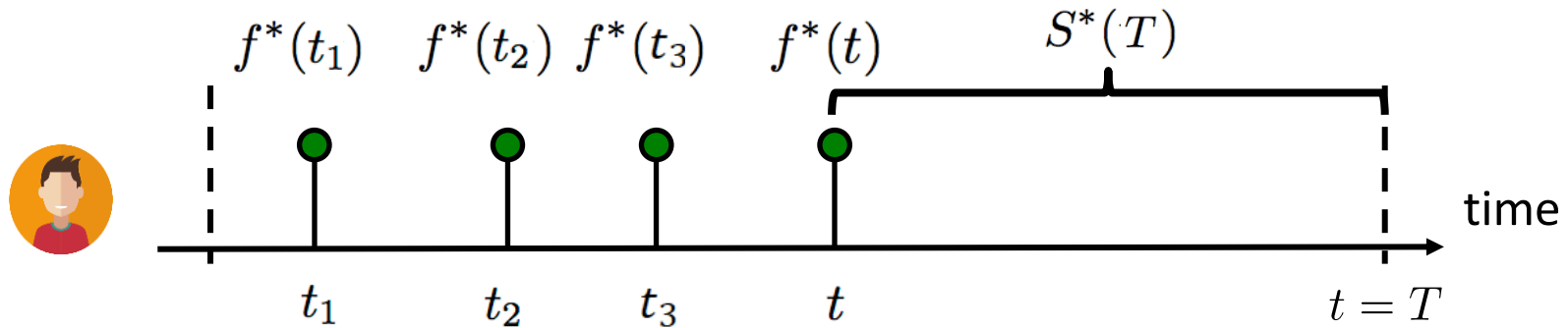
Probability between  $[t, t+dt)$  but not before  $t$

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

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



# Advantages of intensity parametrization (I)



$$\lambda^*(t_1) \lambda^*(t_2) \lambda^*(t_3) \lambda^*(t) \exp\left(-\int_0^T \lambda^*(\tau) d\tau\right)$$

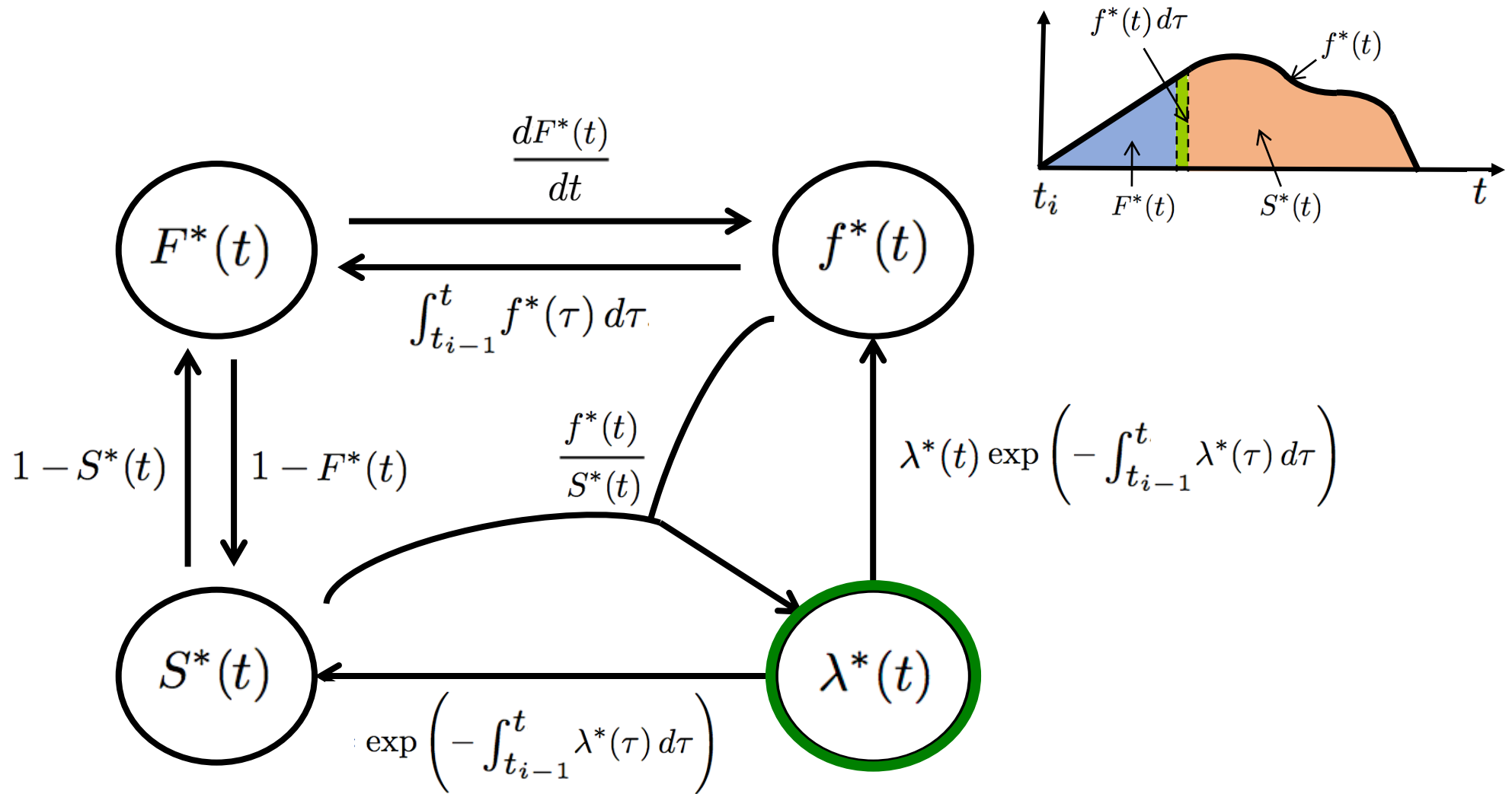
$$\langle w, \phi^*(t_1) \rangle \quad \langle w, \phi^*(t_2) \rangle \quad \langle w, \phi^*(t_3) \rangle \quad \langle w, \phi^*(t) \rangle \quad \exp\left(-\int_0^T \langle w, \phi^*(\tau) \rangle d\tau\right)$$

Arrows indicate the mapping from the inner products  $\langle w, \phi^*(t_i) \rangle$  to the corresponding  $\lambda^*(t_i)$  terms in the first equation, and from the integral of inner products to the exponential term in the second equation.

**Suitable for model design and interpretable:**

1. Intensities only need to be nonnegative
2. Easy to combine timelines

# Relation between $f^*$ , $F^*$ , $S^*$ , $\lambda^*$

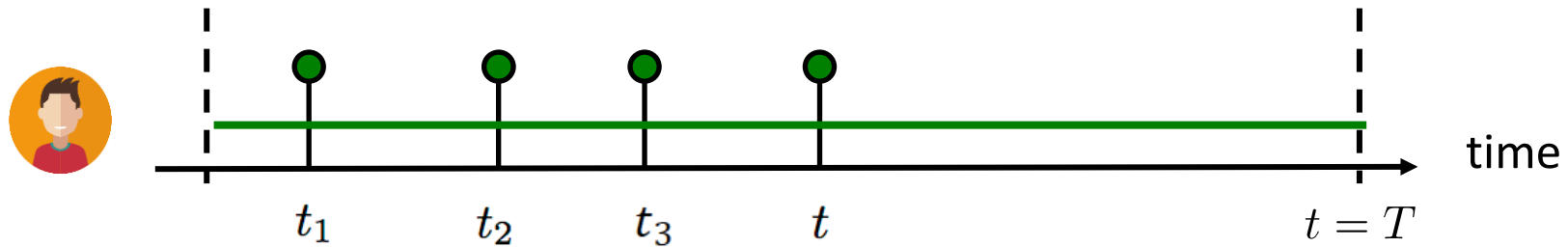


# Representation:

## Temporal Point Processes

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

# Poisson process



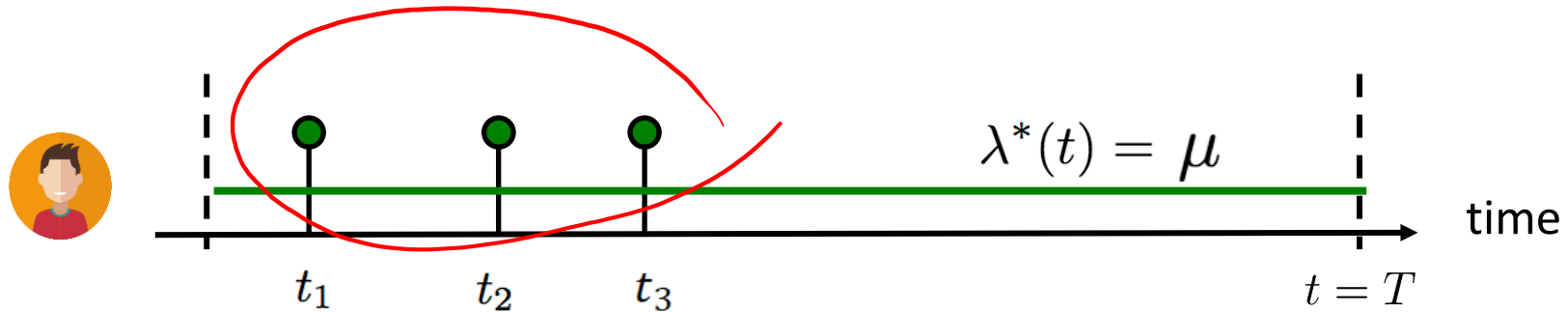
## Intensity of a Poisson process

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

### 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:**

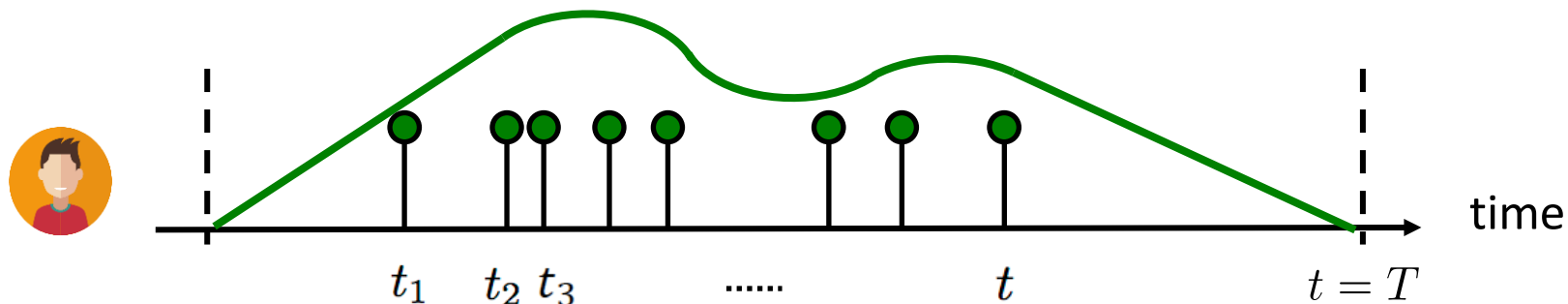
$$\mu^* = \operatorname{argmax}_{\mu} 3 \log \mu - \mu T = \frac{3}{T}$$

**Sampling using inversion sampling:**

$$t \sim \underbrace{\mu \exp(-\mu(t - t_3))}_{f_t^*(t)} \quad \Rightarrow \quad t = \underbrace{-\frac{1}{\mu} \log(1 - u) + t_3}_{F_t^{-1}(u)}$$

*Uniform(0, 1)*  
↓  
 $u$

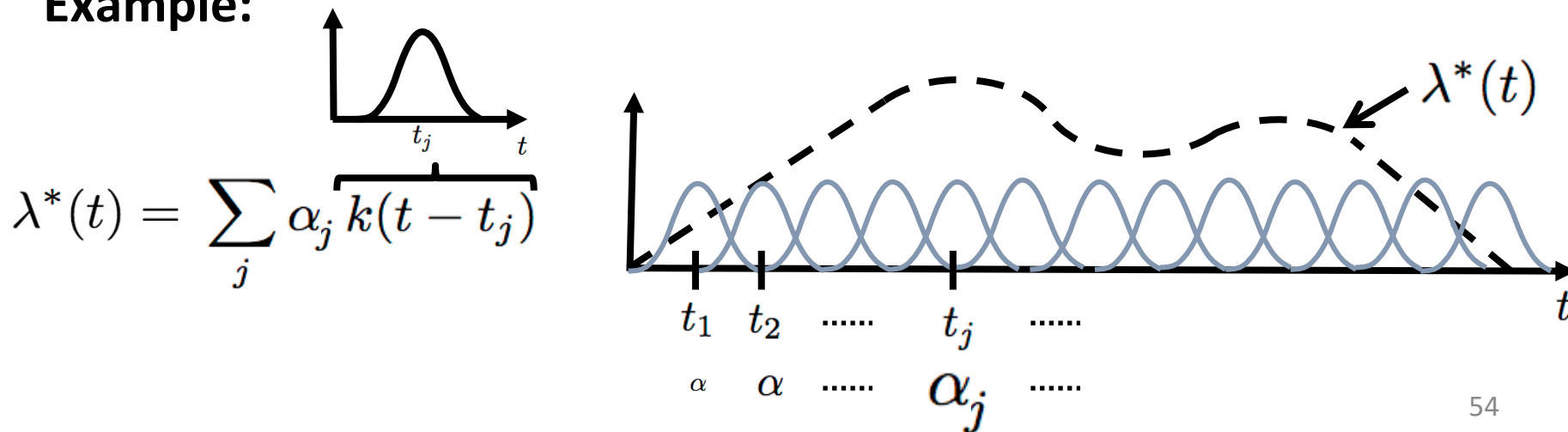
# Inhomogeneous Poisson process



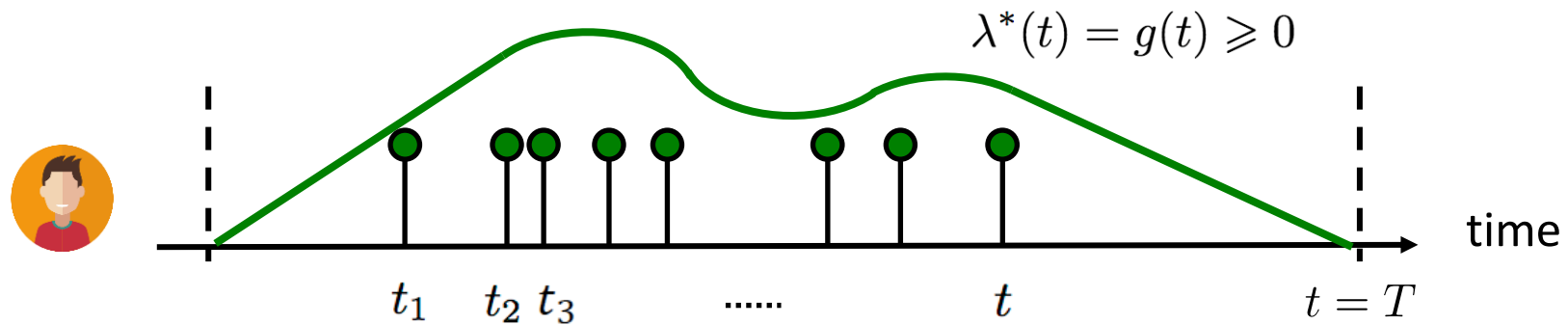
## Intensity of an inhomogeneous Poisson process

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

Example:



# Fitting & sampling from inhomogeneous Poisson

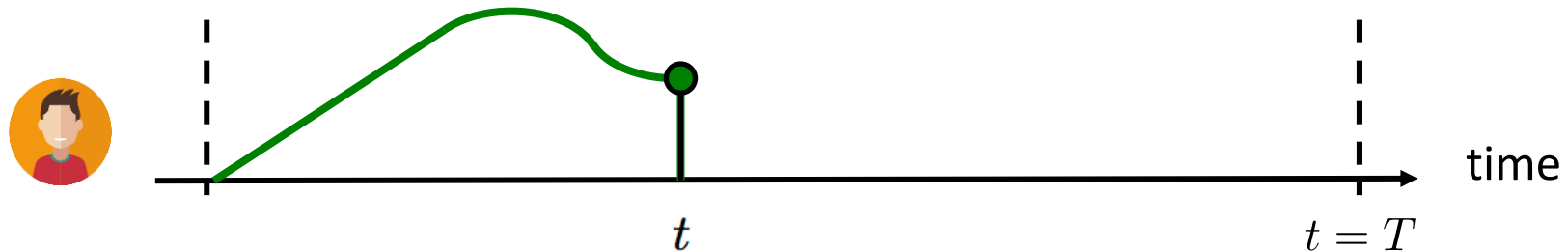


**Fitting by maximum likelihood:**  $\underset{g(t)}{\text{maximize}} \sum_{i=1}^n \log g(t_i) - \int_0^T g(\tau) d\tau.$

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

1. Sample  $t$  from Poisson process with intensity  $\mu$  using inverse sampling
  2. Generate  $u_2 \sim \text{Uniform}(0, 1)$
  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

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

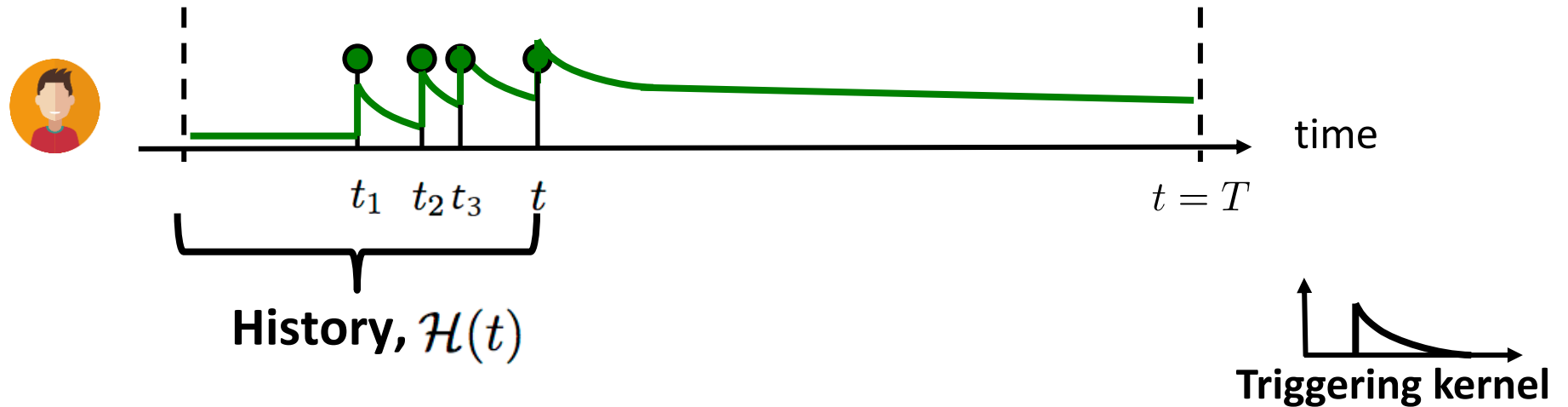
### Observations:

1. Limited number of occurrences

*Try sampling  
and fitting!*



# Self-exciting (or Hawkes) process



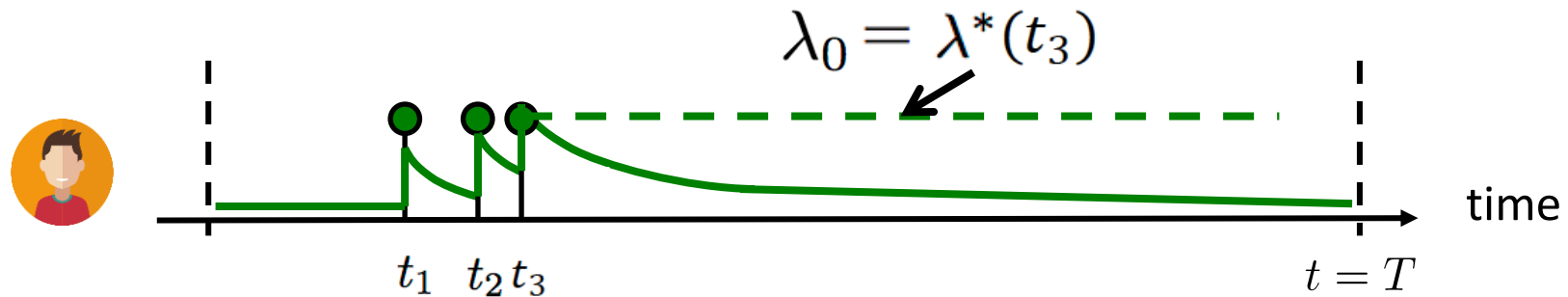
Intensity of self-exciting  
(or Hawkes) process:

$$\begin{aligned}\lambda^*(t) &= \mu + \alpha \sum_{t_i \in \mathcal{H}(t)} \kappa_\omega(t - t_i) \\ &= \mu + \alpha \kappa_\omega(t) \star dN(t)\end{aligned}$$

Observations:

1. Clustered (or bursty) occurrence of events
2. Intensity is stochastic and history dependent

# Fitting a Hawkes process from a recorded timeline



## Fitting by maximum likelihood:

$$\text{maximize}_{\mu, \alpha} \left. \sum_{i=1}^n \log \lambda^*(t_i) - \int_0^T \lambda^*(\tau) d\tau \right\} \begin{array}{l} \text{The max. likelihood} \\ \text{is jointly convex} \\ \text{in } \mu \text{ and } \alpha \end{array}$$

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

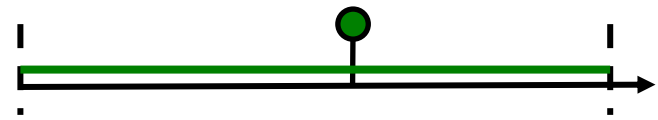
Key idea: the maximum of the intensity  $\lambda_0$  changes over time

# Summary

**Building blocks to represent different dynamic processes:**

Poisson processes:

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



Inho

Term

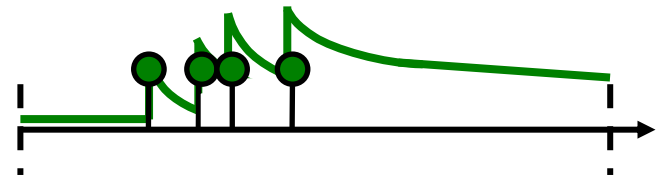
**We know how to fit them  
and how to sample from them**

$$\lambda^*(t) = g(t)(1 - N(t))$$



Self-exciting point processes:

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

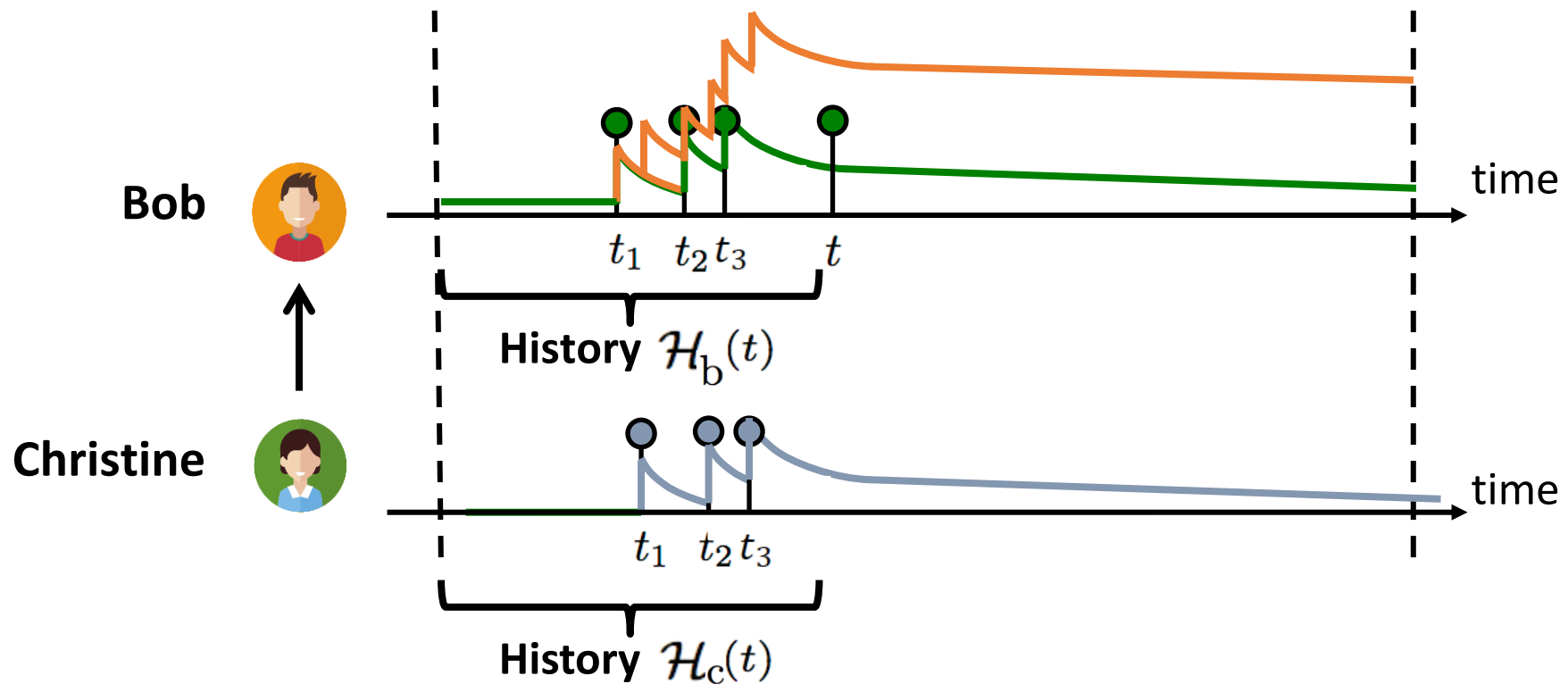


# Representation:

## Temporal Point Processes

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

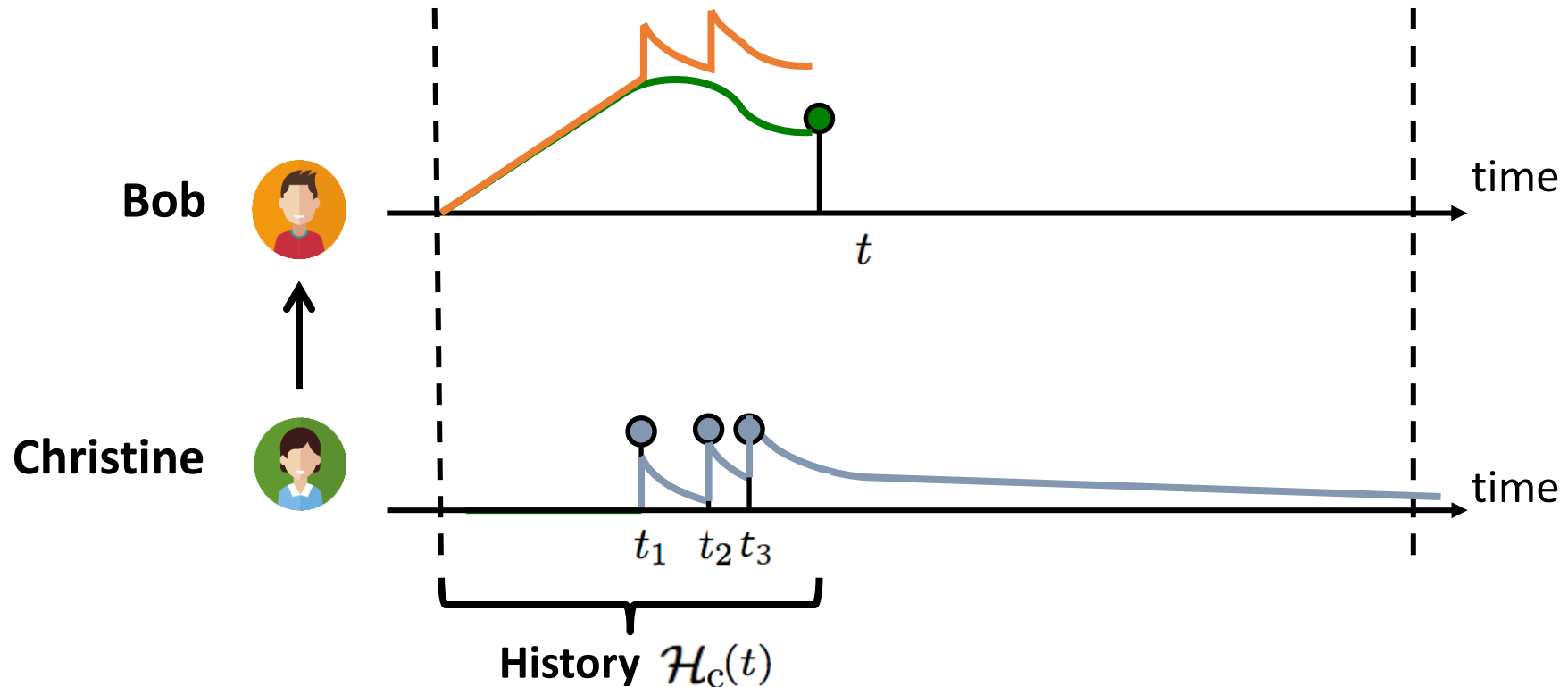
# Mutually exciting process



## 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**

$$\lambda^*(t) = (1 - N(t)) \left( g(t) + \beta \sum_{t_i \in \mathcal{H}_c(t)} \kappa_\omega(t - t_i) \right)$$

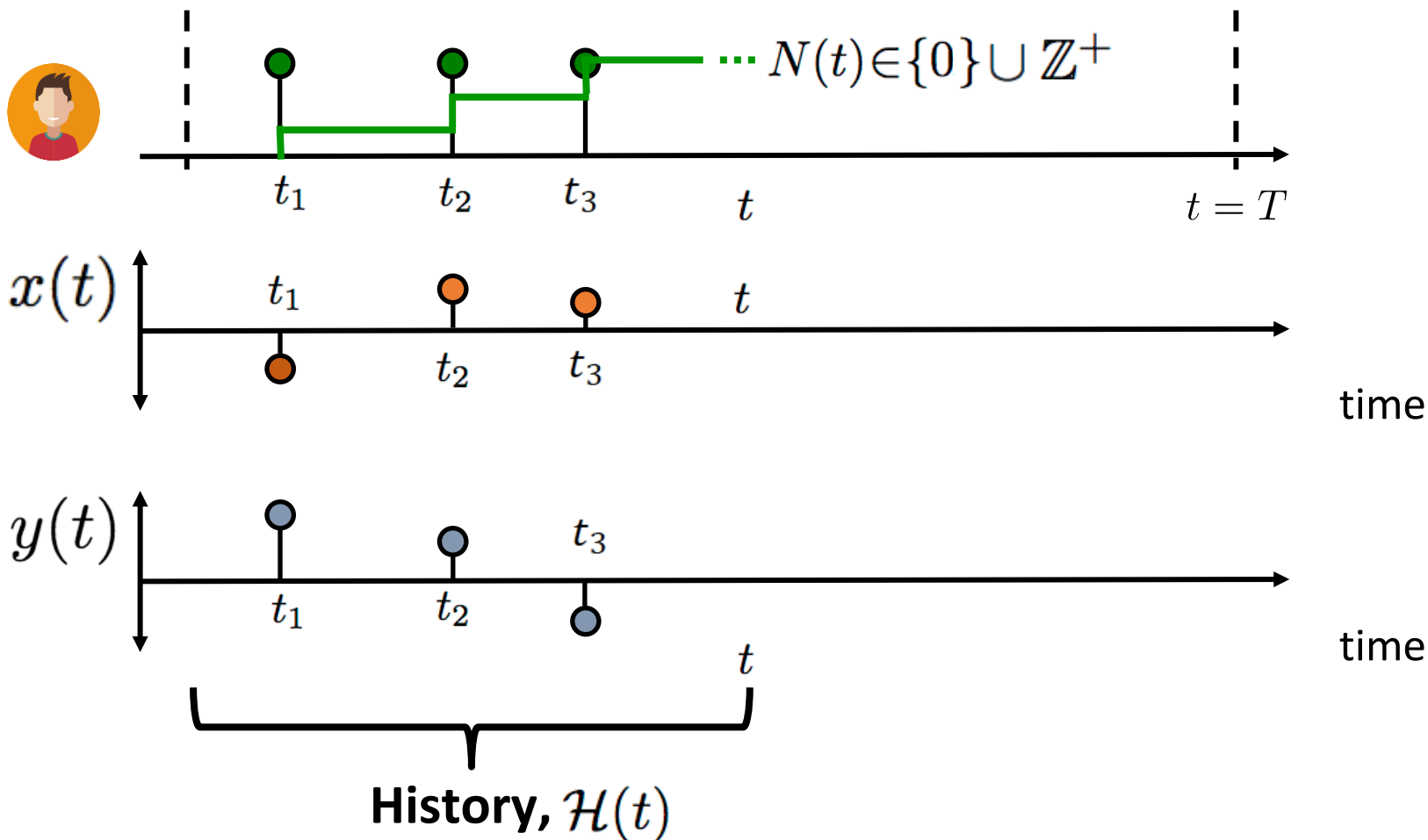
# Representation: Temporal Point Processes

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

# Marked temporal point processes

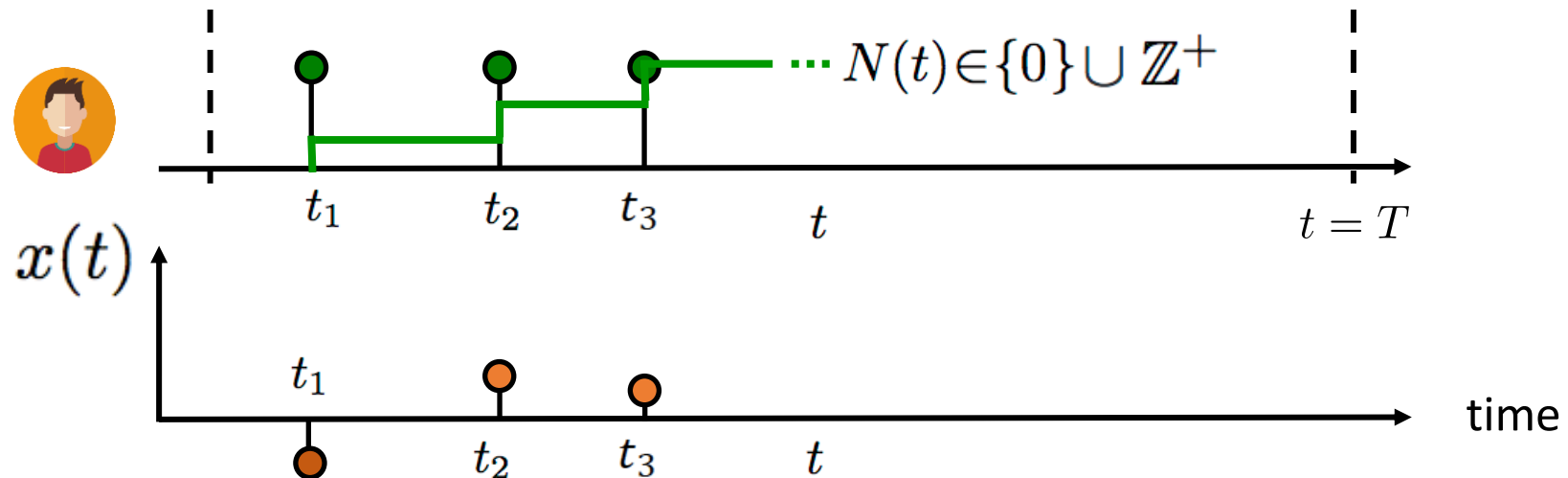
## Marked temporal point process:

A random process whose realization consists of **discrete marked events localized in time**





# Independent identically distributed marks



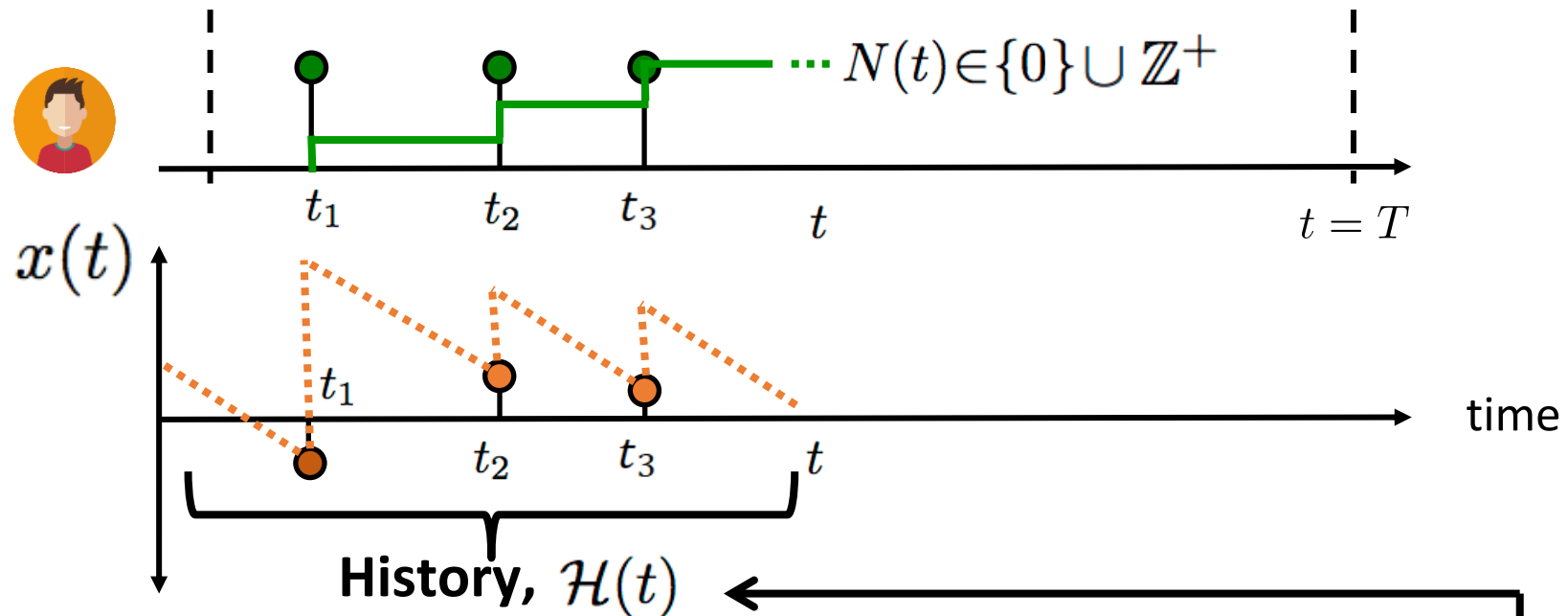
**Distribution for the marks:**

$$x^*(t_i) \sim p(x)$$

**Observations:**

- 1. Marks independent of the temporal dynamics**
- 2. Independent identically distributed (I.I.D.)**

# Dependent marks: SDEs with jumps



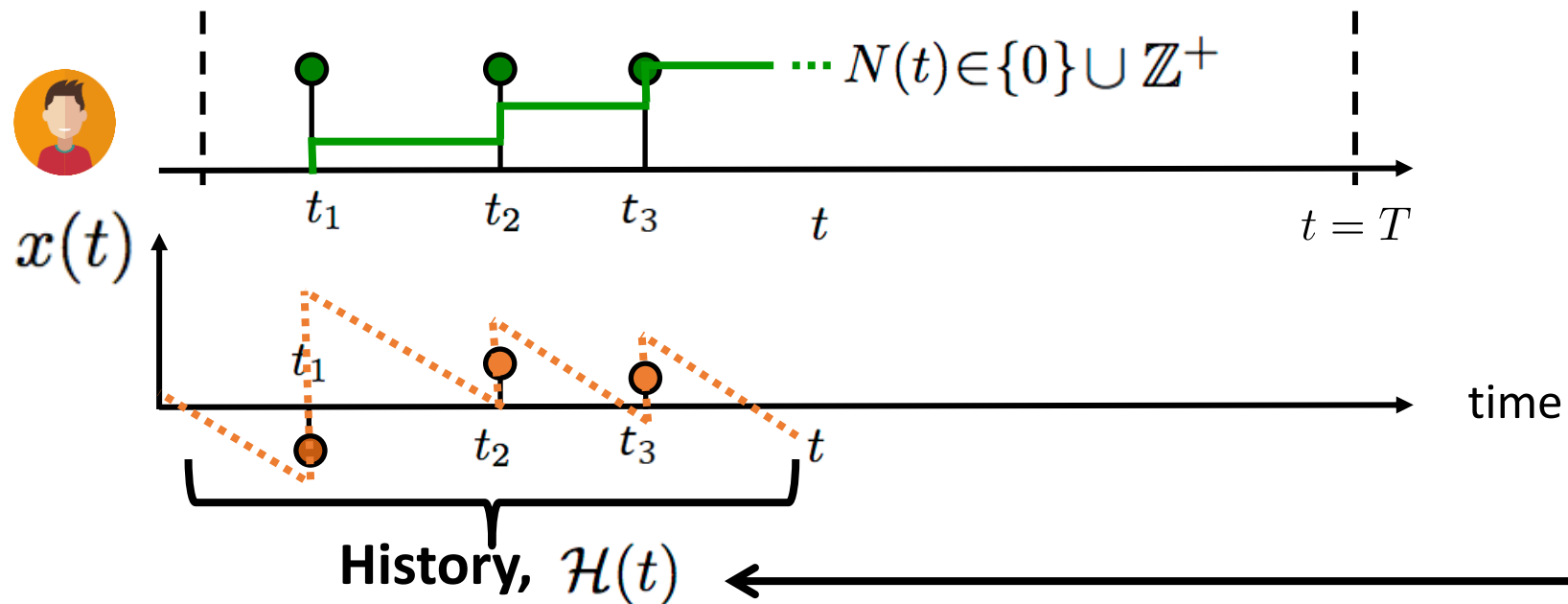
Marks given by stochastic differential equation with jumps:

$$x(t + dt) - x(t) = dx(t) = \underbrace{f(x(t), t)dt}_{\text{Drift}} + \underbrace{h(x(t), t)dN(t)}_{\text{Event influence}}$$

Observations:

1. Marks dependent of the temporal dynamics
2. Defined for all values of  $t$

# Dependent marks: distribution + SDE with jumps



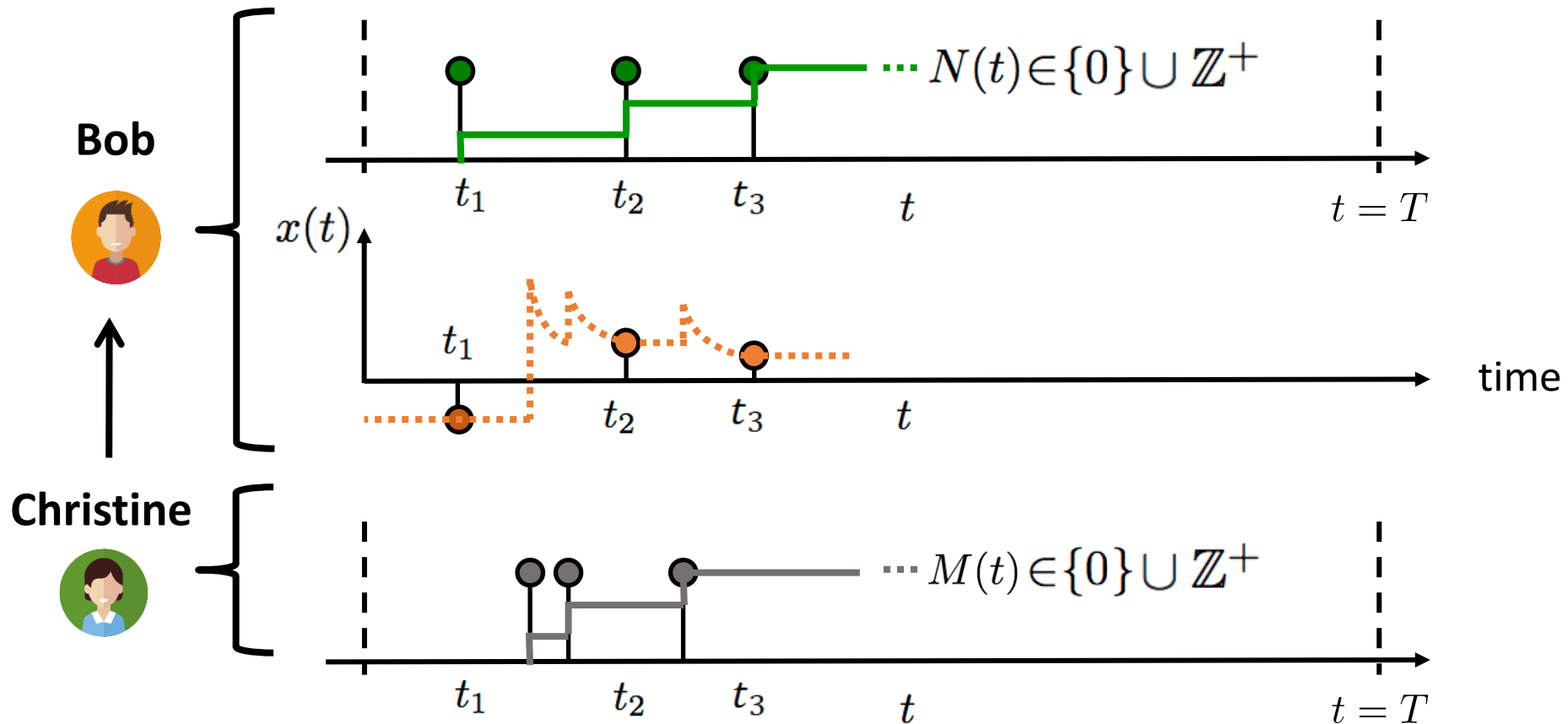
Distribution for the marks:

$$x^*(t_i) \sim p(x^* | x(t)) \Rightarrow dx(t) = \underbrace{f(x(t), t)dt}_{\text{Drift}} + \underbrace{h(x(t), t)dN(t)}_{\text{Event influence}}$$

Observations:

1. Marks dependent on the temporal dynamics
2. Distribution represents additional source of uncertainty

# Mutually exciting + marks

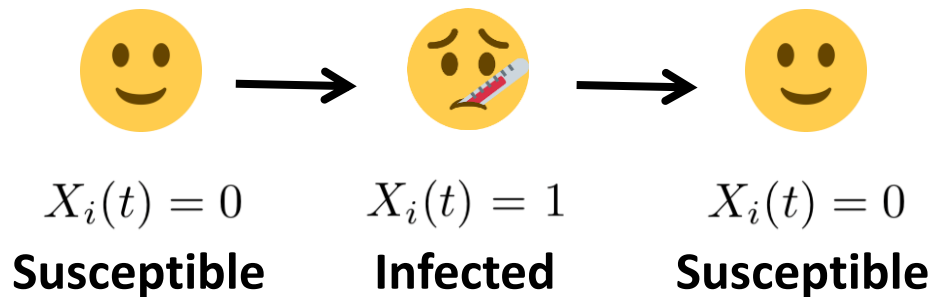


## Marks affected by neighbors

$$dx(t) = \underbrace{f(x(t), t)dt}_{\text{Drift}} + \underbrace{g(x(t), t)dM(t)}_{\text{Neighbor influence}}$$

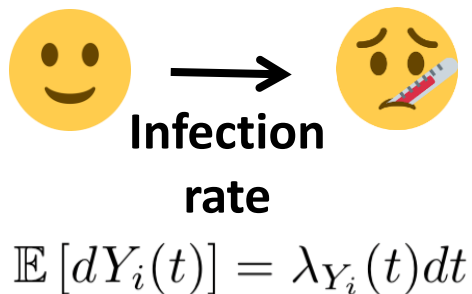
# Marked TPPs as stochastic dynamical systems

## Example: Susceptible-Infected-Susceptible (SIS)



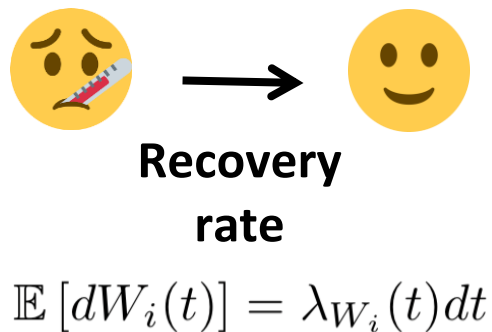
SDE with jumps

$$dX_i(t) = \underbrace{dY_i(t)}_{\text{It gets infected}} - \underbrace{dW_i(t)}_{\text{It recovers}}$$



Node is susceptible

$$\lambda_{Y_i}(t)dt = (1 - X_i(t))\beta \underbrace{\sum_{j \in \mathcal{N}(i)} X_j(t)}_{\text{If friends are infected, higher infection rate}} dt$$



SDE with jumps

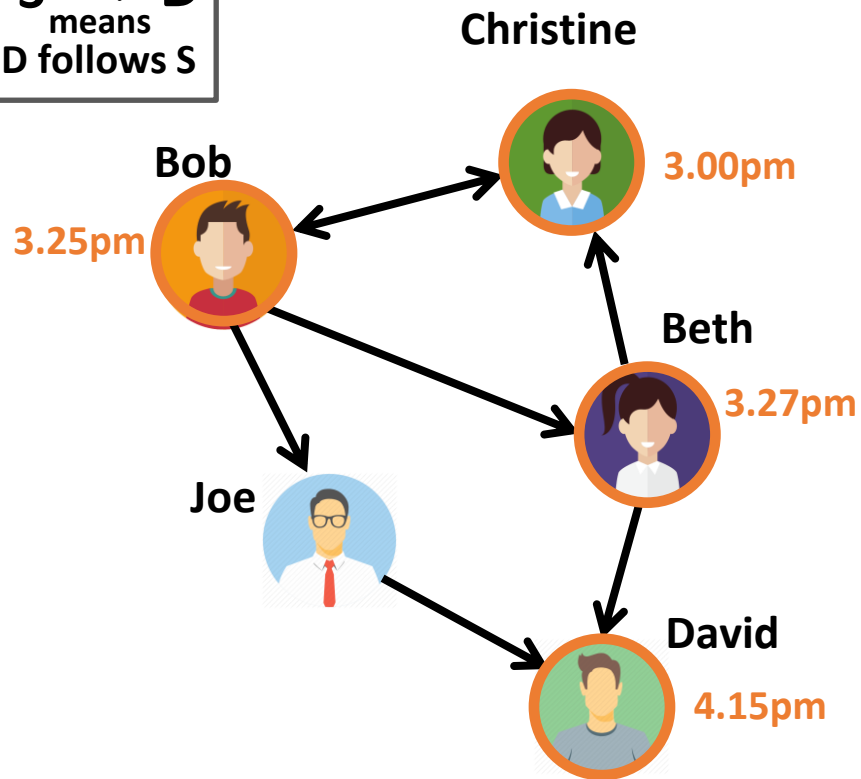
$$d\lambda_{W_i}(t) = \underbrace{\delta dY_i(t)}_{\text{Self-recovery rate when node gets infected}} - \underbrace{\lambda_{W_i}(t)dW_i(t)}_{\text{If node recovers, rate to zero}} + \underbrace{\rho dN_i(t)}_{\text{Rate increases if node gets treated}}$$

# Models & Inference

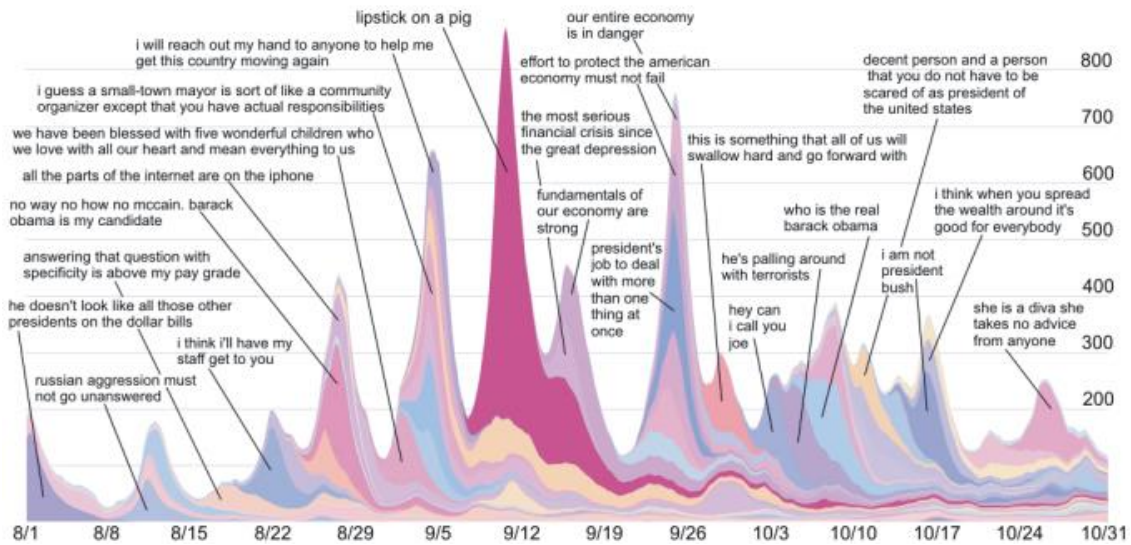
- 1. Modeling event sequences**
2. Clustering event sequences
3. Capturing complex dynamics
4. Causal reasoning on event sequences

# Event sequences as cascades

$S \rightarrow D$   
S means  
D follows S

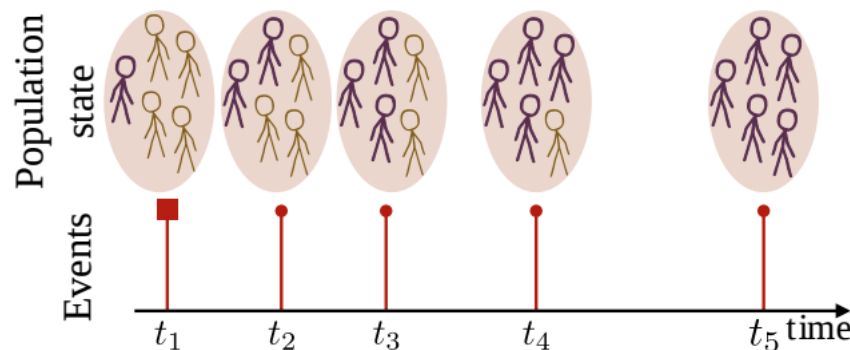
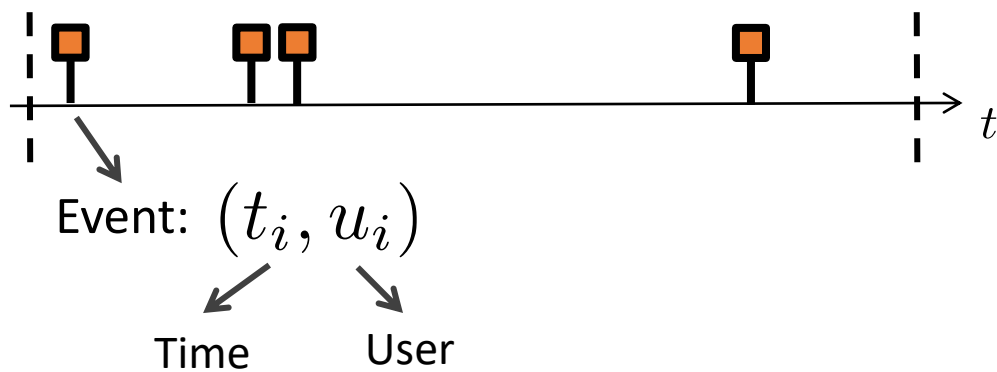


## Information Diffusion



[Leskovec et al., 2009]

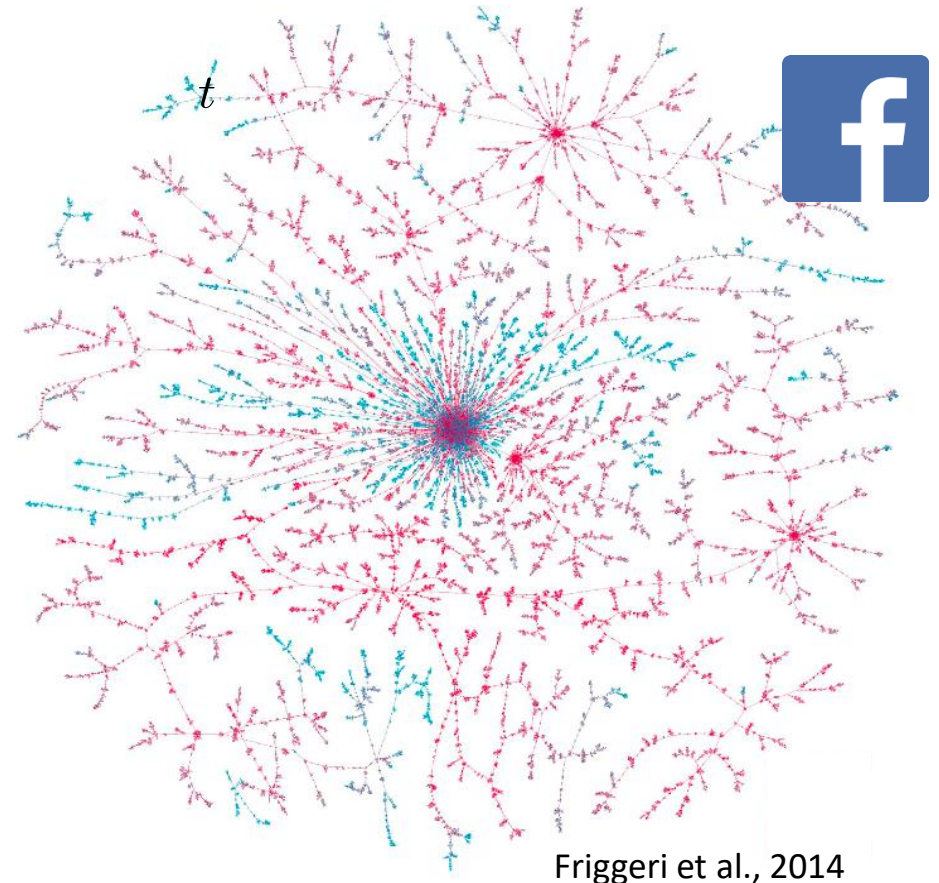
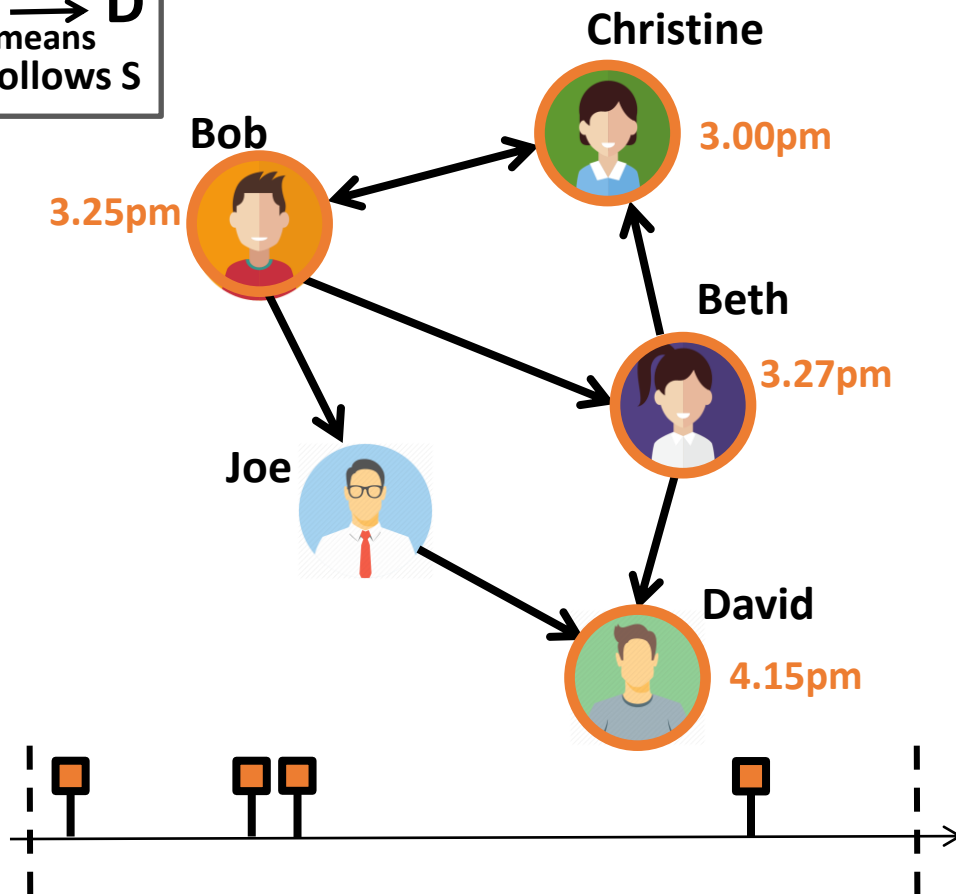
## Disease Diffusion



[Rizoiu et al., 2018]

# An example: idea adoption

**S** → **D**  
means  
D follows S



**They can have an impact  
in the off-line world**

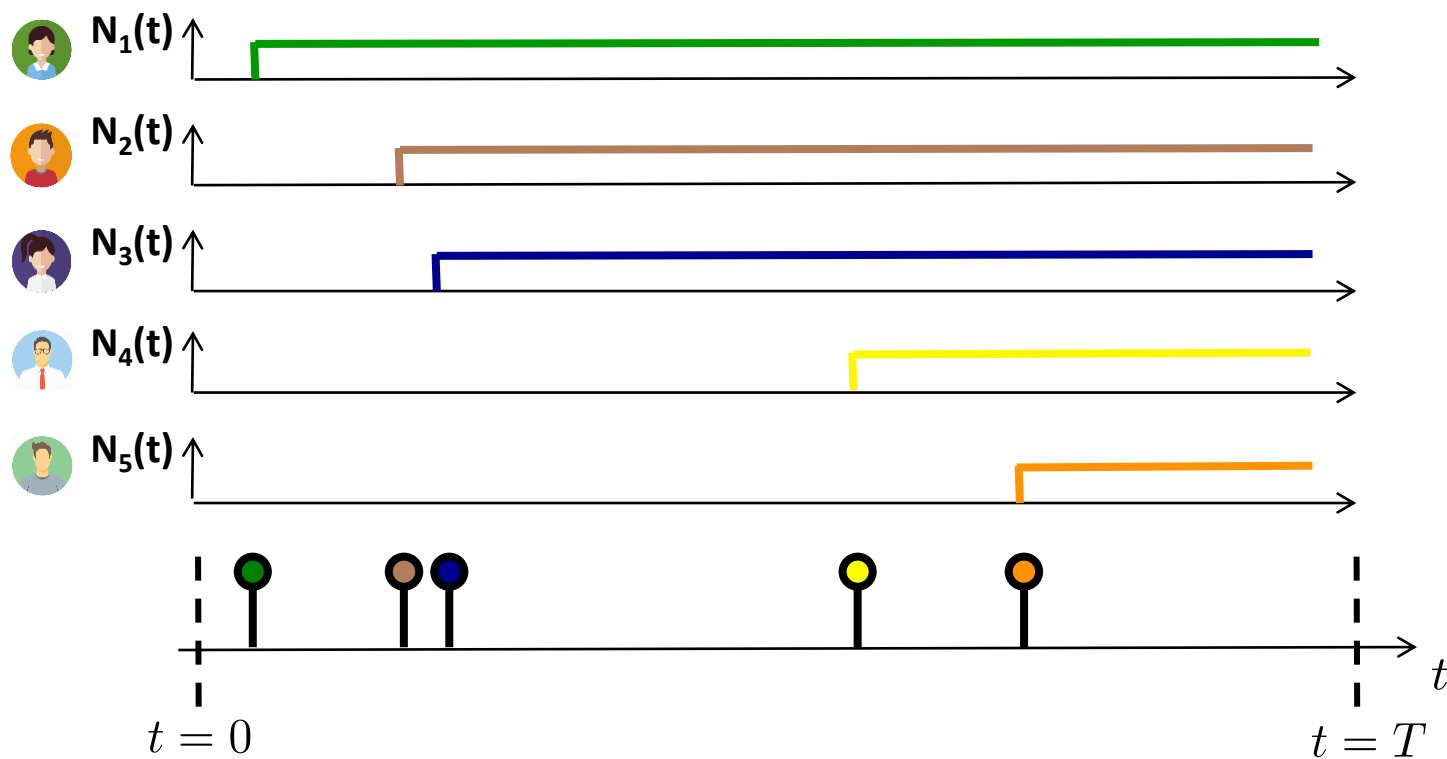
**theguardian**

Click and elect: how fake news helped Donald Trump win a real election

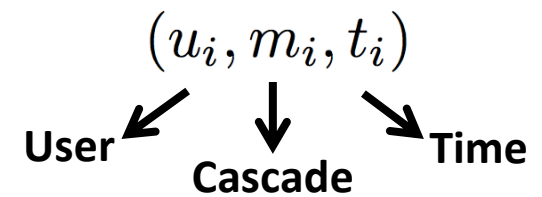


# Infection cascade representation

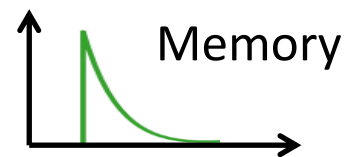
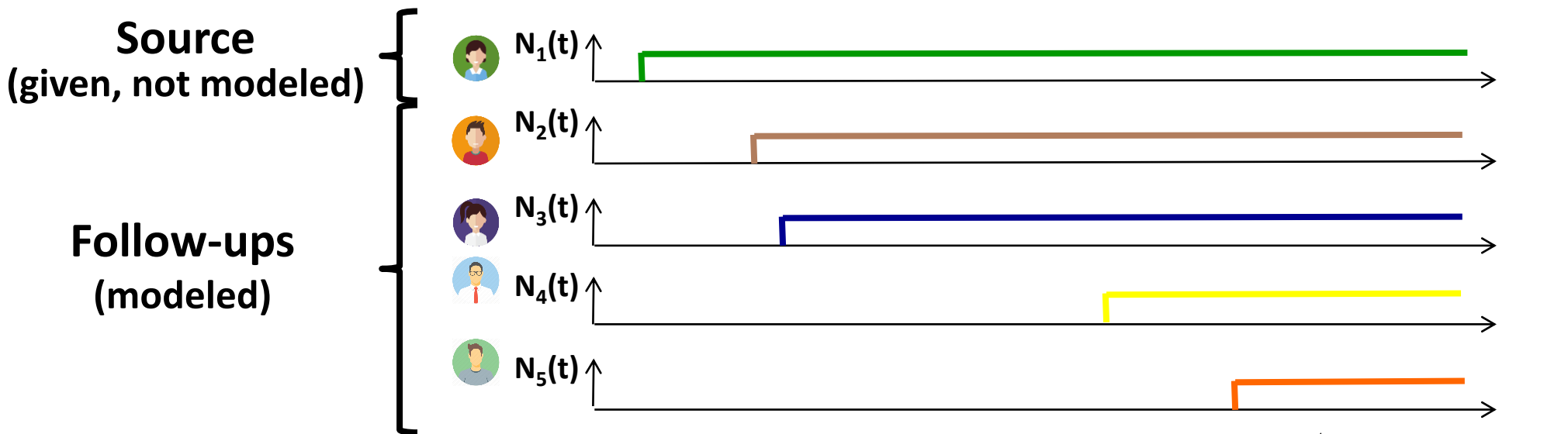
We represent an infection cascade using **terminating temporal point processes**:



**Infection event:**



# Infection intensity

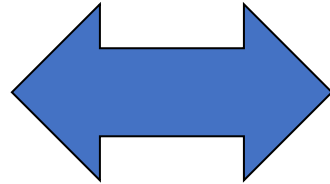


$$\lambda_u^*(t) = \underbrace{(1 - N_u(t))}_{\text{Users get infected only once}} \sum_{v \in [m]} \underbrace{b_{vu} \sum_{e_i \in \mathcal{H}_v(t)} \kappa(t - t_i)}_{\text{Influence from user } v \text{ on user } u \text{ Previous infections of user } v}$$

# Model inference from multiple cascades

**Conditional intensities**

$$\lambda_u^*(t)$$



**Diffusion log-likelihood**

$$\mathcal{L} = \sum_{u=1}^n \log \lambda_u^*(t_u) - \int_0^T \lambda_u^*(\tau) d\tau$$

**Maximum likelihood approach to find model parameters!**



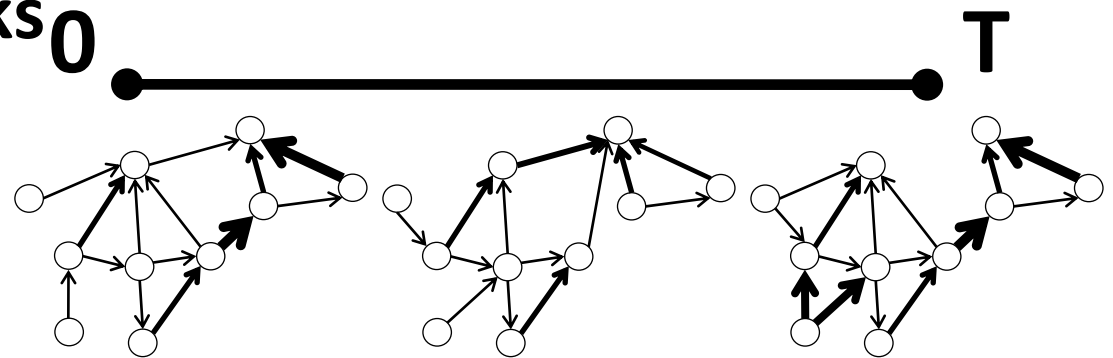
**Sum up log-likelihoods of multiple cascades!**

**Theorem.** For any choice of parametric memory, the **maximum likelihood** problem is **convex**.

In some cases, influence change over time:



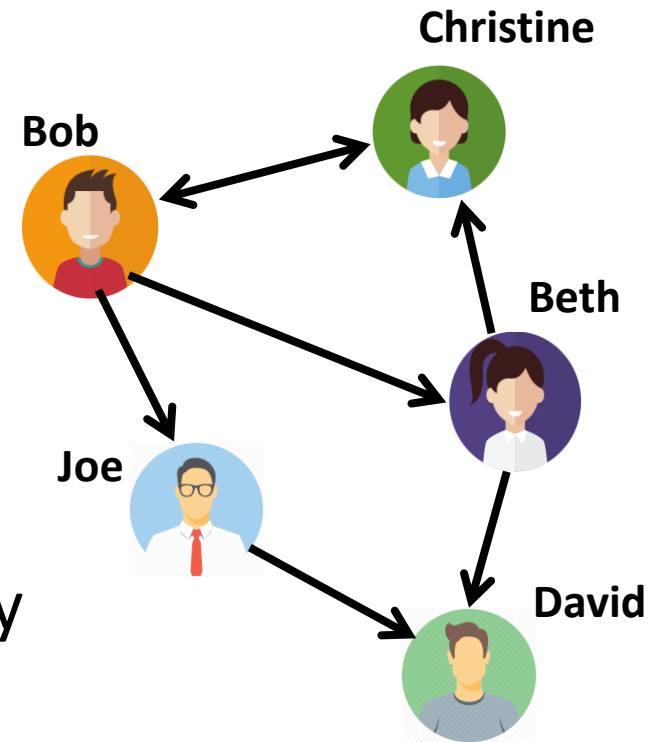
Propagation over networks  
with variable influence



# Recurrent events: beyond cascades

**Up to this point**, each users is only infected once, and event sequences can be seen as cascades.

**In general, users perform recurrent events over time.** E.g., people repeatedly express their opinion online:



How social media is revolutionizing debates

*The New York Times*

*Social Media Are Giving a Voice to Taste Buds*



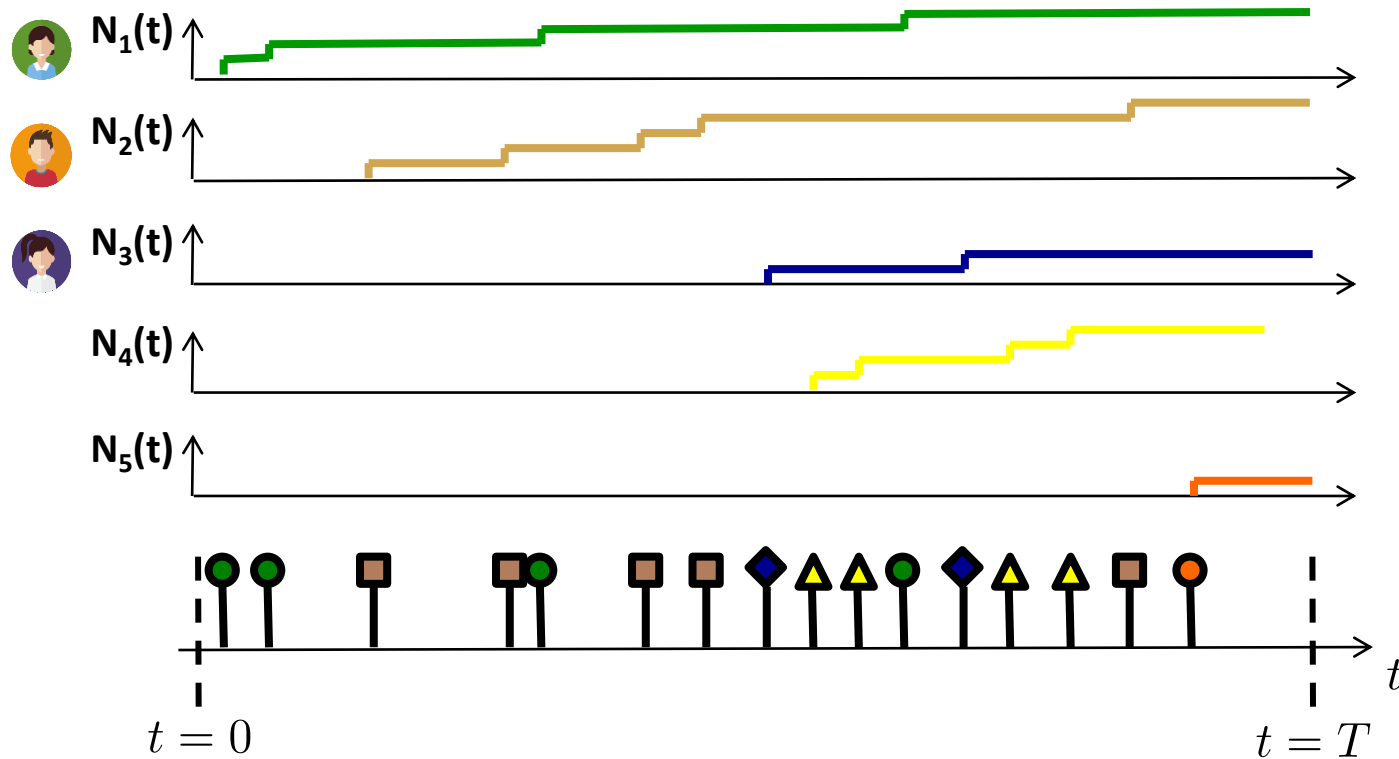
**Twitter Unveils A New Set Of Brand-Centric Analytics**

*The New York Times*

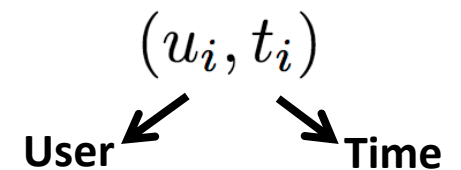
*Campaigns Use Social Media to Lure Younger Voters*

# Recurrent events representation

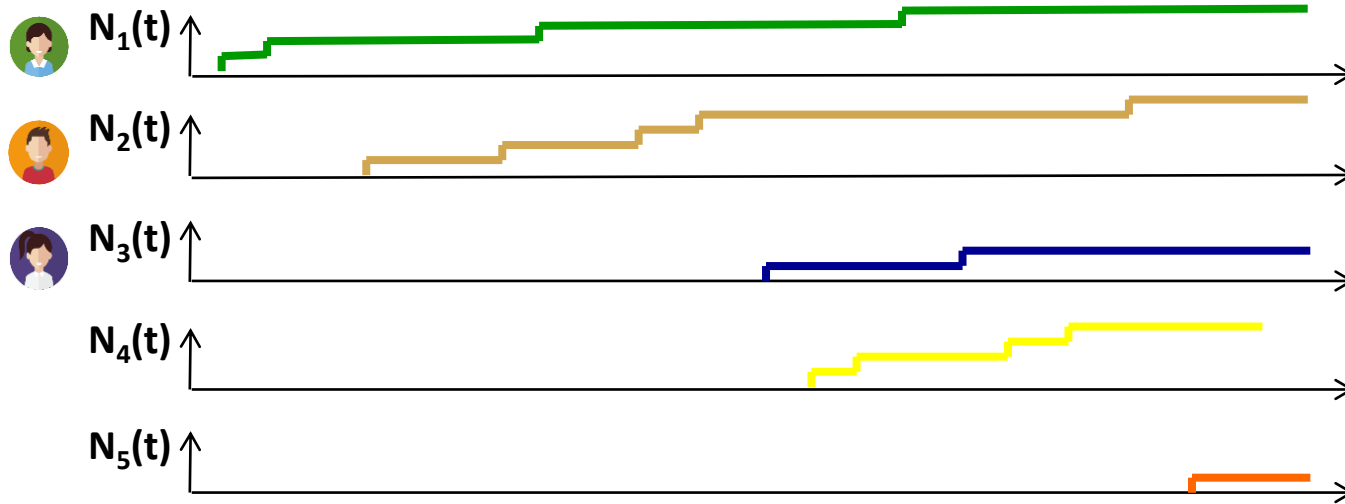
We represent messages using **nonterminating temporal point processes**:



**Recurrent event:**



# Recurrent events intensity



**Cascade sources!**



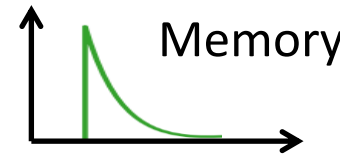
$\underbrace{\lambda_u^*(t)}_{\text{User's intensity}}$

$= \underbrace{\mu_u}_{\text{Events on her own initiative}} +$

$$\sum_{v \in [m]} b_{vu}$$

**Influence from user v on user u**

$$\underbrace{\sum_{e_i \in \mathcal{H}_v(t)} \kappa(t - t_i)}_{\text{Previous messages by user v}}$$



**Hawkes process**

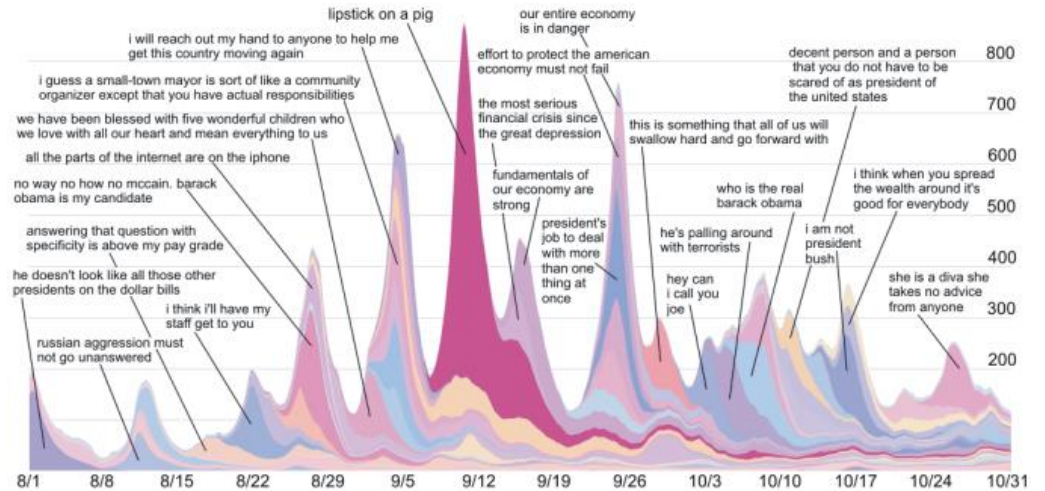
# Models & Inference

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



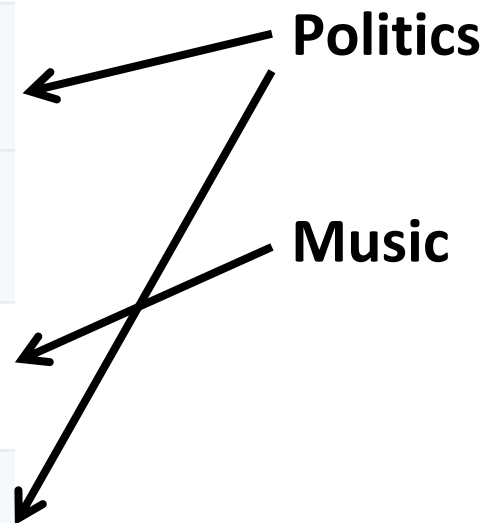
# Event sequences

So far, we have assumed the cascade (topic, meme, etc.) that each event belongs to was known.

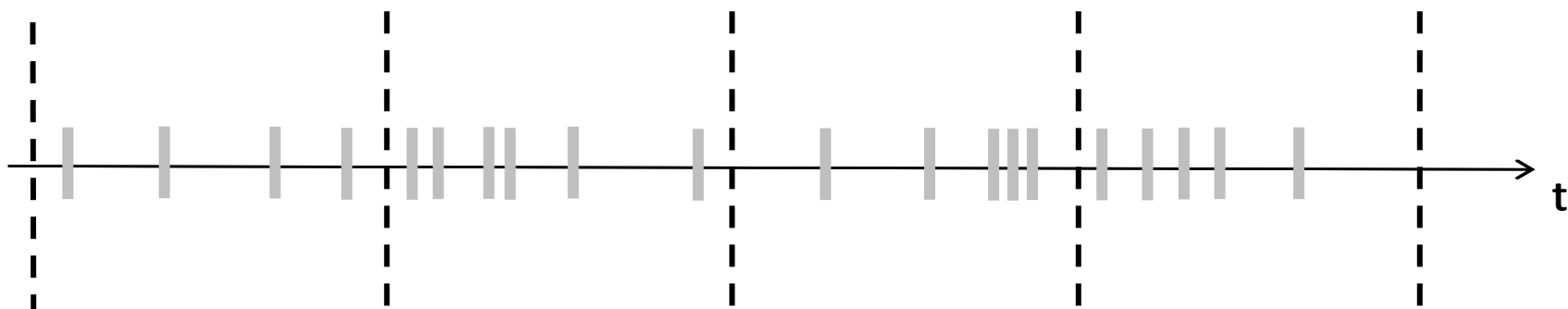


Often, the cluster (topic, meme, etc.) that each event in a sequence belongs to is not known:

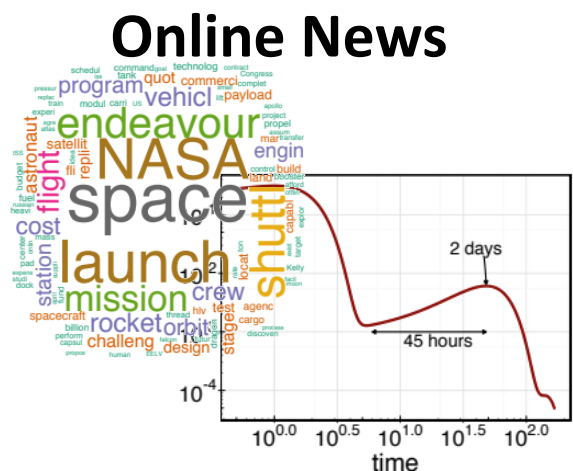
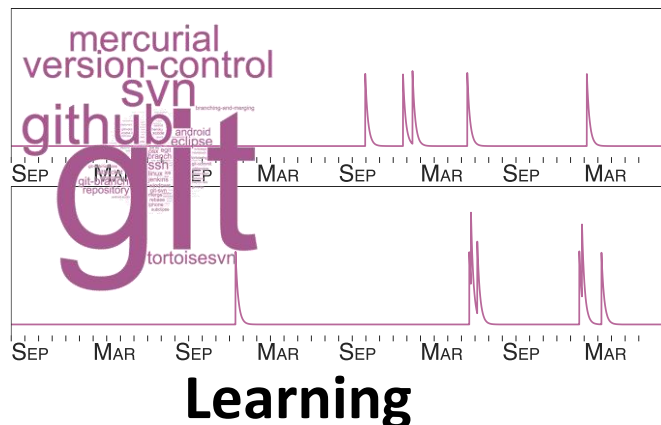
-  **BBC News (World)** @BBCWorld · 4m  
Turkey election: Erdogan win ushers in new presidential era
-  **BBC News (World)** @BBCWorld · 46m  
Dublin church: Seven injured as car hits pedestrians
-  **BBC News (World)** @BBCWorld · 2h  
Nigerian music star D'banj's son 'drowns at home'
-  **BBC News (World)** @BBCWorld · 2h  
Turkey election: Country's heart split over Erdogan victory



Assume the event **cluster** to be **hidden** and aim to automatically **learn the cluster assignments** from the data:



**Bayesian methods** to cluster event sequences in the context of:



**Health care**

Method	DMHP
ICU Patient	<b>0.3778</b>
IPTV User	<b>0.2004</b>

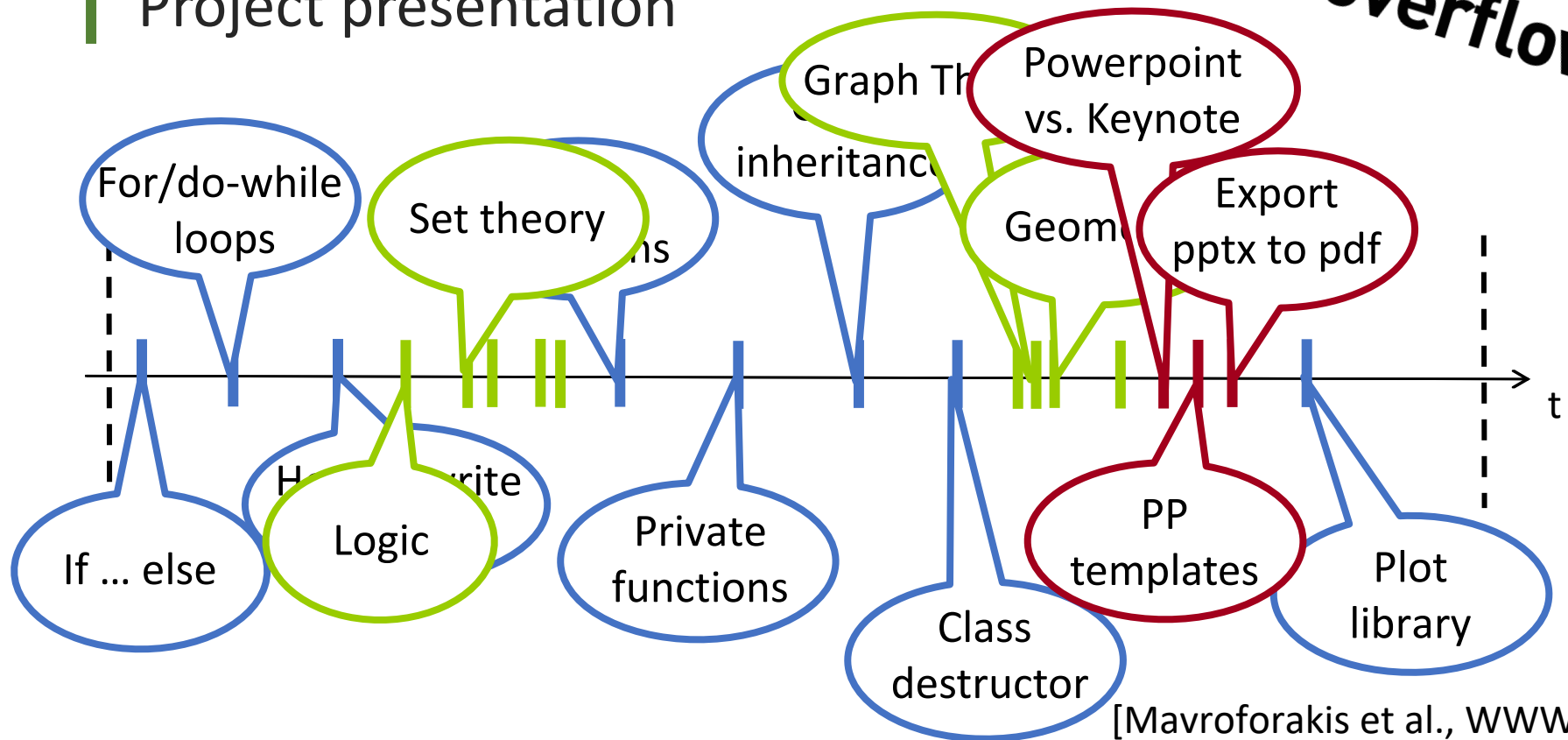
[Du et al., 2015; Mavroforakis et al., 2017; Xu & Zha, 2017]

# Hierarchical Dirichlet Hawkes process



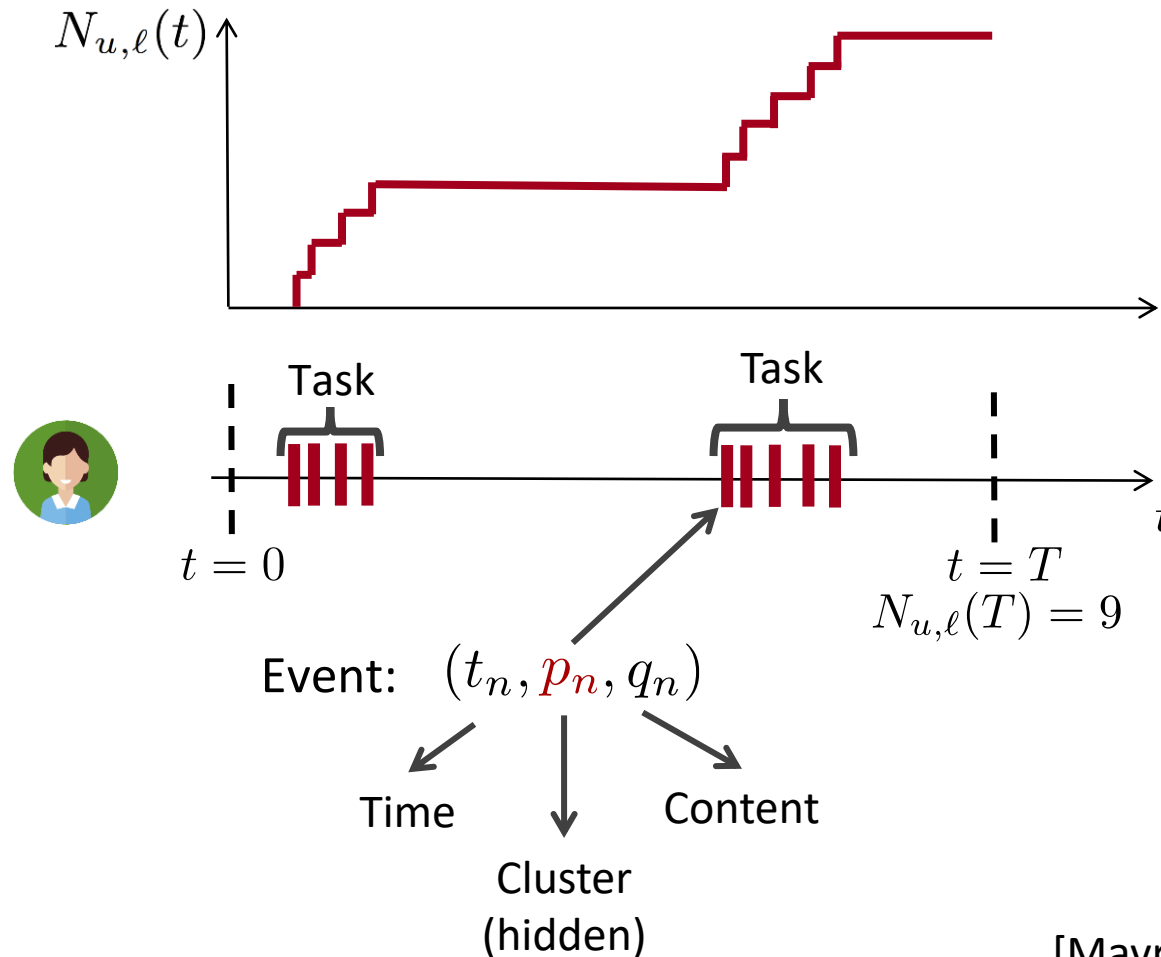
## 1st year computer science student

- Introduction to programming
- Discrete math
- Project presentation

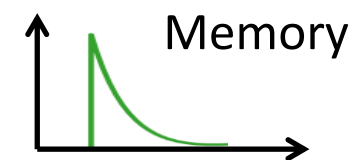
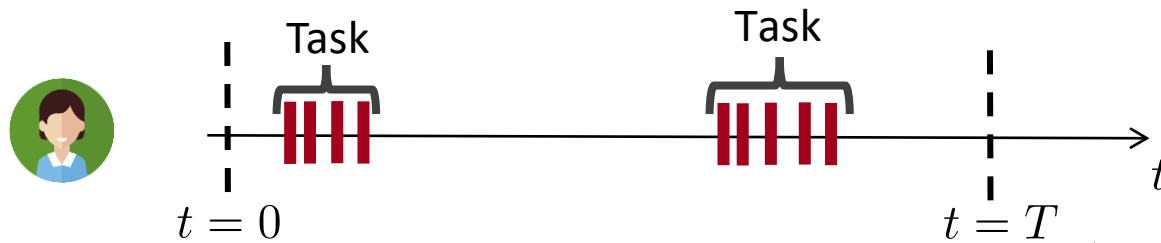
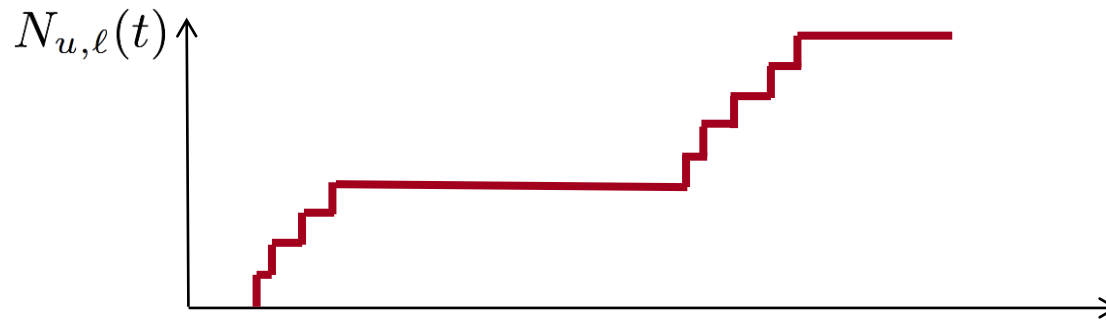


# Events representation

We represent the events using **marked temporal point processes**:



# Cluster intensity



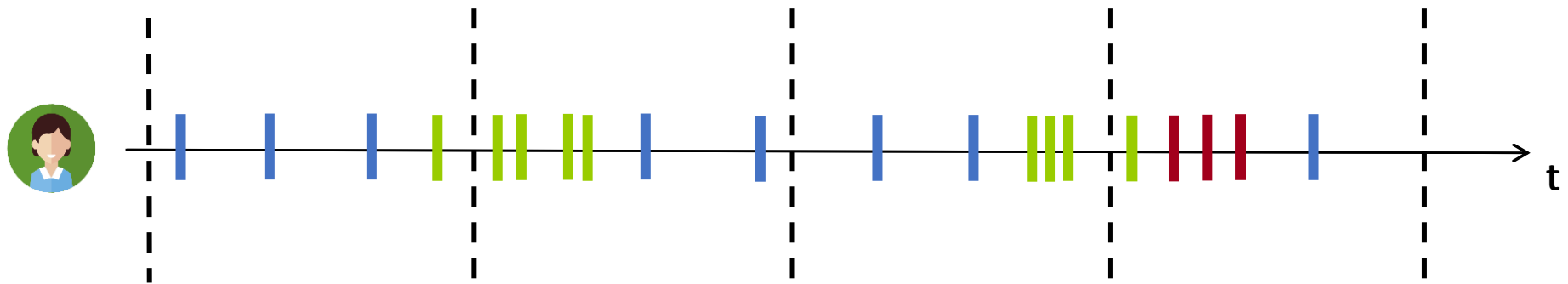
New cascade rate  
Cluster popularity

$$\underbrace{\lambda_{u,l}^*(t)}_{\substack{\text{Intensity} \\ \text{or rate} \\ \text{(events / hour)}}} = \underbrace{\mu_u \pi_l}_{\text{Own initiative}} + \underbrace{\sum_{j:t_j \in \mathcal{H}_{u,l}(t)} k_{\theta_l}(t - t_j)}_{\text{Follow-up}}$$

**Hawkes process**

# User events intensity

Users adopt more than one cluster:



A user's learning events as a multidimensional Hawkes:

$$\begin{array}{l} \text{Time} \\ \swarrow \\ (t_n, p_n) \end{array} \begin{array}{l} \text{cluster} \\ \swarrow \\ p_n \end{array} \sim \text{Hawkes} \begin{pmatrix} \lambda_{u,1}^*(t) \\ \vdots \\ \lambda_{u,\infty}^*(t) \end{pmatrix}$$

$$\text{Content} \rightarrow q_n = \boldsymbol{\omega} \quad \omega_j \sim \text{Multinomial}(\boldsymbol{\theta}_p)$$

# People share same clusters

*Different users adopt same clusters*



Efficient model inference using  
Sequential Montecarlo!

Clus

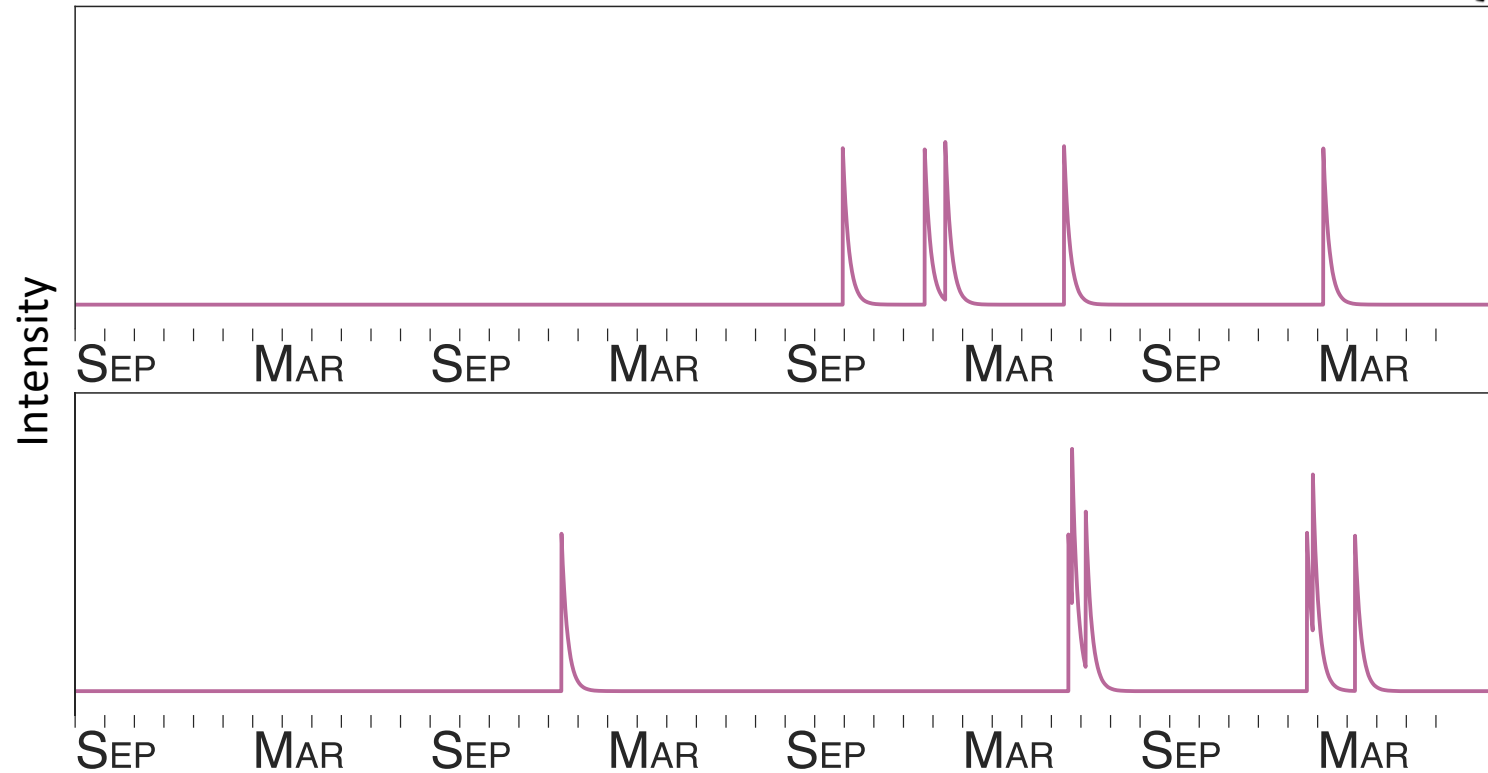
- Shared parameters across users.

*Details in the  
reference below!*

# Content



# Intensities



**Version control tasks tend to be specific,  
quickly solved after performing few questions**



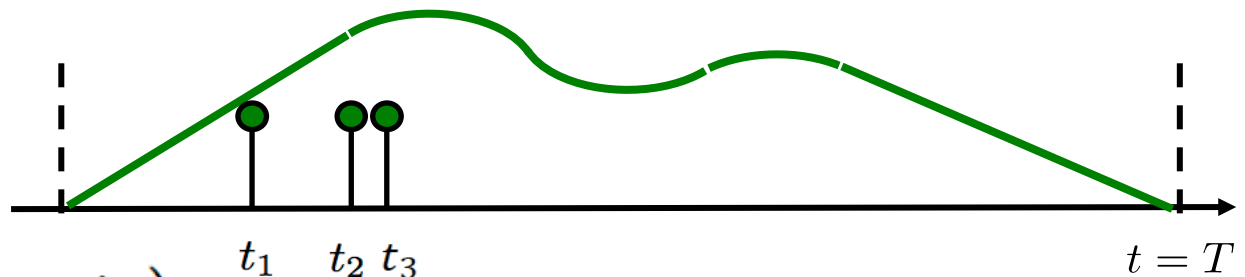


# Models & Inference

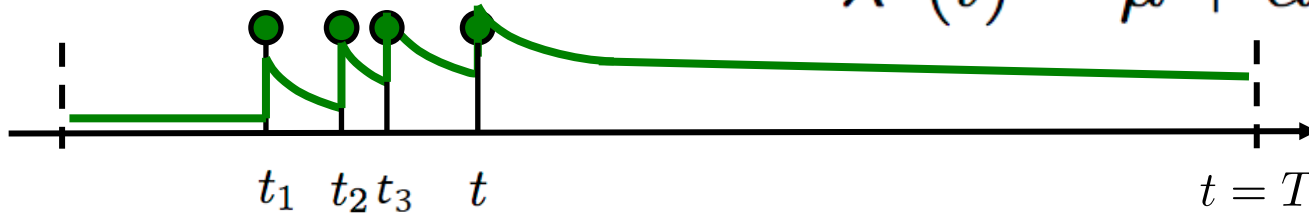
1. Modeling event sequences
2. Clustering event sequences
- 3. Capturing complex dynamics**
4. Causal reasoning on event sequences

Up to now, we have focused on simple temporal dynamics (and intensity functions):

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



$$\lambda^*(t) = \sum_j \alpha_j k(t - t_j)$$



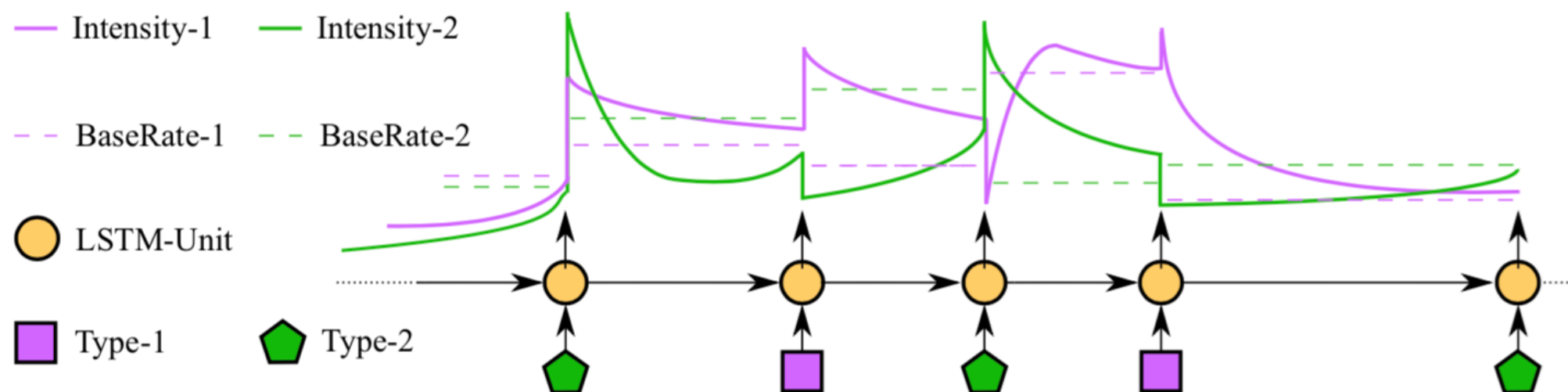
$$\lambda^*(t) = \mu + \alpha \sum_{t_i \in \mathcal{H}(t)} \kappa_\omega(t - t_i)$$

Recent works make use of **RNNs** to capture more complex dynamics

[Du et al., 2016; Dai et al., 2016; Mei & Eisner, 2017; Jing & Smola, 2017; Trivedi et al., 2017; Xiao et al., 2017a; 2018]

# Neural Hawkes process

- 1) History effect does not need to be additive
- 2) Allows for complex memory effects (such as delays)



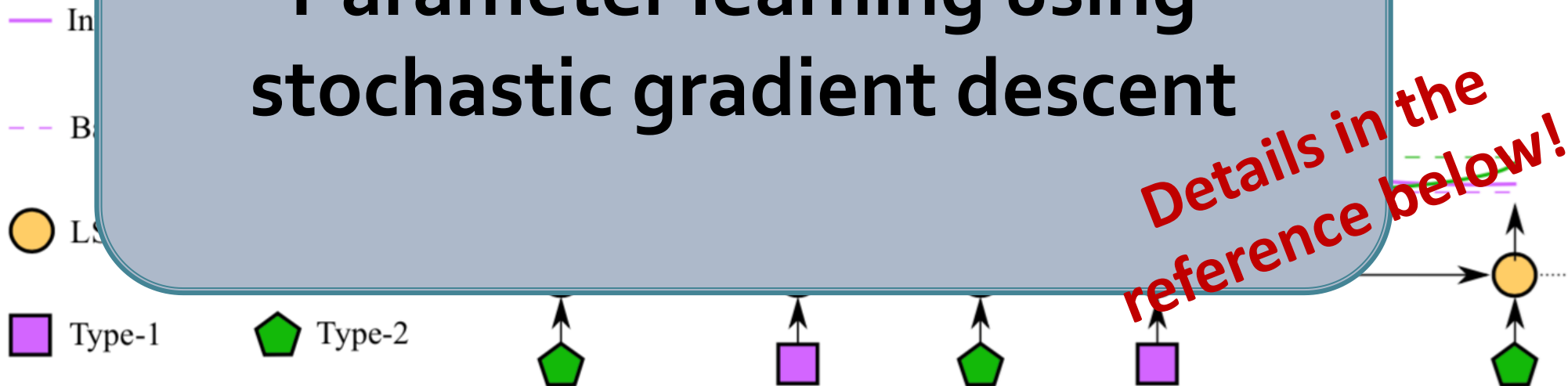
# Neural Hawkes process

$$\lambda_u(t) = f_u(\mathbf{w}_u^\top \mathbf{h}(t))$$

$$\mathbf{h}(t) = \text{DNN}(\mathcal{U}(t))$$

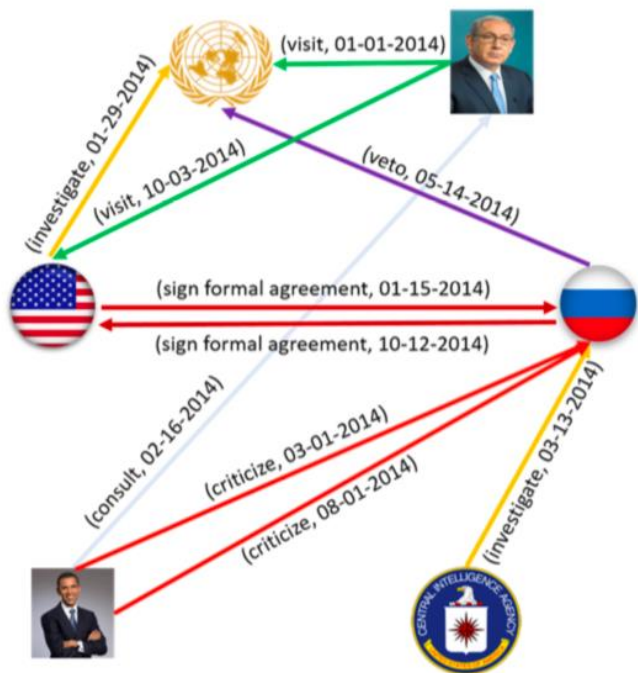
Memory

Parameter learning using  
stochastic gradient descent

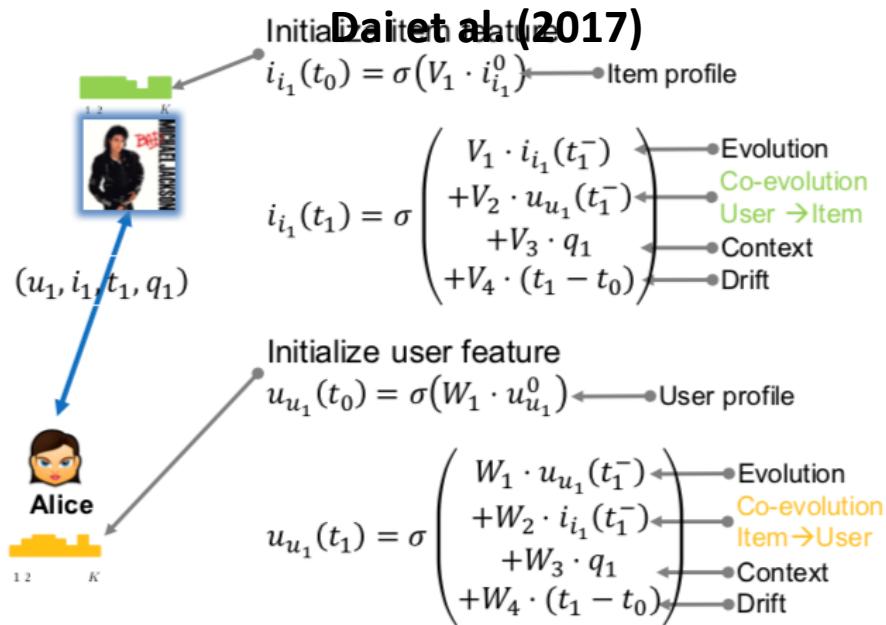


# Applications (I): Predictive Models

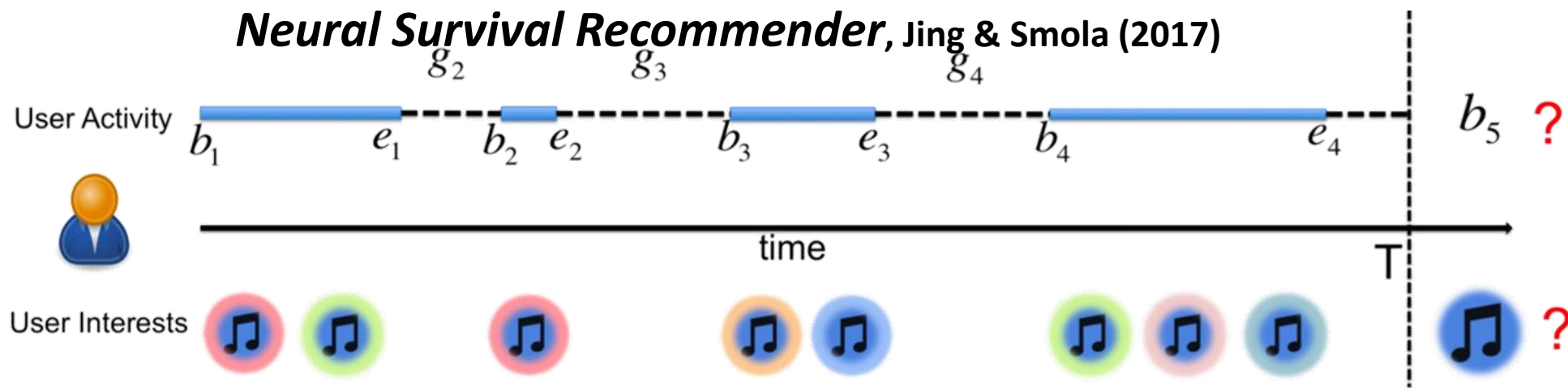
## Know-Evolve, Trivedi et al. (2017)



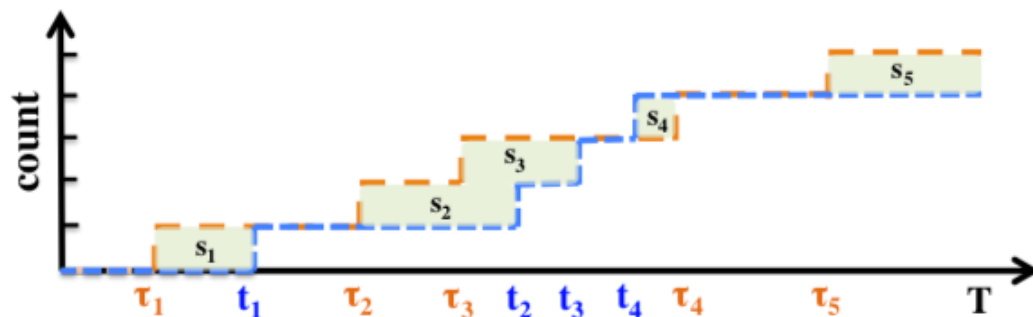
## Coevolutionary Embedding, Dai et al. (2017)



## Neural Survival Recommender, Jing & Smola (2017)

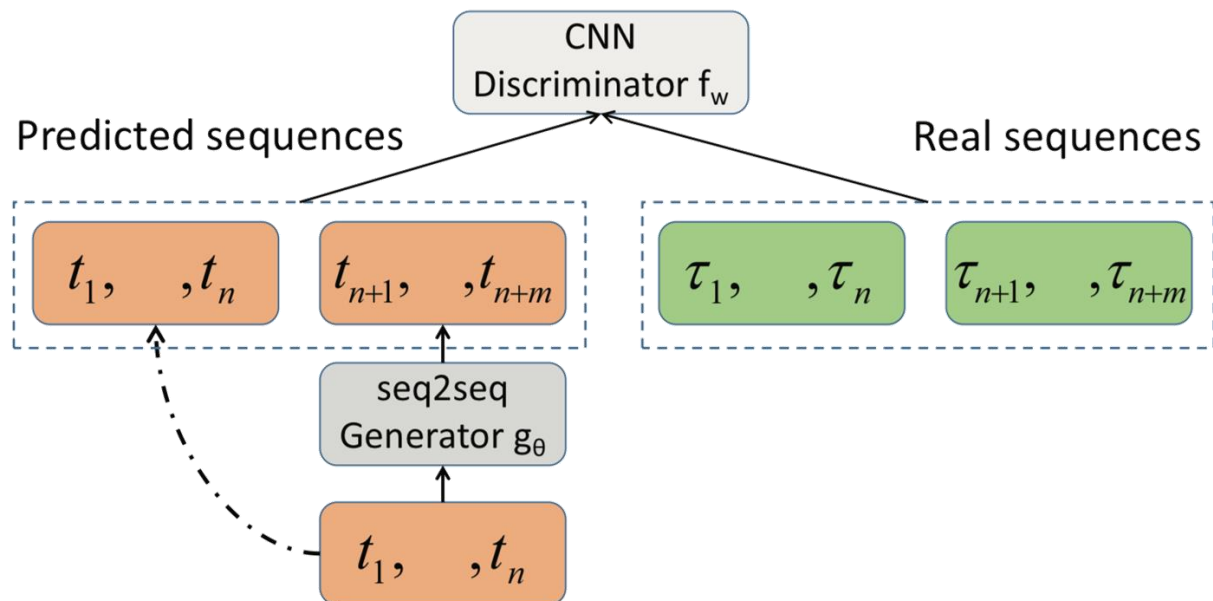


# Key idea: Intensity- and likelihood-free models



## Wasserstein-Distance for Temporal Point Processes

GAN architecture



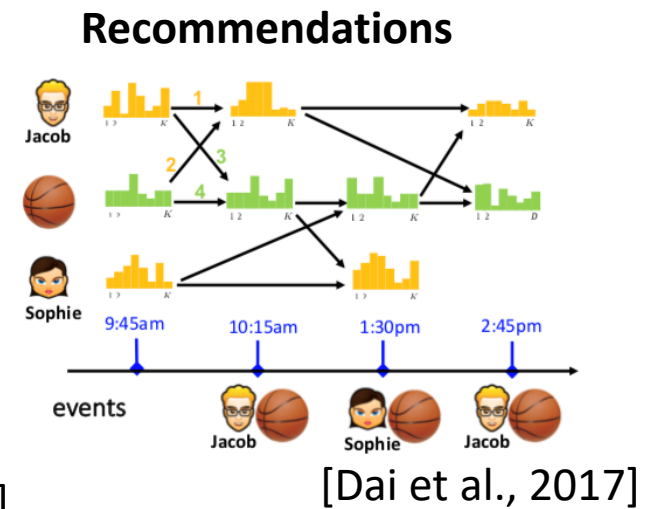
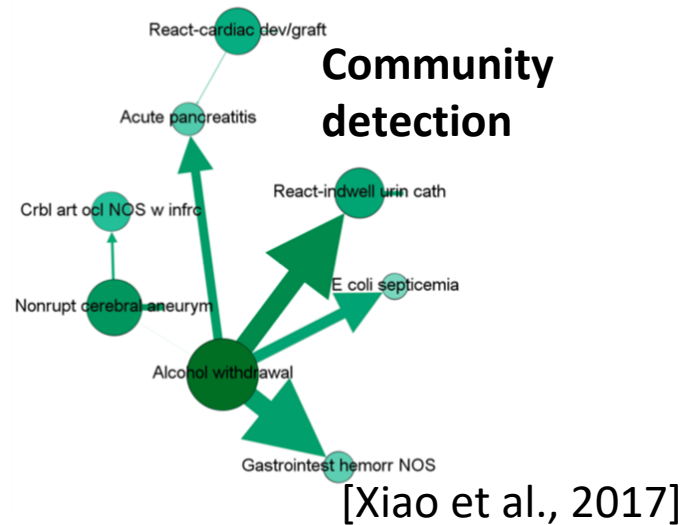
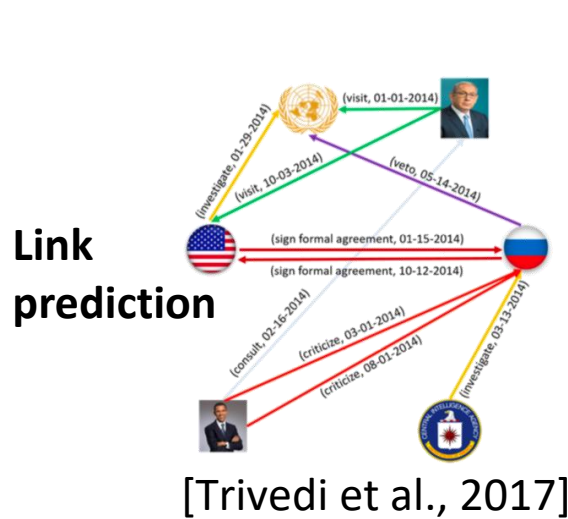
# Models & Inference

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

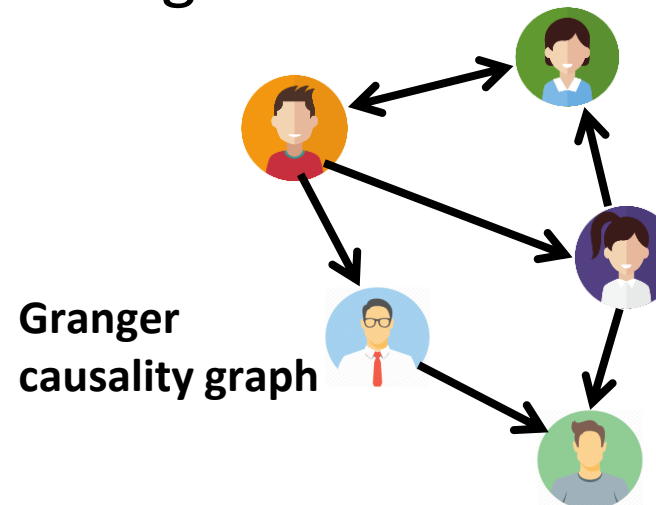
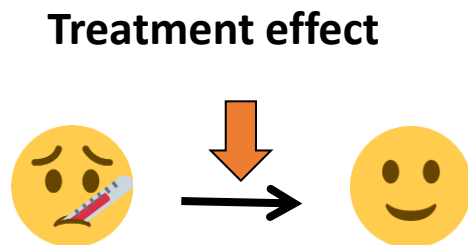


# Temporal point processes beyond prediction

So far, we have focused on models that improve predictions:



Recent works have focused on performing **causal inference** using event sequences:



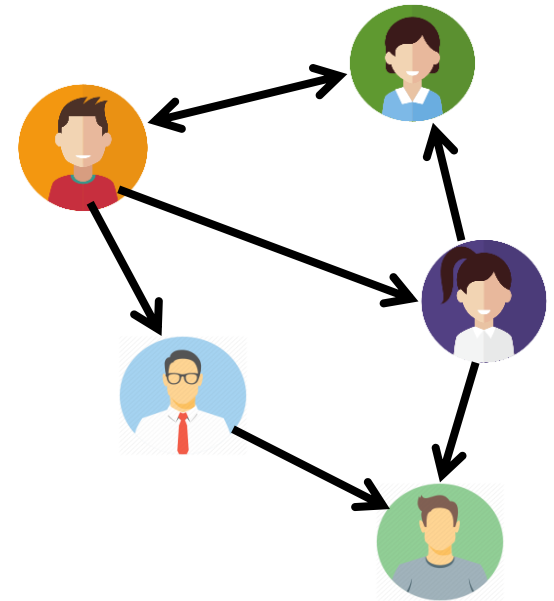
[Xu et al., 2016; Achab et al., 2017; Kuśmierczyk & Gomez-Rodriguez, 2018]

# Uncovering Causality from Hawkes Processes

## Multivariate Hawkes process:

$$N(t) = \sum_{u \in \mathcal{U}} N_u(t)$$

$$\lambda_u(t) = \mu_u + \sum_{v \in \mathcal{U}} \underbrace{\int_0^t k_{u,v}(t-t') dN_v(t')}_{\text{Effect of v's past events on u}}$$



## Granger causality:

"X causes Y in the sense of Granger causality if forecasting future values of Y is more successful while taking X past values into account"

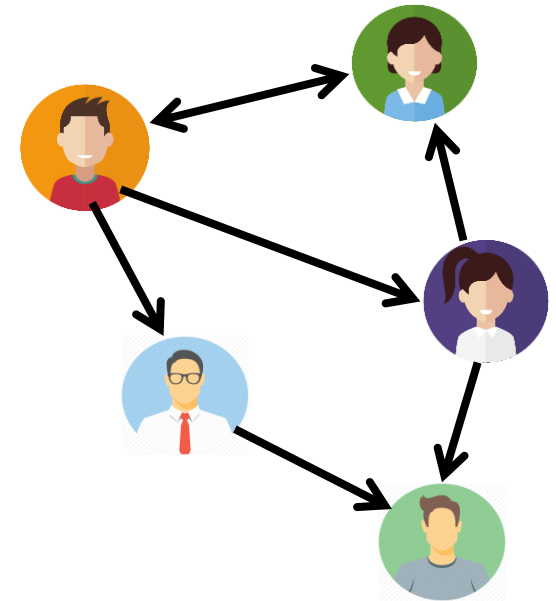
[Granger, 1969]

# Uncovering Causality from Hawkes Processes

## Multivariate Hawkes process:

$$N(t) = \sum_{u \in \mathcal{U}} N_u(t)$$

$$\lambda_u(t) = \mu_u + \sum_{v \in \mathcal{U}} \underbrace{\int_0^t k_{u,v}(t-t') dN_v(t')}_{\text{Effect of v's past events on u}}$$



## Granger causality on multivariate Hawkes processes:

“  $N_v(t)$  does not Granger-cause  $N_u(t)$  w.r.t.  $N(t)$  if and only if  $k_{u,v}(\tau) = 0$  for  $\tau \in \mathbb{R}^+$  ”

[Eichler et al., 2016]

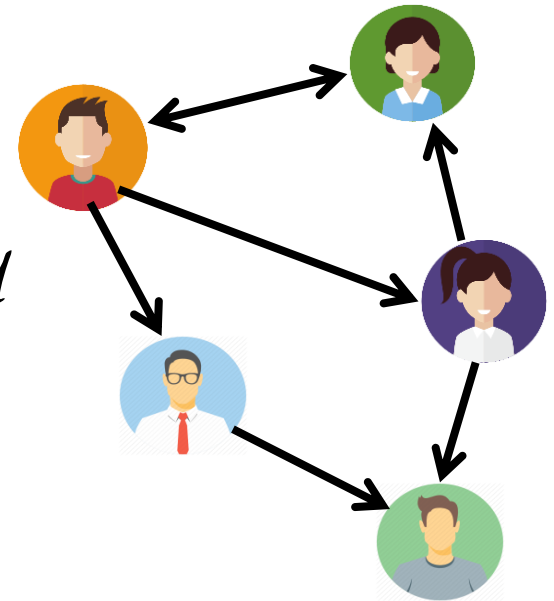
[Achab et al., ICML 2017]

# Uncovering Causality from Hawkes Processes

Goal is to estimate  $G = [g_{uv}]$ , where:

$$g_{uv} = \int_0^{+\infty} k_{u,v}(\tau) d\tau \geq 0 \text{ for all } u, v \in \mathcal{U}$$

Average total # of events of node  $u$  whose *direct* ancestor is an event by node  $v$



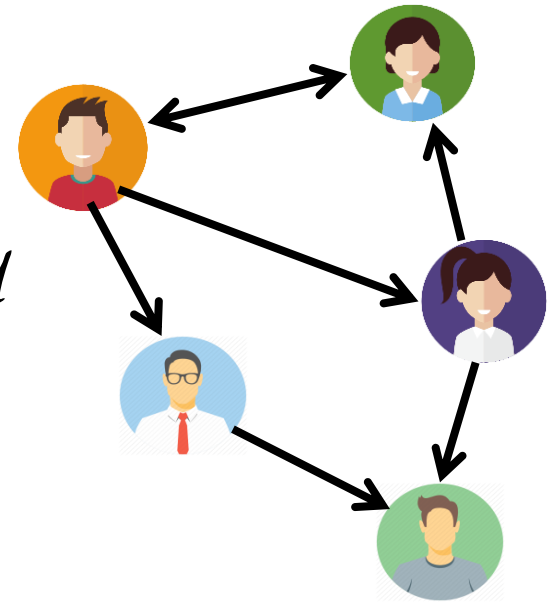
Then,  $G = [g_{uv}]$  quantifies the *direct causal relationship* between nodes.

# Uncovering Causality from Hawkes Processes

Goal is to estimate  $G = [g_{uv}]$ , where:

$$g_{uv} = \int_0^{+\infty} k_{u,v}(\tau) d\tau \geq 0 \text{ for all } u, v \in \mathcal{U}$$

Average total # of events of node  $u$  whose *direct* ancestor is an event by node  $v$



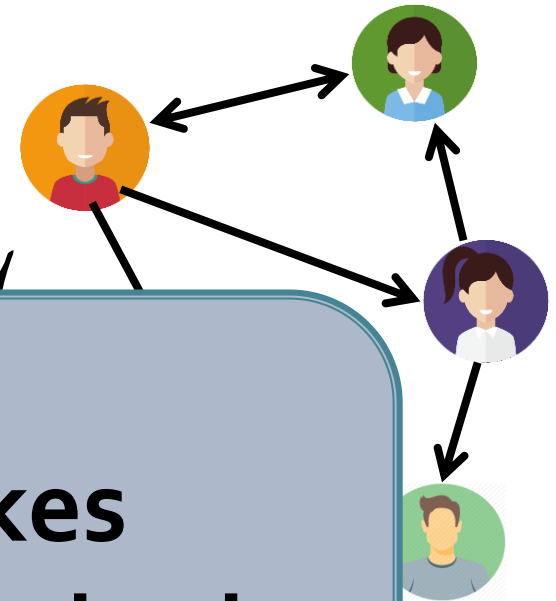
Then,  $G = [g_{uv}]$  quantifies the *direct causal relationship* between nodes.

**Key idea:** Estimate  $G$  using the cumulants  $dN(t)$  of the Hawkes process.

# Uncovering Causality from Hawkes Processes

Goal is to estimate  $G = [g_{uv}]$ , where:

$$g_{uv} = \int_0^{+\infty} k_{uv}(\tau) d\tau > 0 \text{ for all } u, v \in \mathcal{U}$$



**Non parametric Hawkes  
cumulant estimation method**

(with TensorFlow implementation)

**Details in the *hip*  
reference below!**

**Key idea:** Estimate  $G$  using the cumulants the  $dN(t)$  of the Hawkes process.

**Next Week:**

**Gaussian Process**

**Have a good day!**