Learning from event sequences

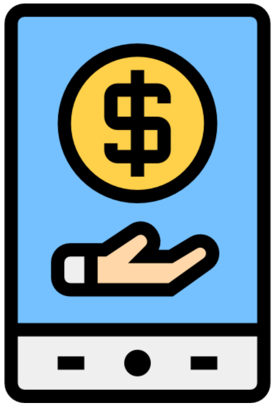
Alexey Zaytsev
Assistant professor, Skoltech

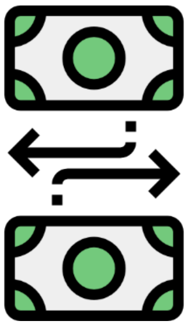
Uses material from ICML Tutorial, July 2018 by I. Valera & M.G. Rodriguez and slides by Sber AI lab

Temporal Point Processes (TPPs): applied problems

Skoltech

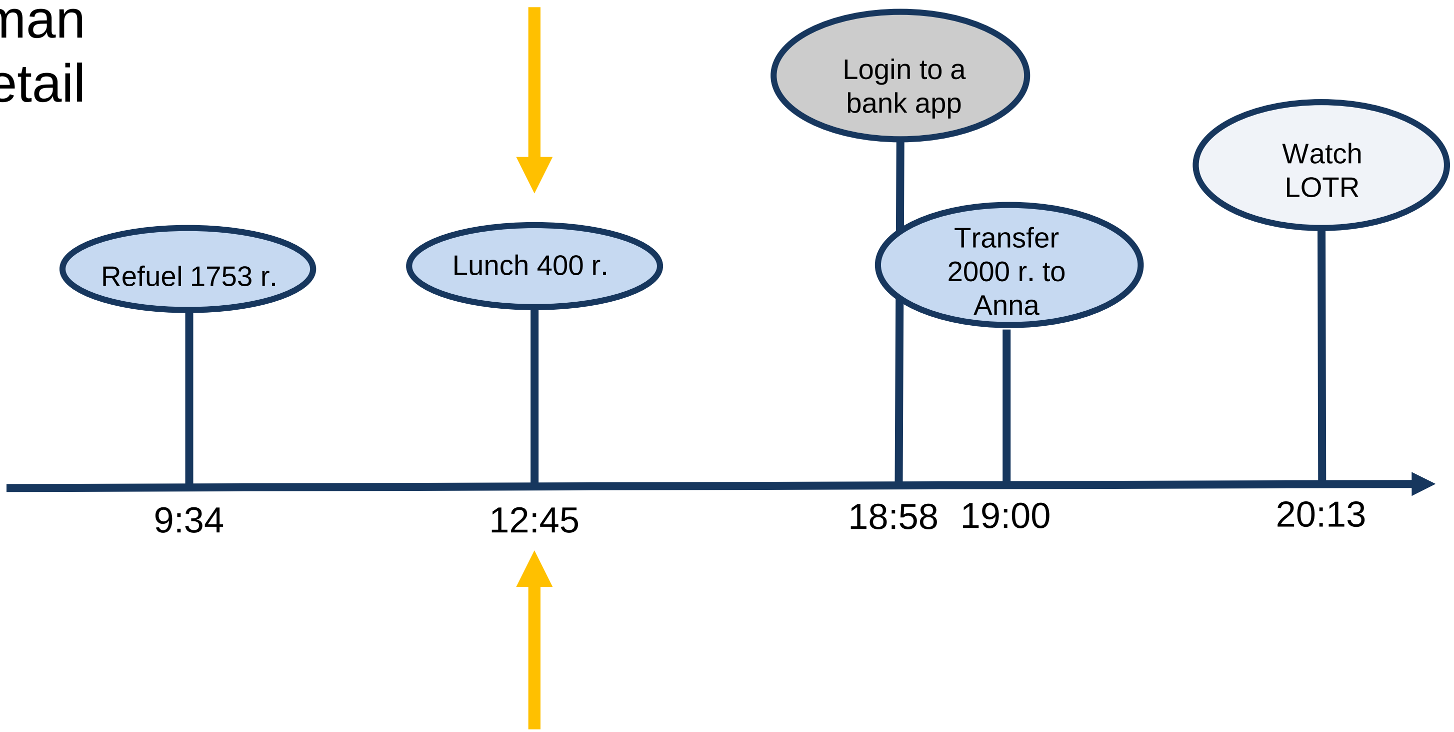
Example: financial transactions are event sequences





Alex, 33, man
works in retail

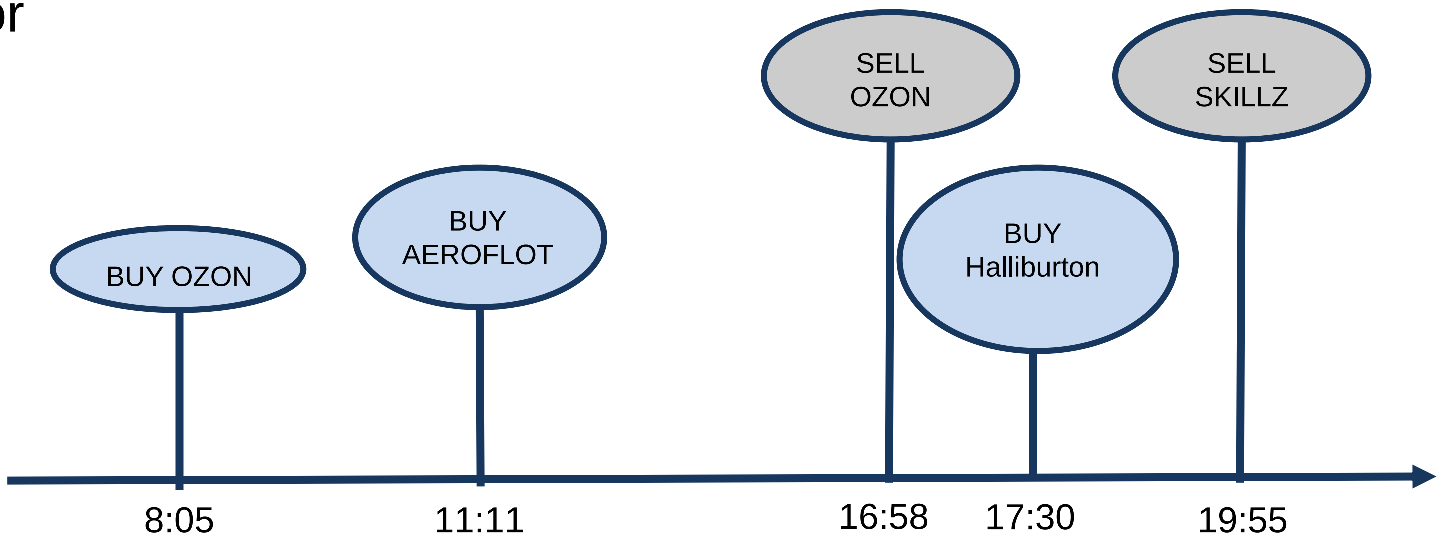
Event features:
sum, geolocation



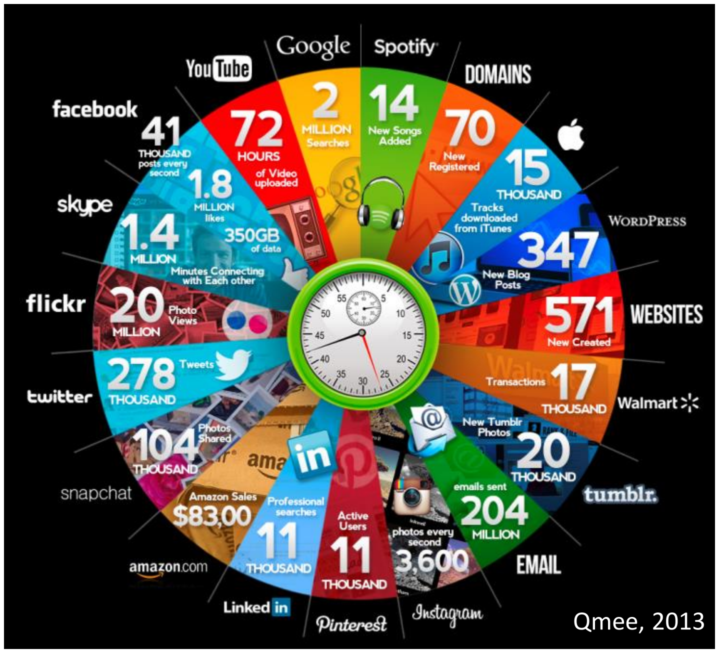
Event type + event time

Example: operations in markets are event sequences

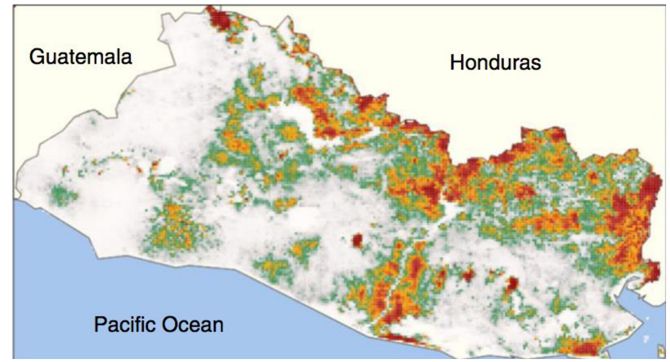
An investor



Many discrete *events* in continuous time



Online actions



Disease dynamics

Financial trading

Financial transactions

Skoltech

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

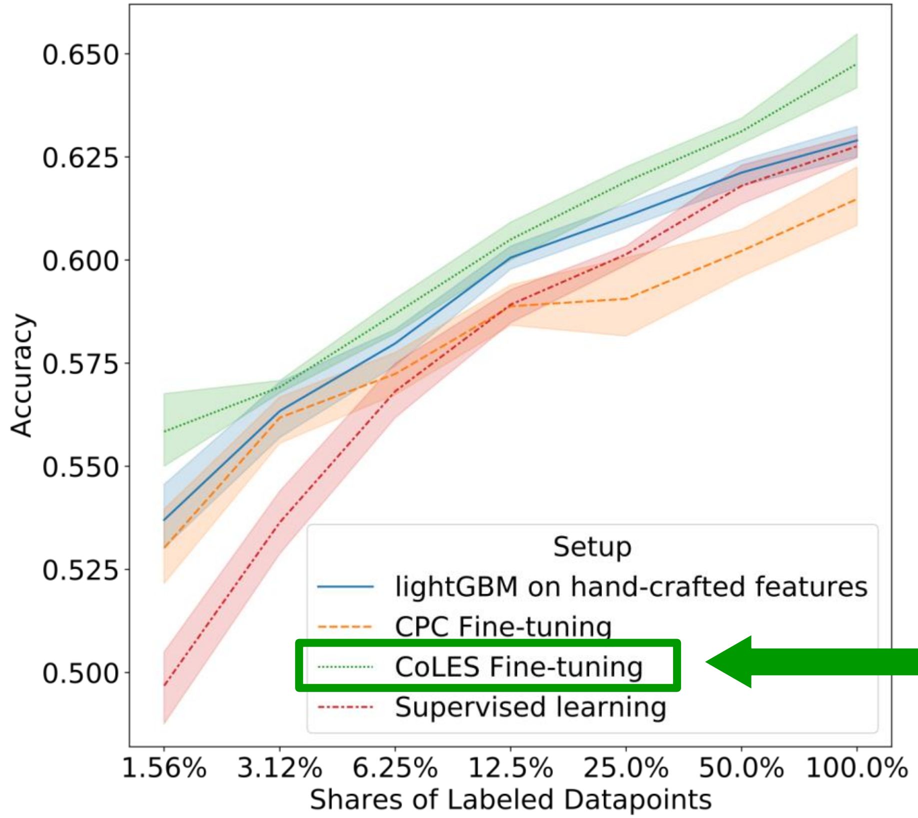
FAST → SLOW

…in a wide range of temporal scales.

Embeddings idea

Financial transactions uncover fundamental information about a customer.

Learn universal embeddings from event sequences that can be used to solve various task in bank.



Embeddings by Sber

Applications of event sequences in Sber

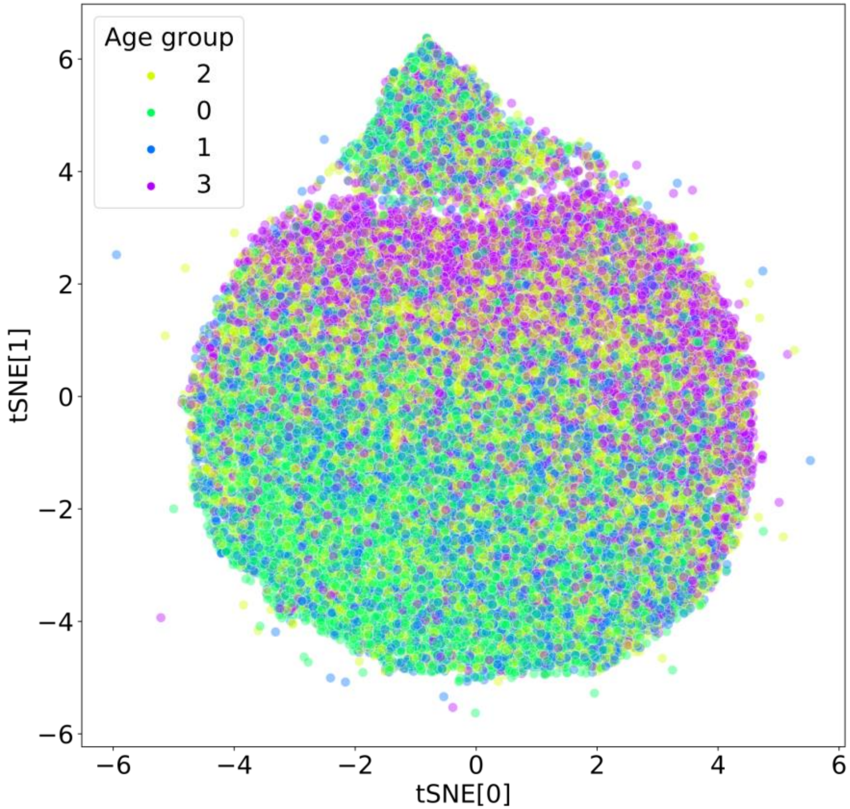
Embeddings are used in more than 50 different models in the bank

The reason: the quality is superior

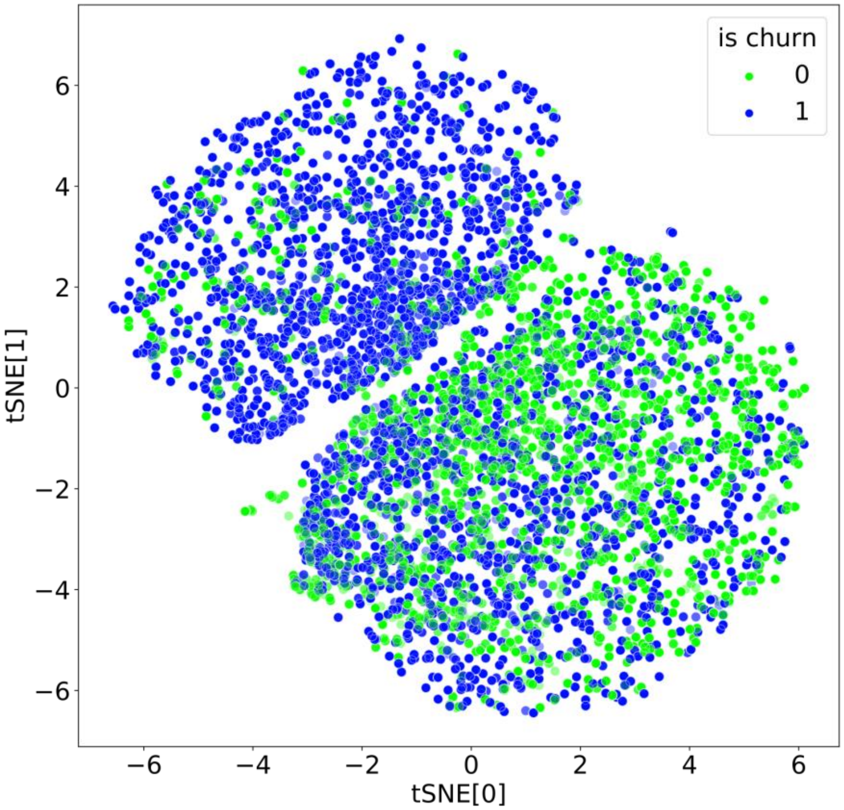
Company money transfer history embeddings		Card transaction history embeddings	
Credit product look-alike	+6 Gini	**Depositors churn**	+5 Gini
Corporate insurance look-alike	+28 Gini	**Remote medicine look-alike**	+4 Gini
Holding structure prediction	+6 Gini	**Movie recomendations cold start**	+12,7% NDCG
Scoring for small businesses	+8 Gini	**Retail credit scoring**	+4 Gini

Let’s look at embeddings

Color shows age group

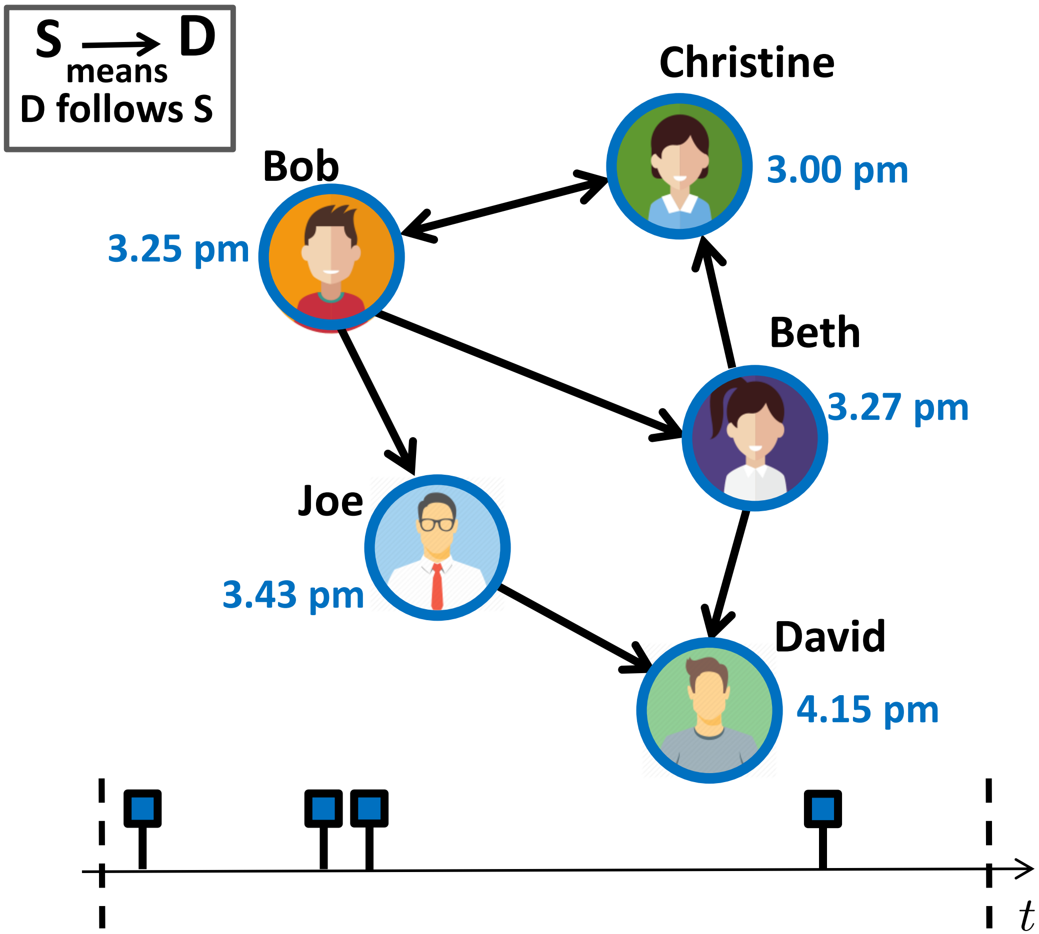


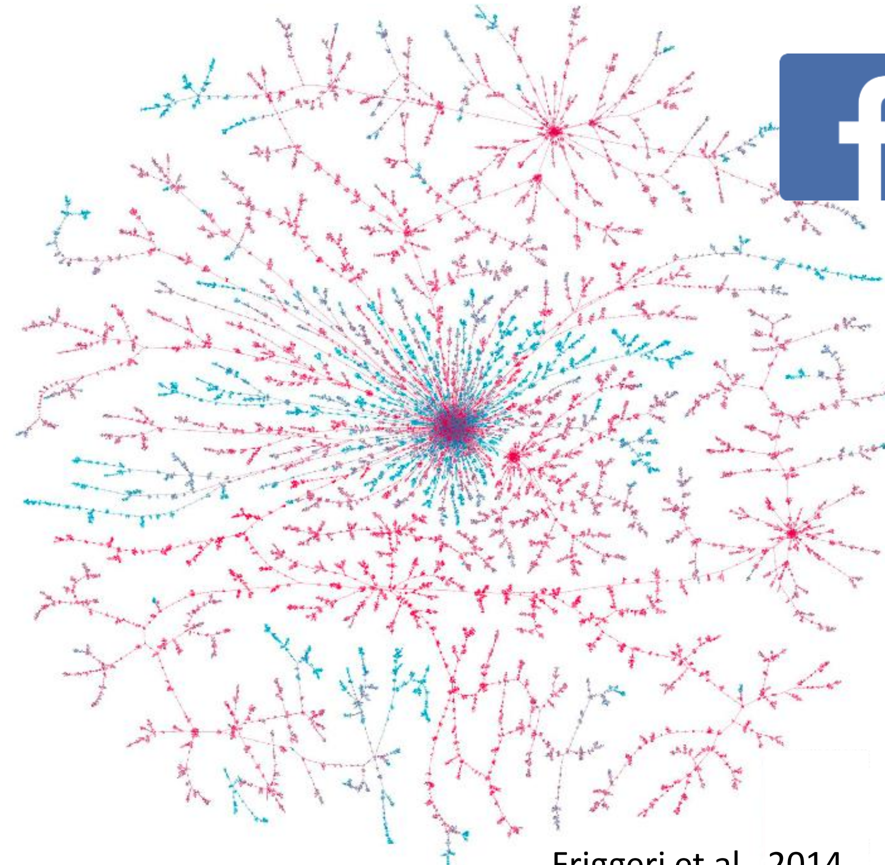
Color shows churn fact



Skoltech

Example I: Information propagation





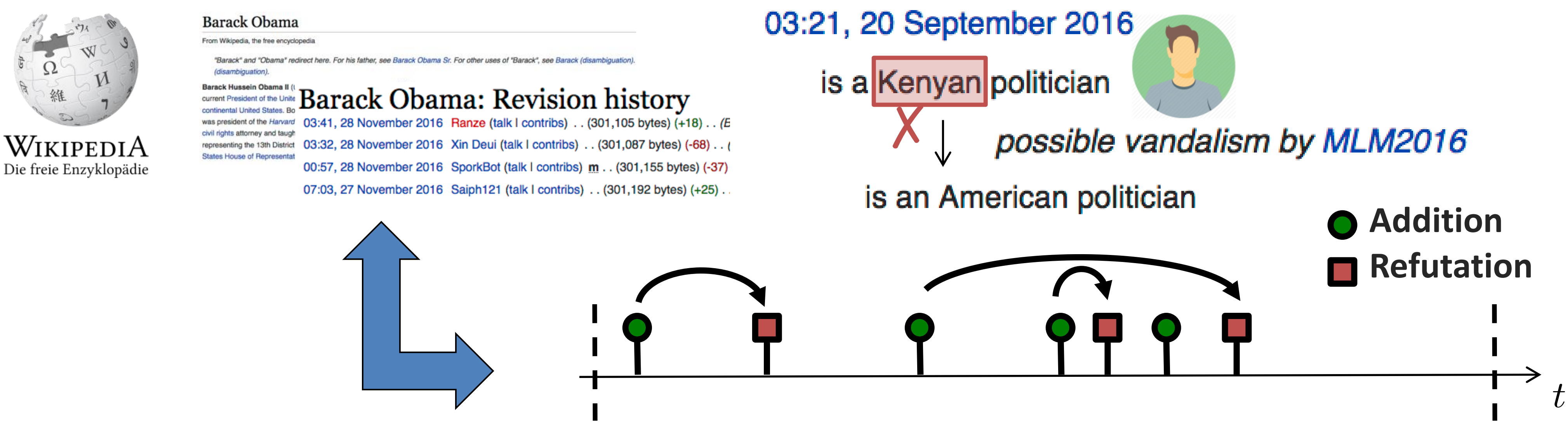
Friggeri et al., 2014

They can have an impact in the off-line world

theguardian

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

Example 2: response history



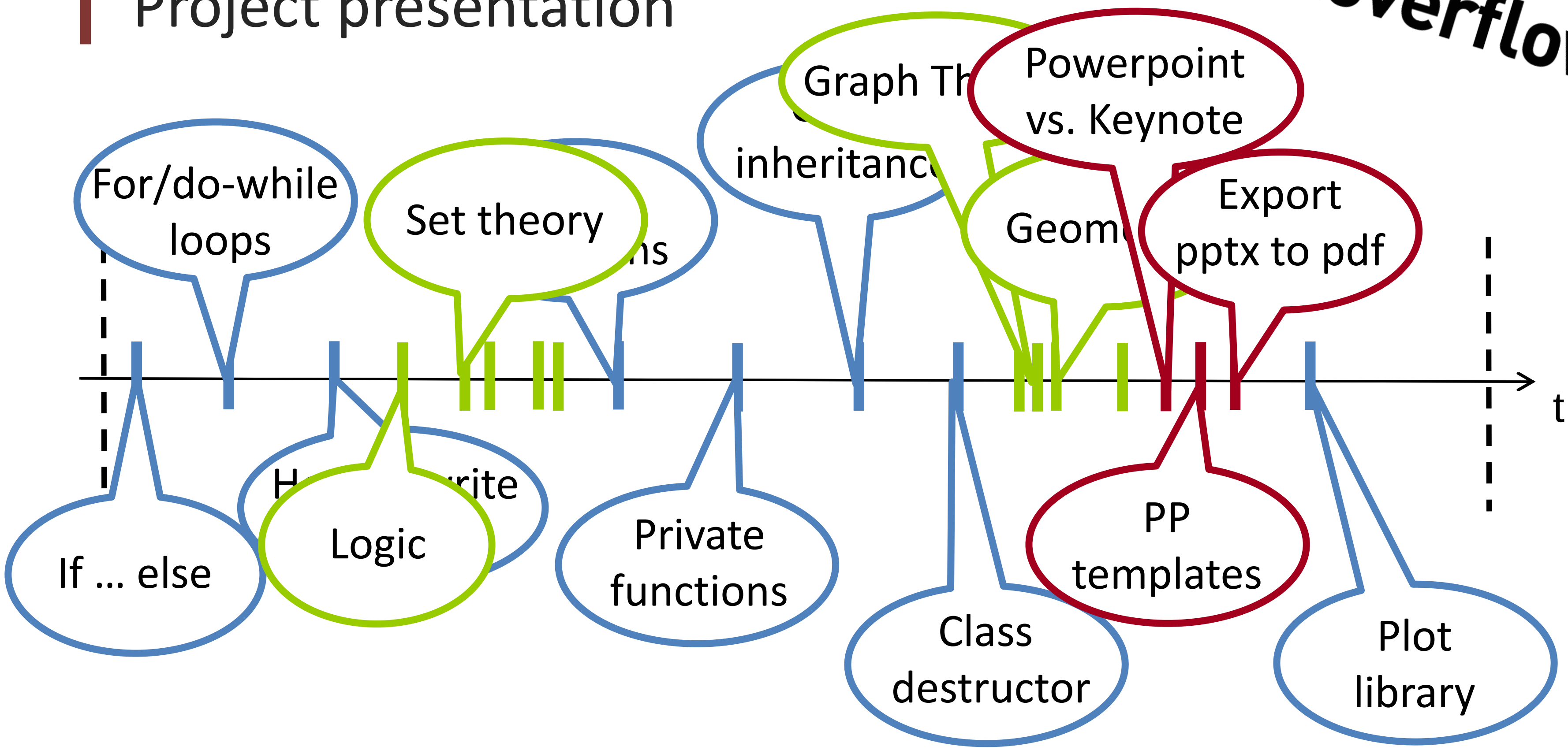
Skoltech

Example 3: development

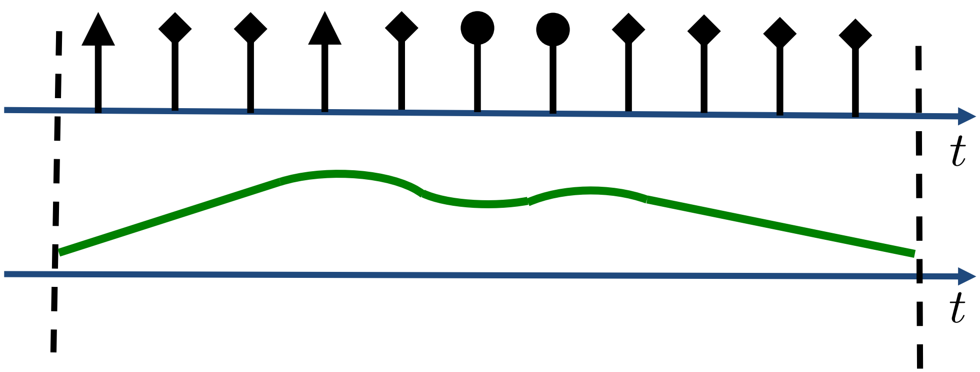
1st year computer science student

- Introduction to programming
- Discrete math
- Project presentation

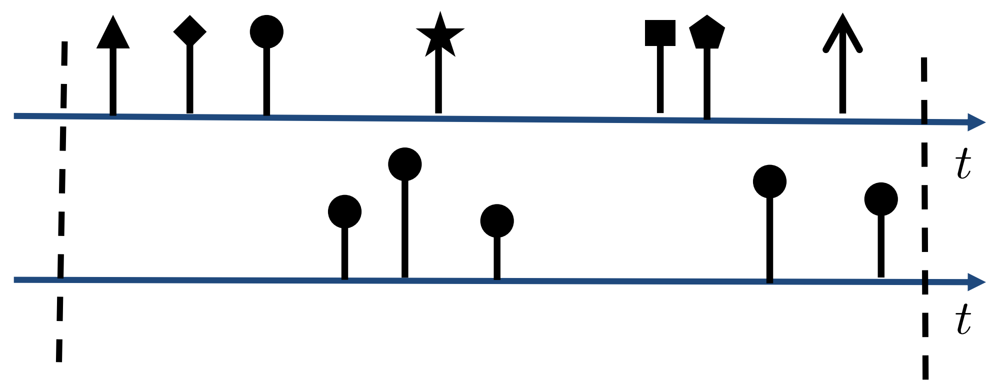
stackoverflow



Aren’t these event traces just time series?

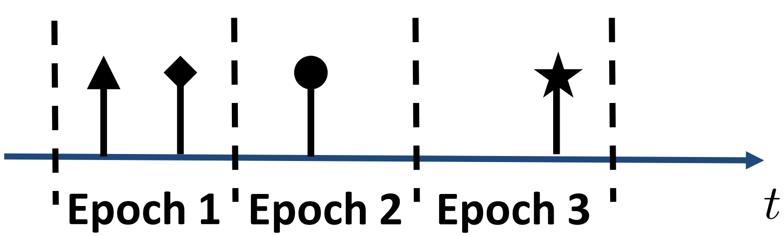


Discrete and continuous times series



Discrete events in continuous time

What about aggregating events in *epochs*?



How long is each epoch?

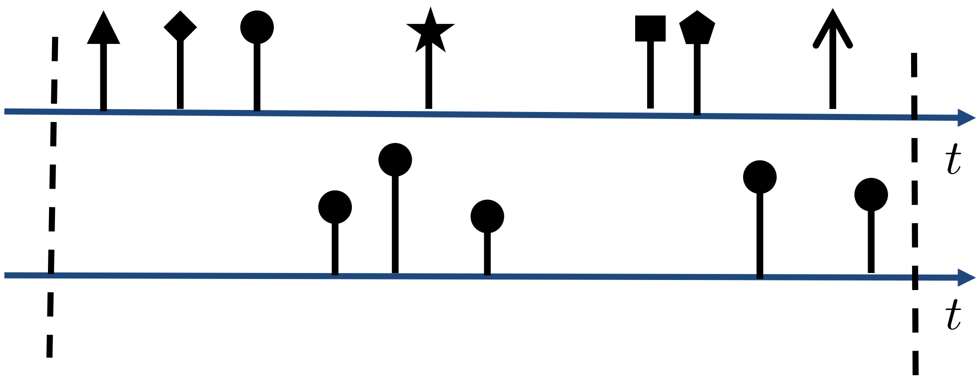
How to aggregate events per epoch?

What if no event in one epoch?

What about time-related queries?

Problems for event sequences

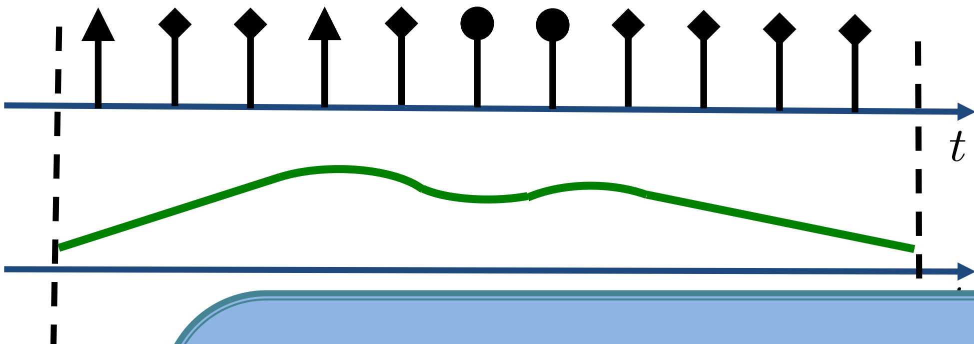
- Compact description of data: models
- Interpretation
- Forecasting/Prediction
- Control
- Hypothesis testing
- Simulation

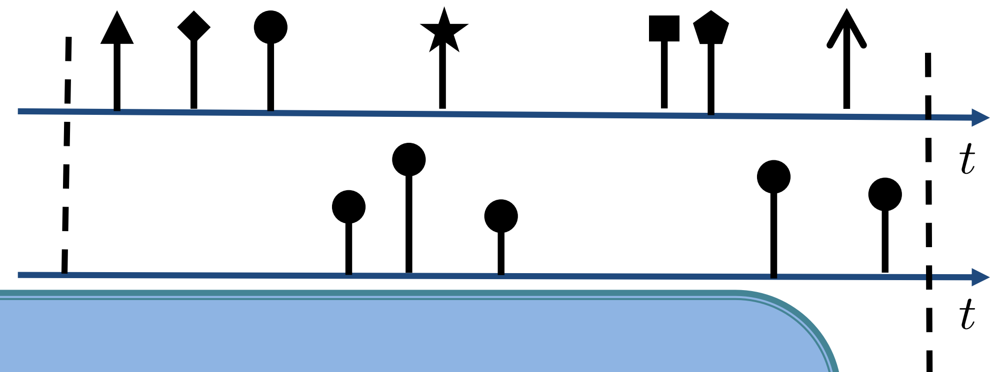


Discrete events in continuous time

Skoltech

Aren’t these event traces just time series?





The framework of

temporal point processes

provides a ***native representation***

Di

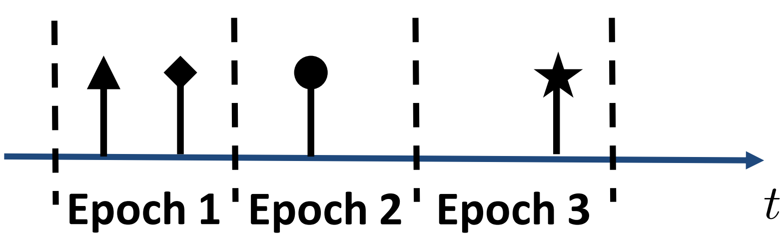
W

poch?

events in epochs?

What if no event in one epoch?

What about time-related queries?

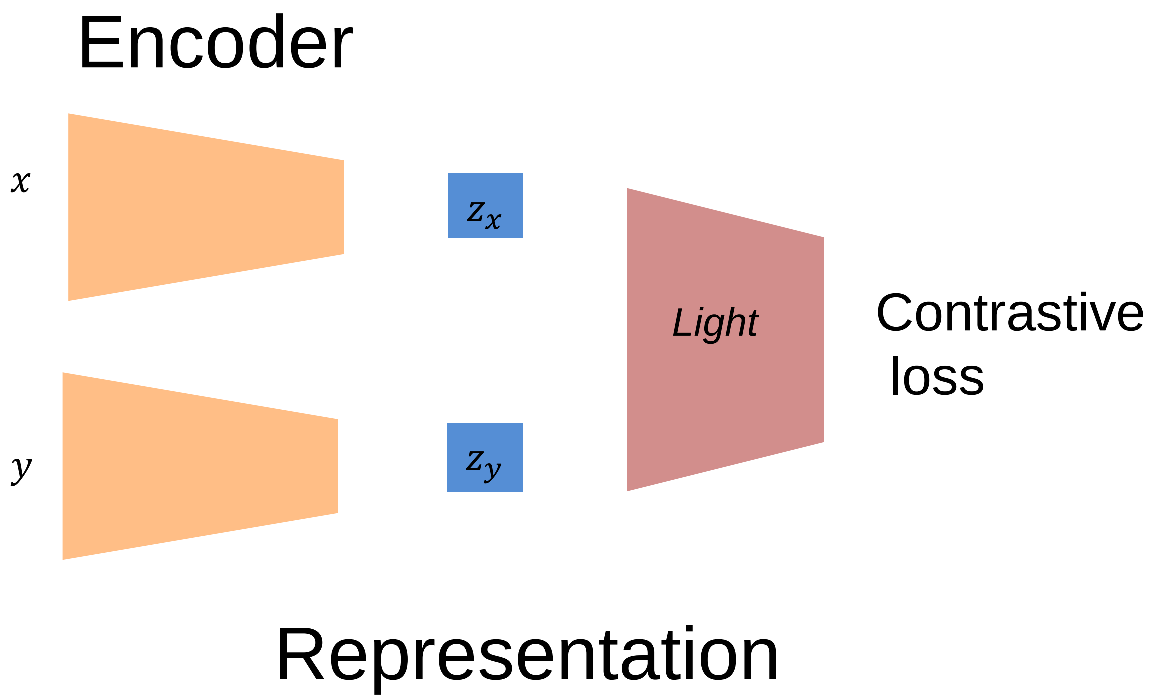


Learning universal embeddings

Contrastive approach

Skoltech

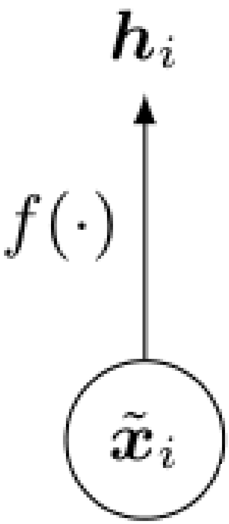
Contrastive self-supervised learning



Two models: Encoder & Discriminator.
Encoder produces representations.

What if we have labels…

Contrastive learning idea for supervised learning



Representation

Encoder

Image

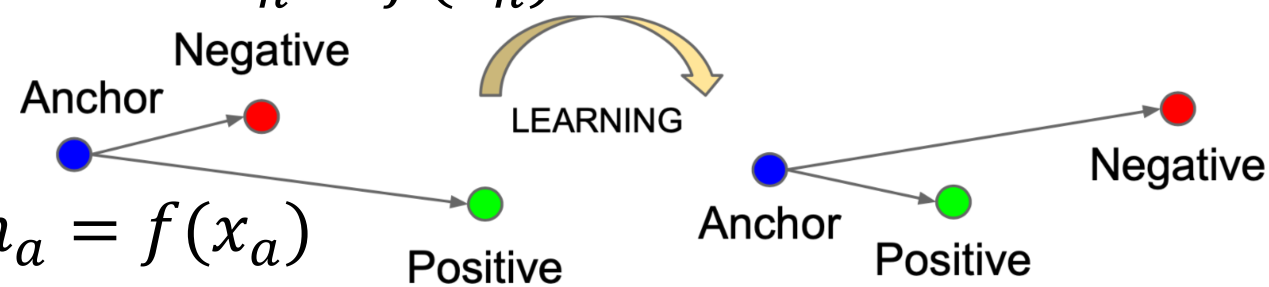
x_a x_p x_n

Anchor

Positive (similar)

Negative (different)

$h_n = f(x_n)$

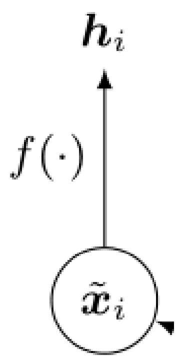


$h_a = f(x_a)$

$h_p = f(x_p)$

What if we don't have labels…

Generate views!



Representation

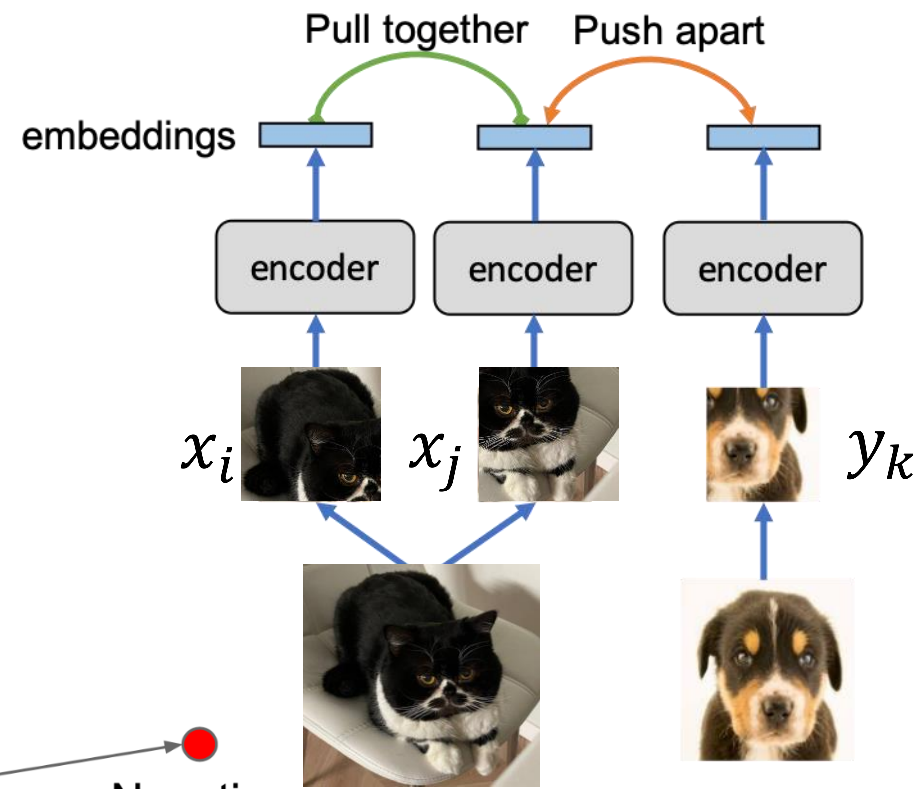
Encoder

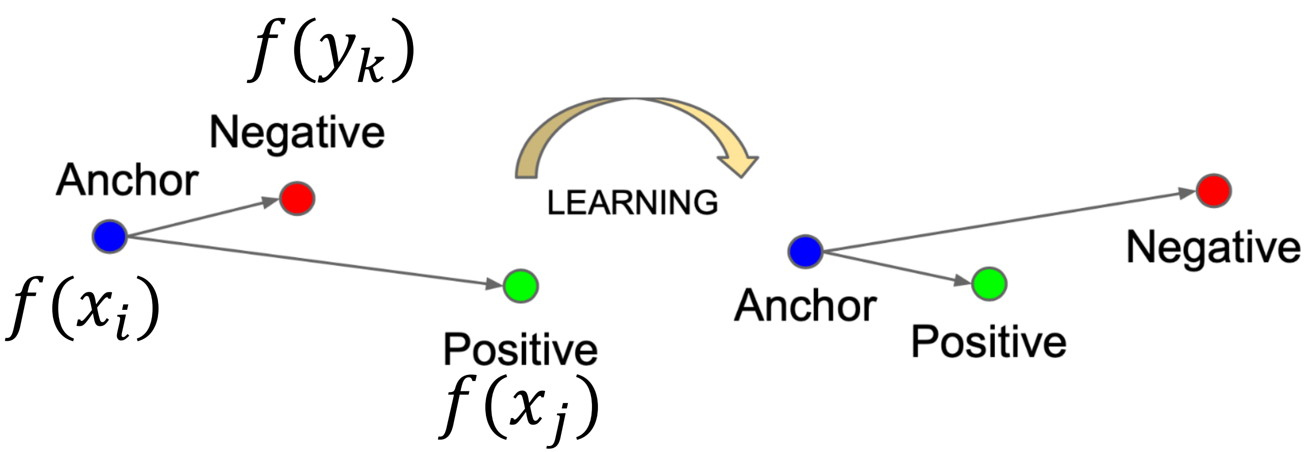
Image

x_i – one view

x_j – another view

y_k – negative example

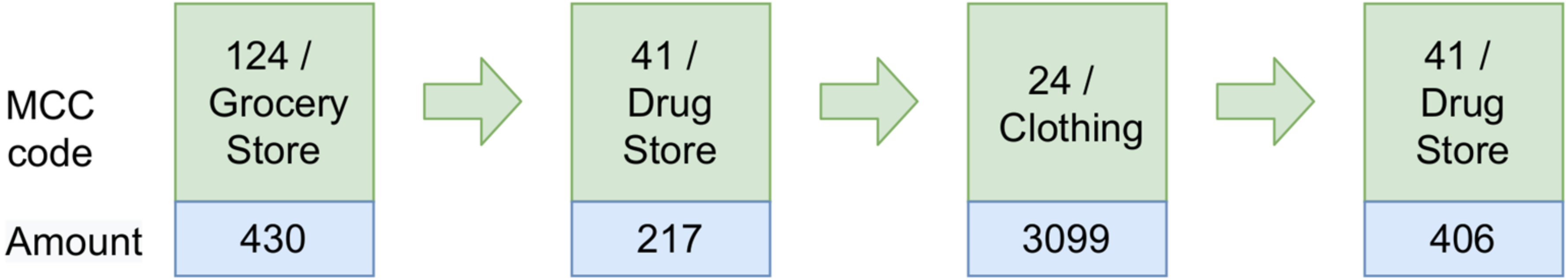




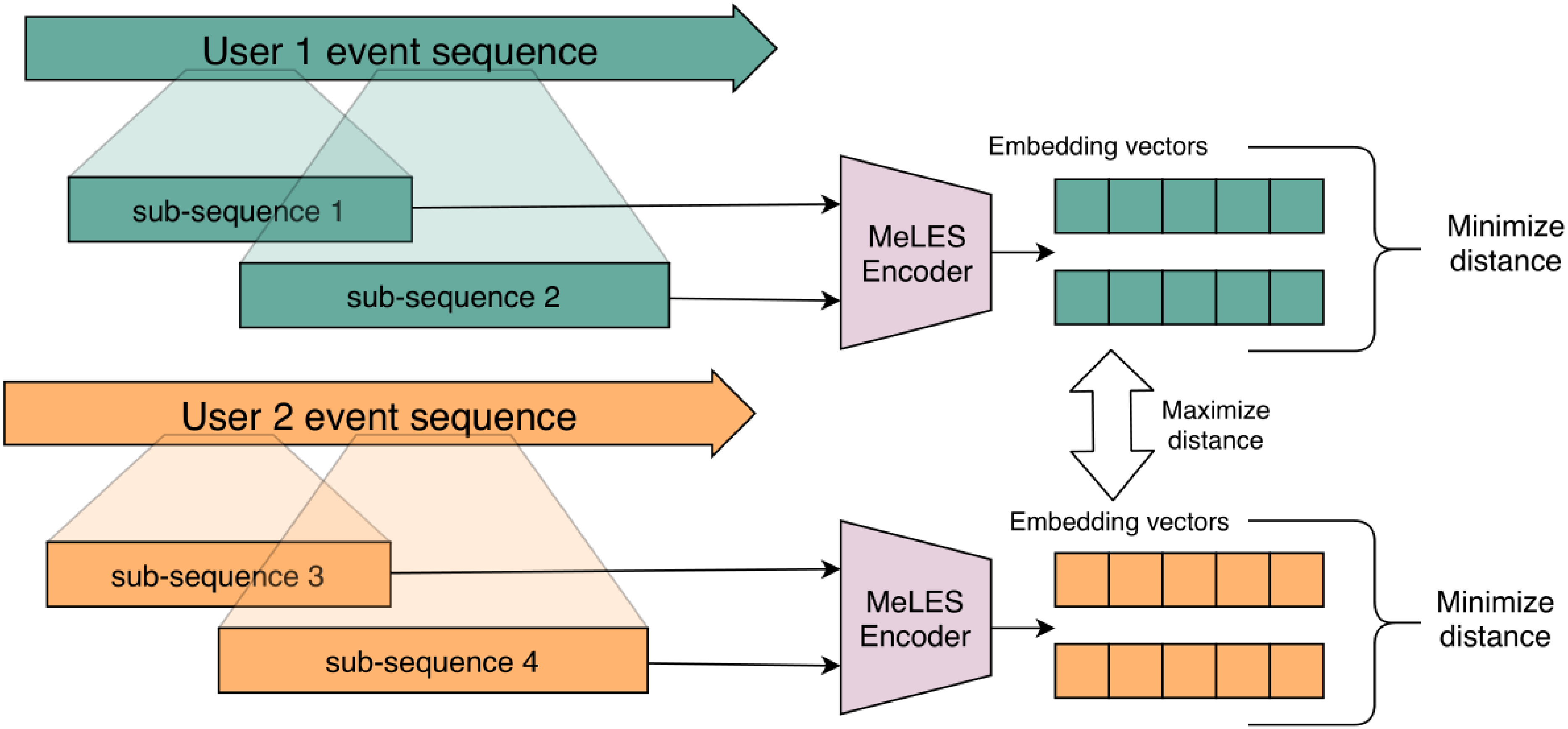
Discrete sequential transactional data

Transaction records data sequence includes:

- MCC (Merchant Category Codes)
- Purchase amount
- Time values
- Transaction location
- …



MeLES contrastive learning



https://github.com/dllllb/pytorch-lifestream

Babaev, Dmitrii, et al. "Event sequence metric learning" KDD. *2022*.

Usage of embeddings

Obtain embeddings from our event processing model.

Train a model on top of embeddings:

- SVM
- Gradient boosting
- Logistic regression

Doesn't need to access to an initial model

Alternatives:

- Fine-tuning all model: longer, required for weaker models)
- Use kNN on top of embeddings: requires stronger model

Transformers or not?

Table 4: Comparison of encoder types

Econder type	Age, Accuracy ±95%	Gender, AUROC ±95%
LSTM	0.620 ± 0.003	0.870 ± 0.005
GRU	**0.639 ± 0.006**	**0.871 ± 0.004**
Transformer	0.621 ± 0.001	0.848 ± 0.002

Table 5: Comparison of metric learning losses

Loss type	Age, Accuracy ±95%	Gender, AUROC ±95%
Contrastive loss	0.639 ± 0.006	**0.871 ± 0.003**
Binomial deviance loss	0.535 ± 0.005	0.853 ± 0.005
Histogram loss	**0.642 ± 0.002**	0.851 ± 0.004
Margin loss	0.631 ± 0.003	**0.871 ± 0.004**
Triplet loss	0.610 ± 0.006	0.855 ± 0.003

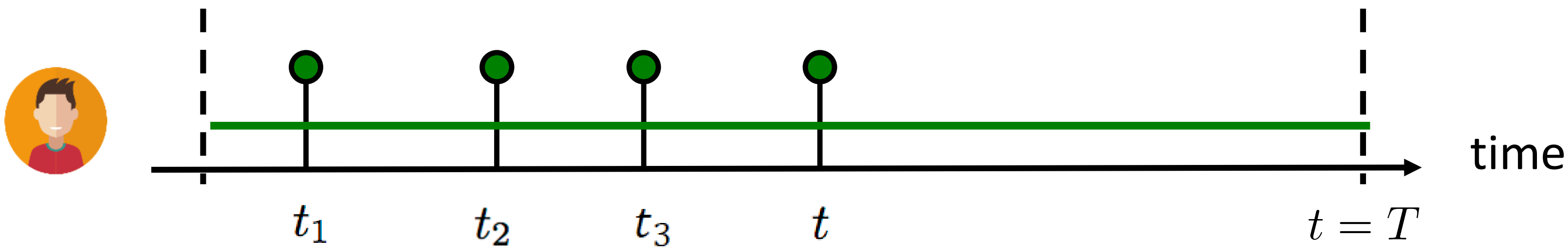
Skoltech

Temporal Point Processes (TPPs): Introduction

Poisson process and intensity function

Poisson process

Coarse approximation of many real-life processes



Intensity is the expected number of events per unit time.

Intensity of a Poisson process is constant:

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

Observations:

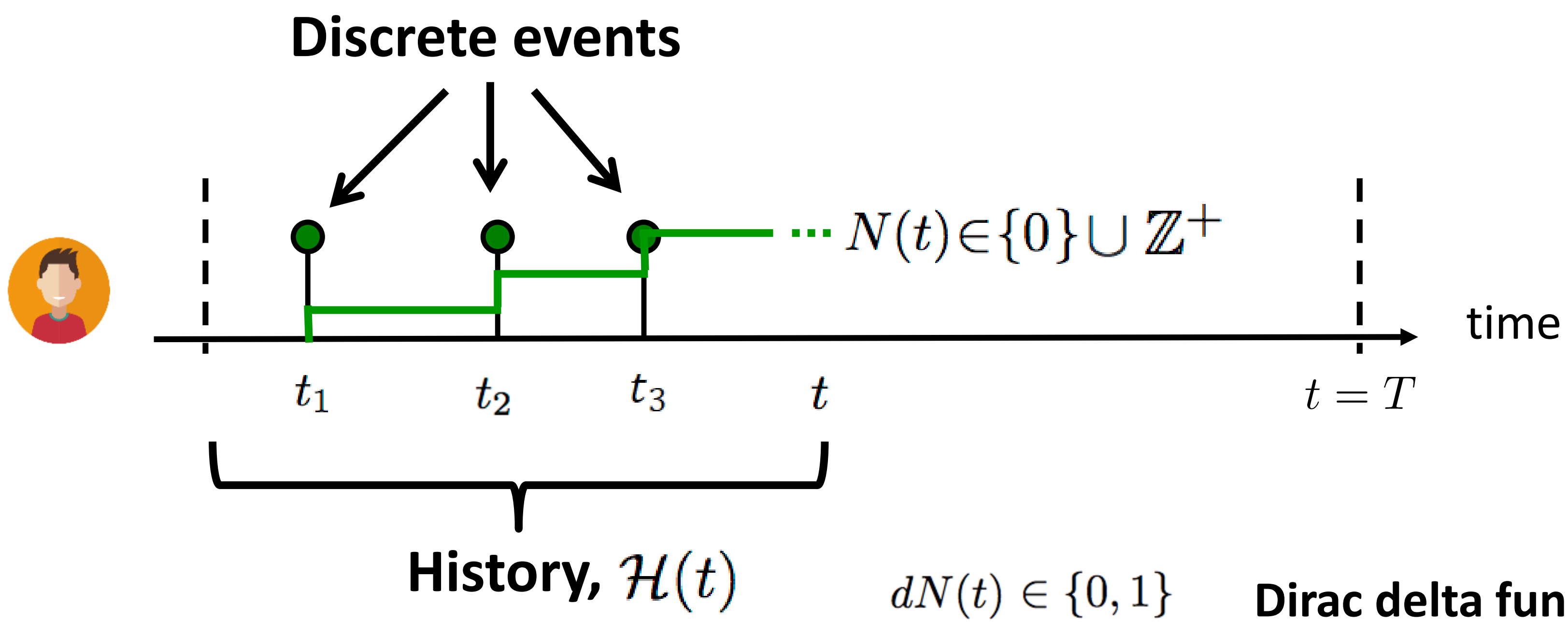
1. Intensity independent of history
2. Uniformly random occurrence

Skoltech

Temporal point processes

Temporal point process:

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

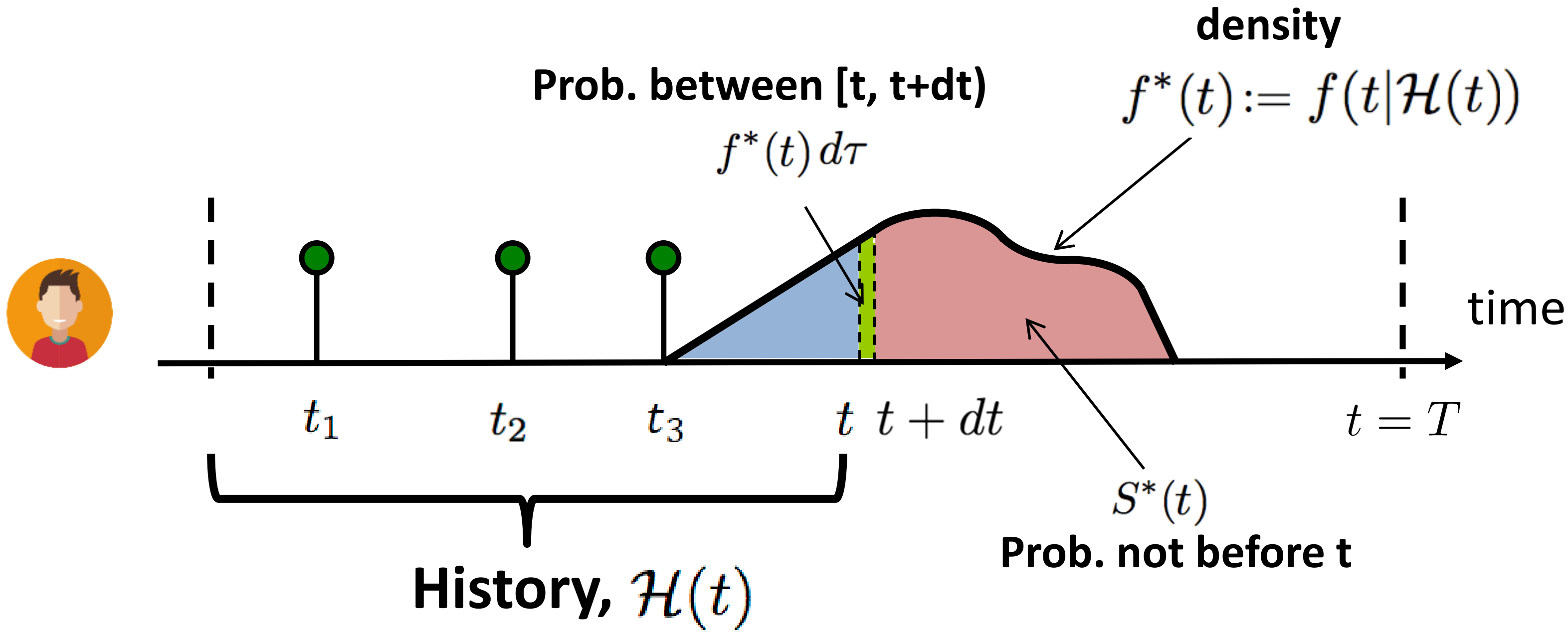


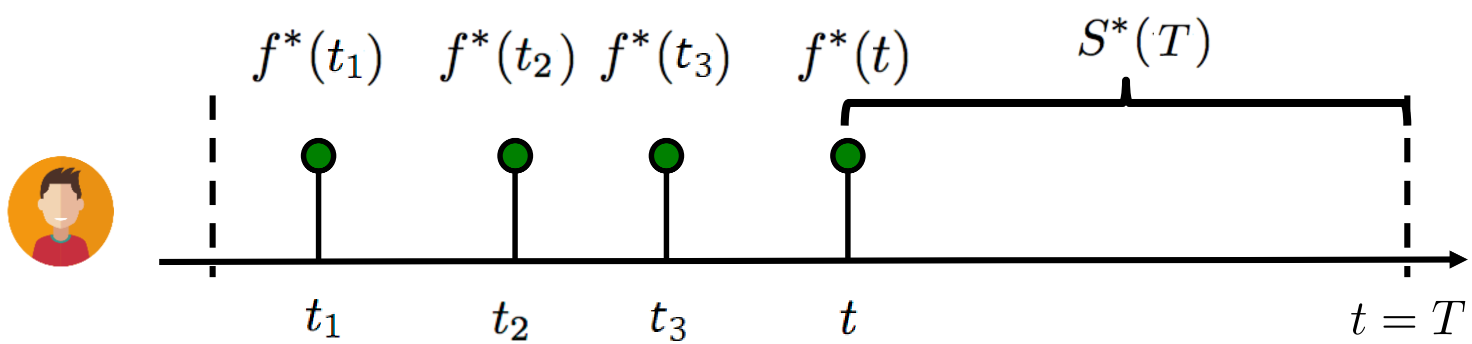
$dN(t) \in \{0, 1\}$

Dirac delta function

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

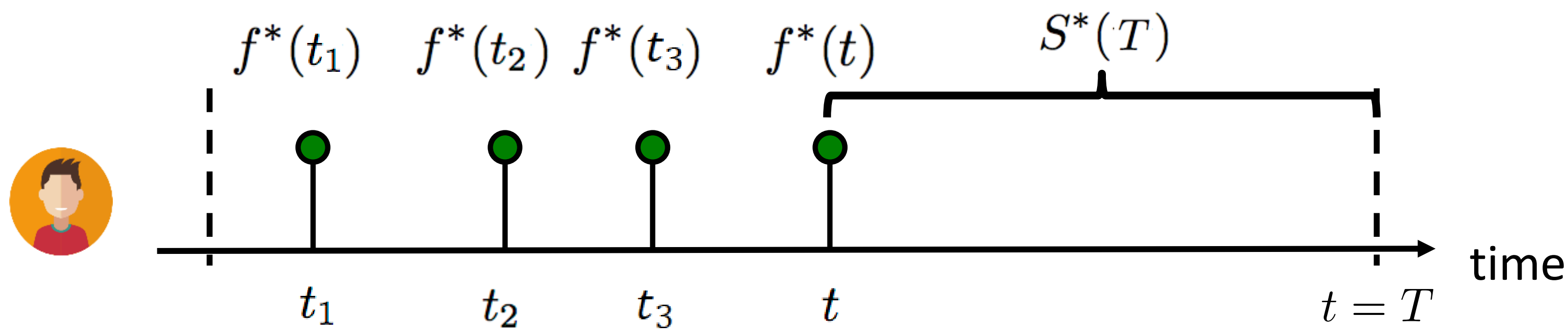




Likelihood of a timeline: $f^*(t_1)\ f^*(t_2)\ f^*(t_3)\ f^*(t)\ S^*(T)$

Skoltech

Problems of density parametrization (I)



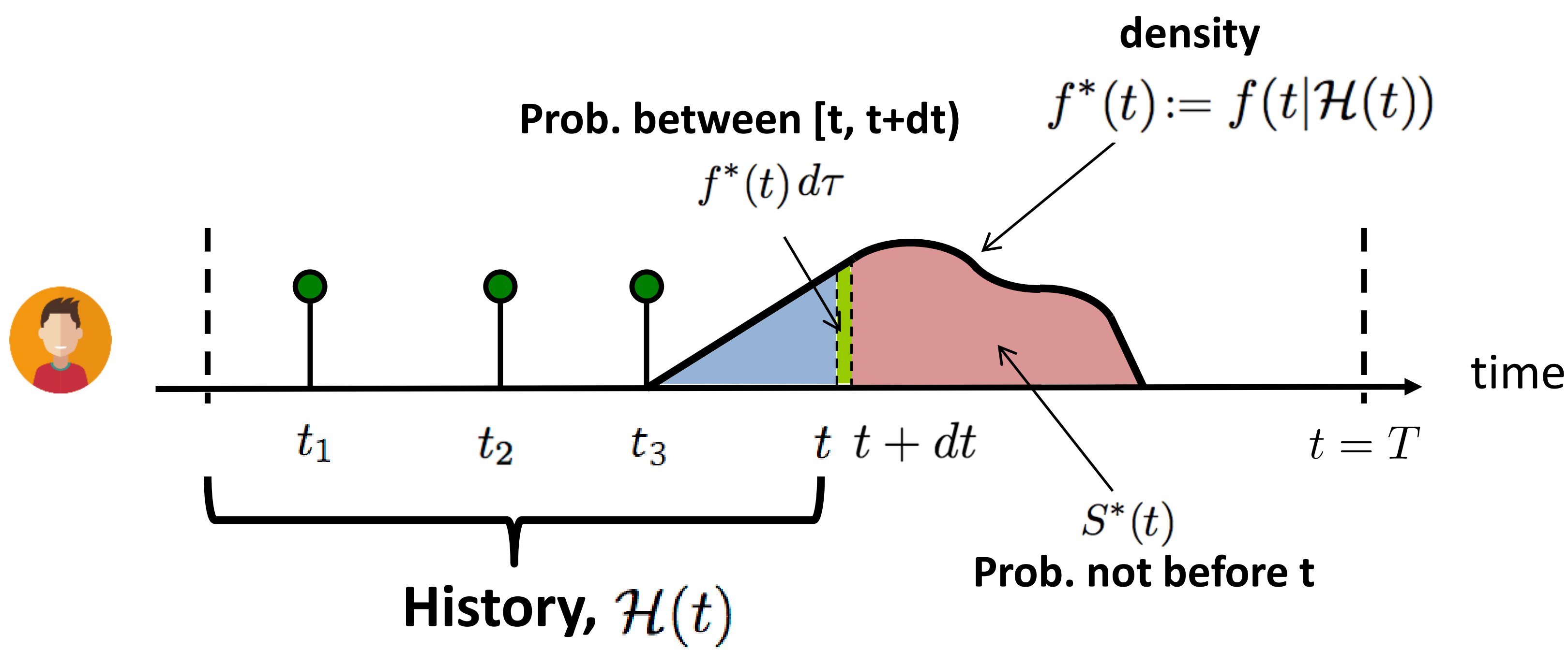
$$f^*(t_1) \quad f^*(t_2) \quad f^*(t_3) \quad f^*(t) \quad S^*(T)$$

$$\frac{\exp\langle w, \psi^*(t_1)\rangle}{Z} \quad \frac{\exp\langle w, \psi^*(t_2)\rangle}{Z} \quad \frac{\exp\langle w, \psi^*(t_3)\rangle}{Z} \quad \frac{\exp\langle w, \psi^*(t)\rangle}{Z} \quad 1 - \int_t^T \frac{\exp\langle w, \psi^*(\tau)\rangle}{Z} d\tau$$

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



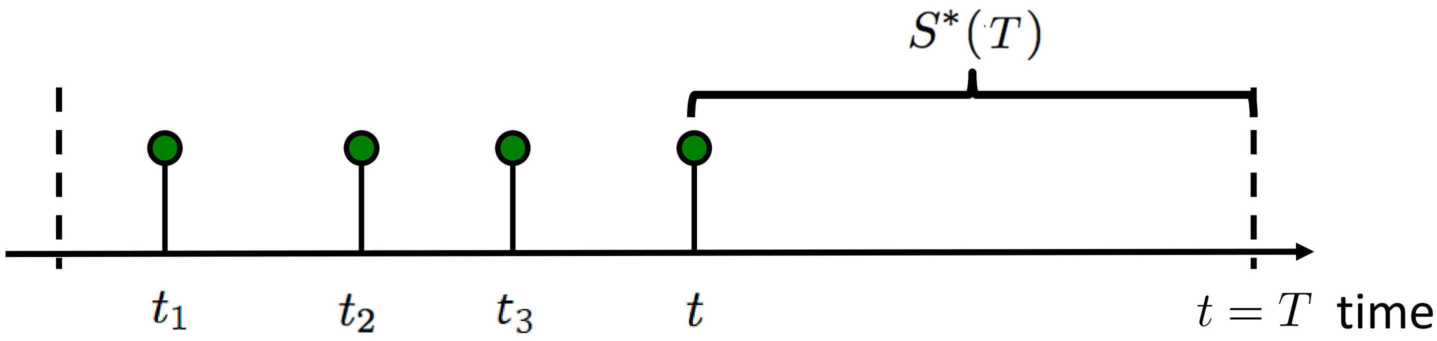
Intensity:

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

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

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

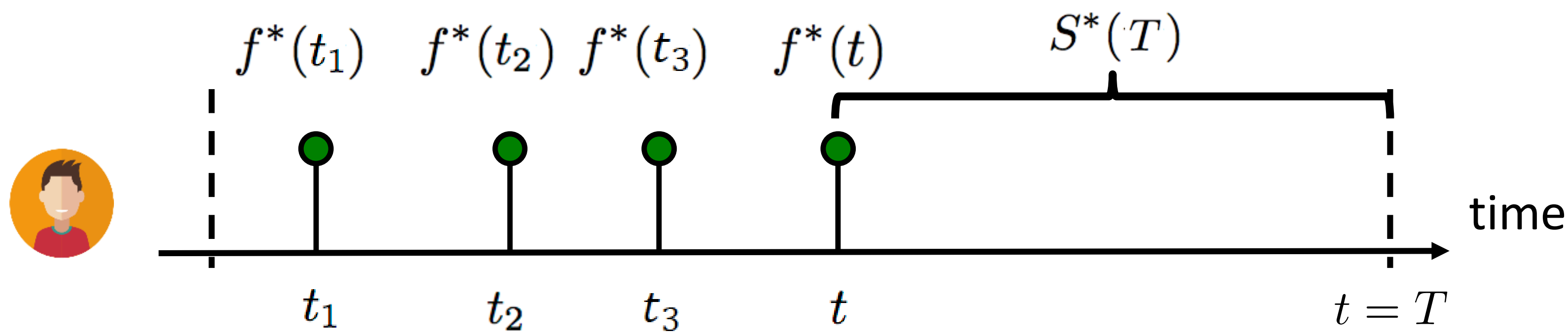
Log likelihood via intensity and density



$$L = f(t_1|\mathcal{H}_0) f(t_2|\mathcal{H}_{t_1}) \cdots f(t_n|\mathcal{H}_{t_{n-1}})(1 - F(T|\mathcal{H}_{t_n}))$$

$$\begin{aligned} L &= \left(\prod_{i=1}^{n} f(t_i|\mathcal{H}_{t_{i-1}})\right) \frac{f(T|\mathcal{H}_{t_n})}{\lambda^*(T)} \\ &= \left(\prod_{i=1}^{n} \lambda^*(t_i) \exp\left(-\int_{t_{i-1}}^{t_i} \lambda^*(s) \mathrm{d}s\right)\right) \exp\left(-\int_{t_n}^{T} \lambda^*(s) \mathrm{d}s\right) \\ &= \left(\prod_{i=1}^{n} \lambda^*(t_i)\right) \exp\left(-\int_{0}^{T} \lambda^*(s) \mathrm{d}s\right), \end{aligned}$$

Advantages of intensity parametrization (I)



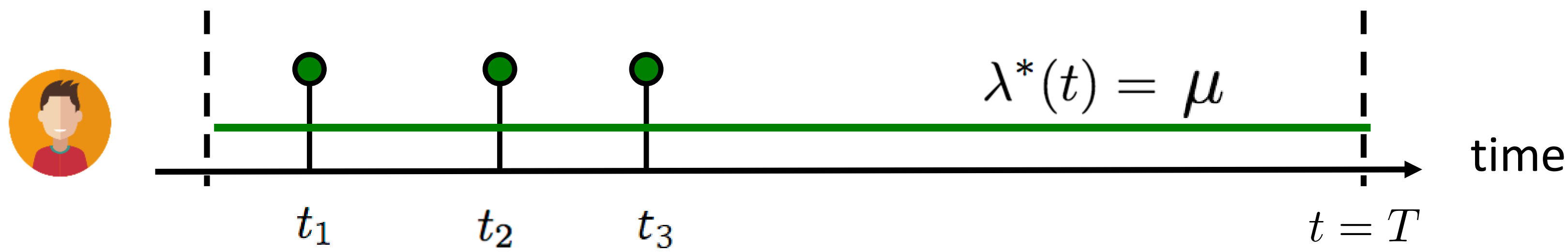
$$\lambda^*(t_1)\ \lambda^*(t_2)\ \lambda^*(t_3)\ \lambda^*(t)\ \exp\left(-\int_0^T \lambda^*(\tau)\,d\tau\right)$$

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

Suitable for model design and interpretable:

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

Fitting & sampling from a Poisson



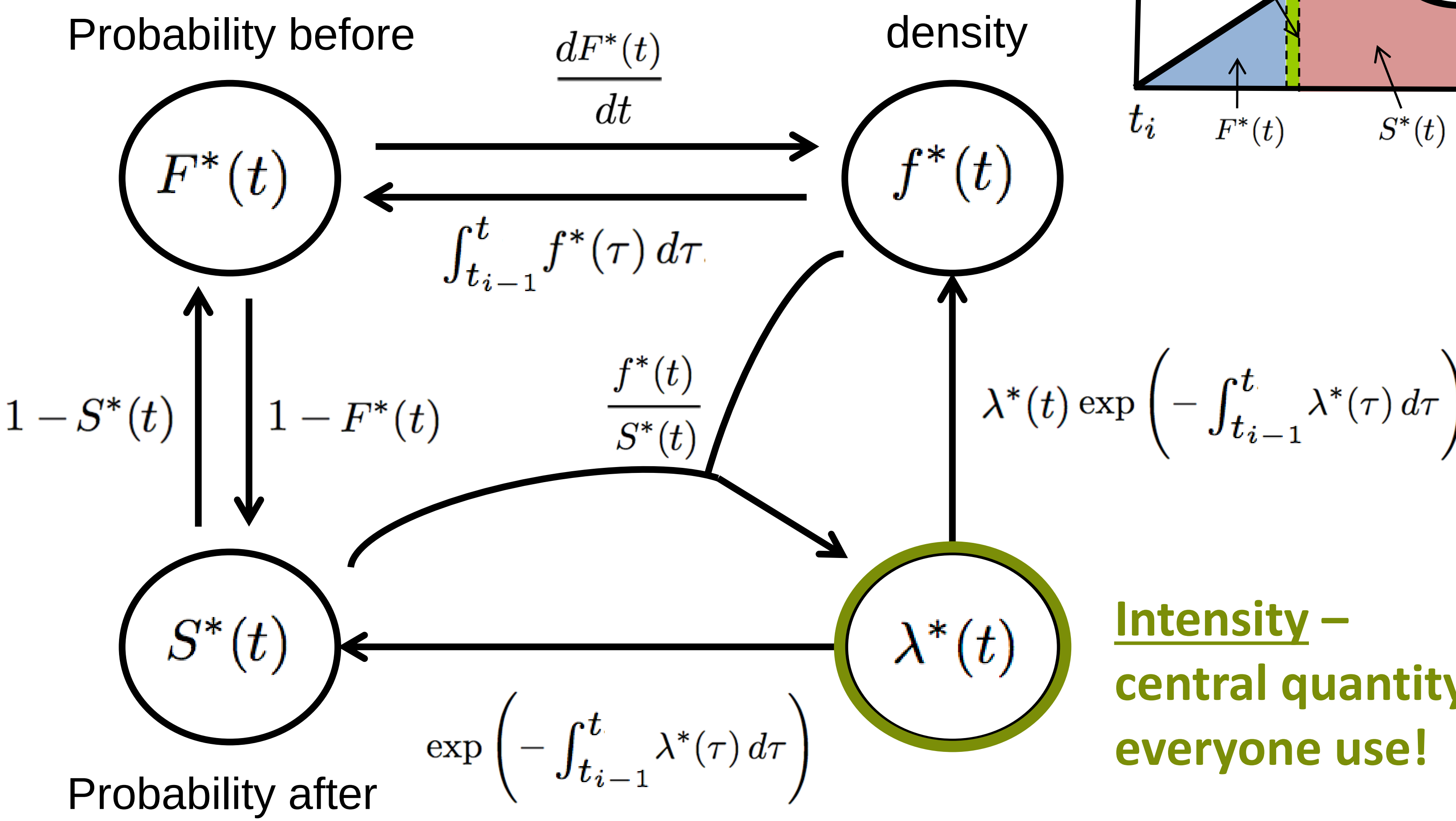
Fitting by maximum likelihood:

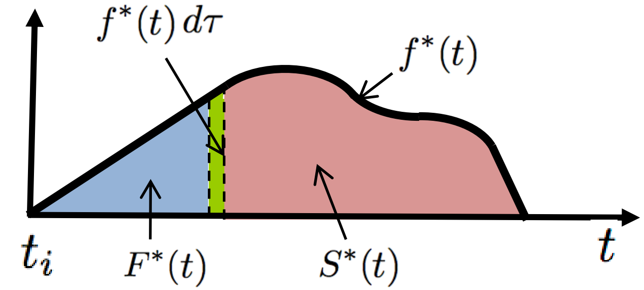
$$\mu^* = \underset{\mu}{\operatorname{argmax}} \; 3 \log \mu - \mu T = \frac{3}{T}$$

Sampling using inversion sampling:

$$t \sim \underbrace{\mu \exp(-\mu(t - t_3))}_{f_t^*(t)} \Rightarrow t = \underbrace{-\frac{1}{\mu} \log(1 - \overset{Uniform(0,1)}{\overset{\downarrow}{u}}) + t_3}_{F_t^{-1}(u)}$$

Relation between f*, F*, S*, λ*





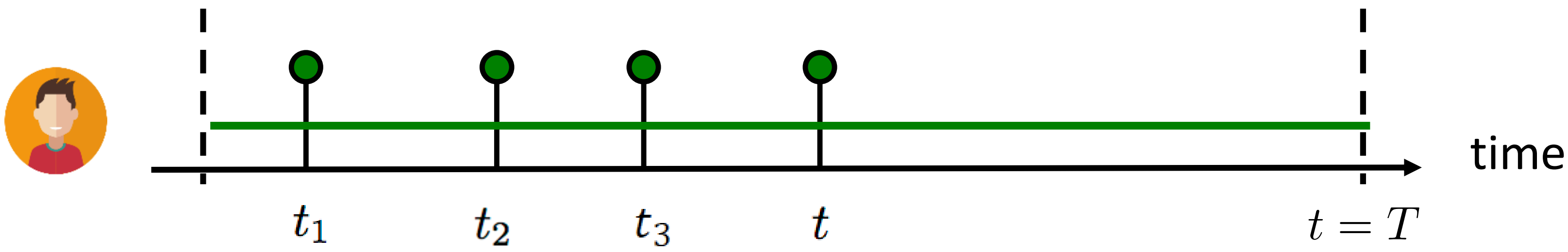
<u>Intensity</u> – central quantity everyone use!

Representation:
Temporal Point Processes

Examples of processes

Poisson process

Coarse approximation of many real-life processes



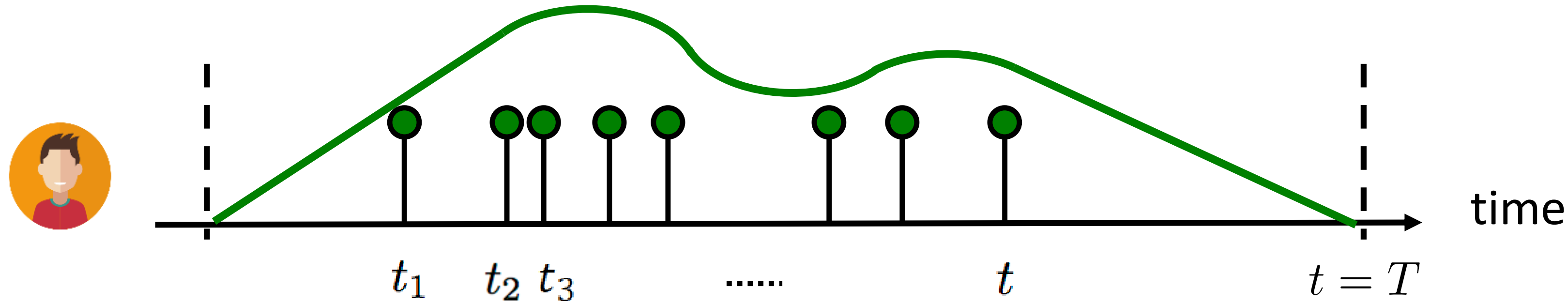
Intensity of a Poisson process

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

Observations:

1. Intensity independent of history
2. Uniformly random occurrence
3. Time interval follows exponential distribution

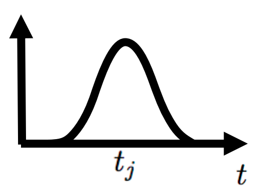
Inhomogeneous Poisson process



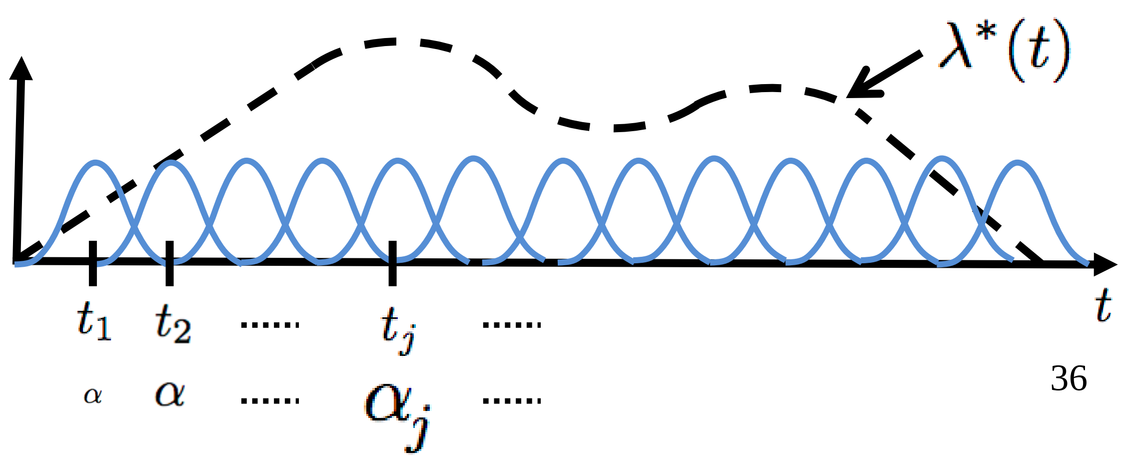
Intensity of an inhomogeneous Poisson process

$$\lambda^*(t) = g(t) \geqslant 0$$ – Independent of history

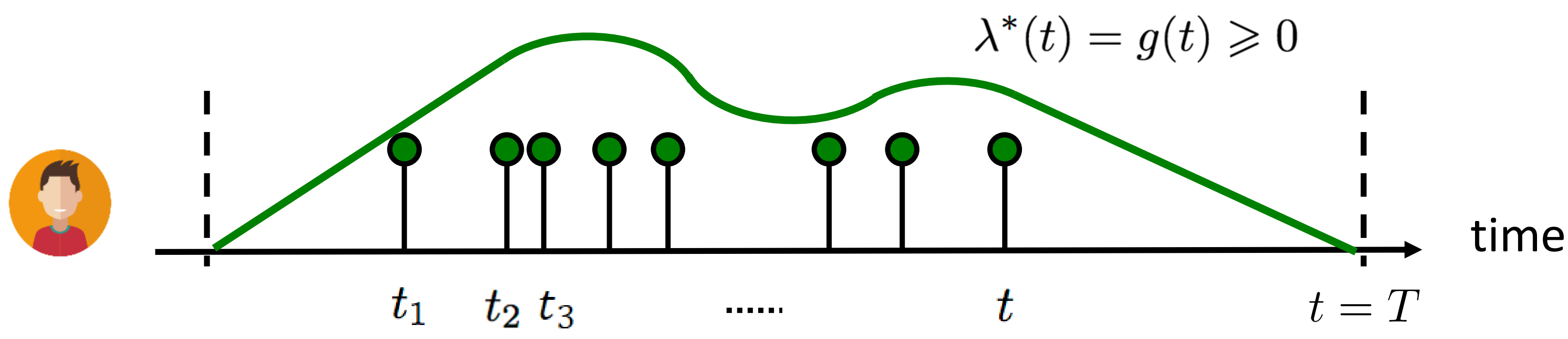
Example:



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



Fitting from inhomogeneous Poisson



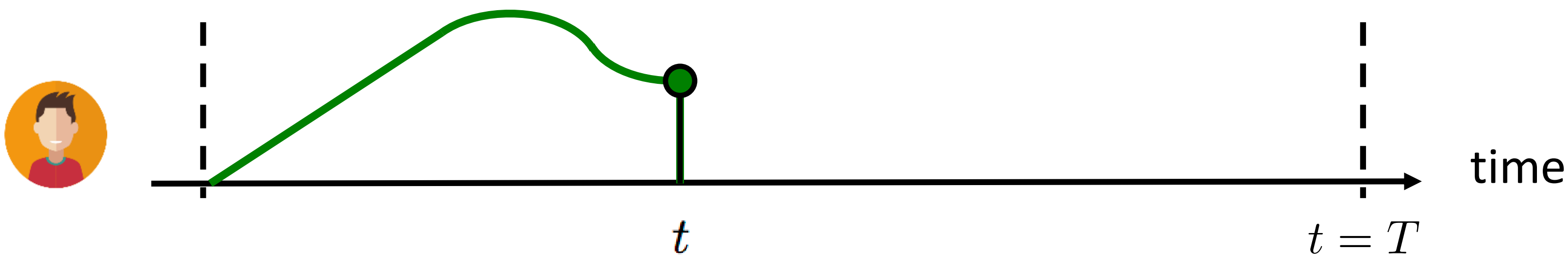
Fitting by maximum log-likelihood:

$$\underset{g(t)}{\text{maximize}} \sum_{i=1}^{n} \log g(t_i) - \int_0^T g(\tau)\, d\tau$$

<u>Idea:</u> we have additional features, so we can use a generalized linear model for it

Intensity is $g(t) = g(\boldsymbol{x}_t) = \exp(\boldsymbol{x}_t^T w)$

Terminating (or survival) process



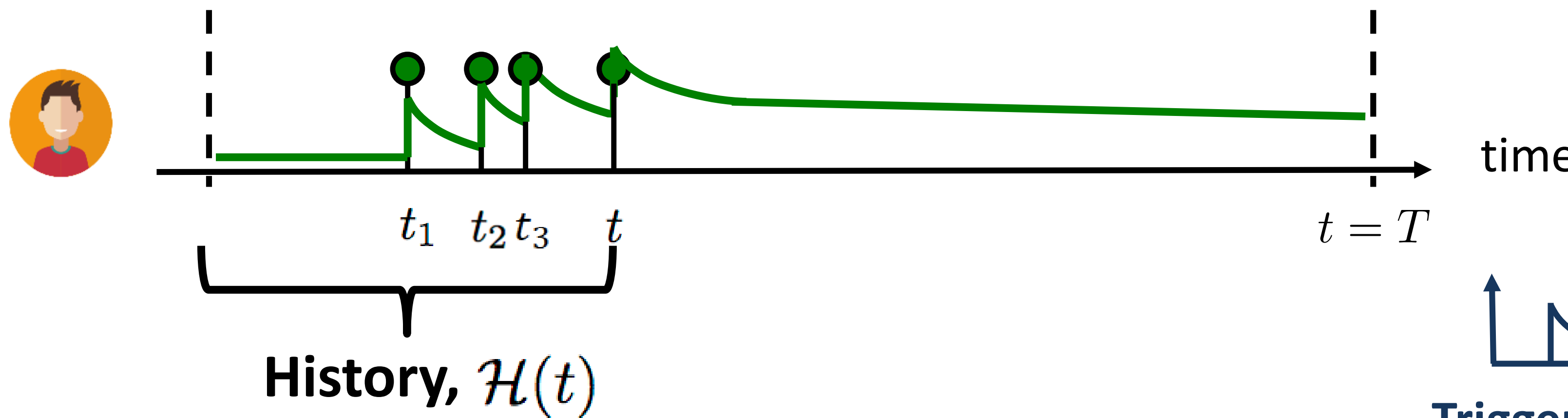
Intensity of a terminating (or survival) process

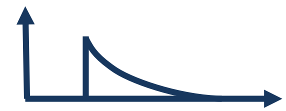
$$\lambda^*(t) = g^*(t)(1 - N(t)) \geqslant 0$$

Observations:

1. **Limited number of occurrences**
2. **Hazard function in actuarial science**

Self-exciting Hawkes process





Intensity of self-exciting (or Hawkes) process:

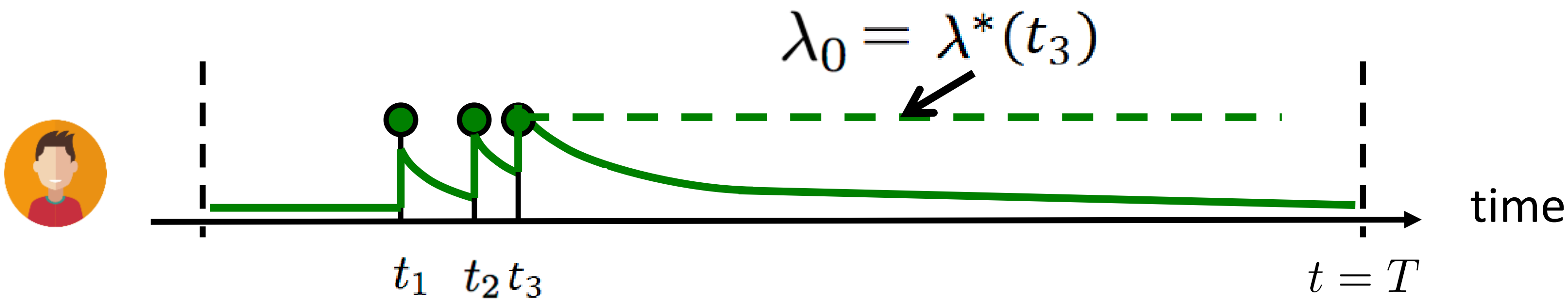
$$\lambda^*(t) = \mu + \alpha \sum_{t_i \in \mathcal{H}(t)} \overbrace{\kappa_\omega(t - t_i)}^{\text{Triggering kernel}}$$
$$= \mu + \alpha \kappa_\omega(t) \star dN(t)$$

Observations:

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

Hawkes, Alan G. "Spectra of some self-exciting and mutually exciting point processes." *Biometrika* 58.1 (1971): 83-90.

Fitting a Hawkes process from a recorded timeline



Fitting by maximum likelihood:

$$\underset{\mu,\alpha}{\text{maximize}} \sum_{i=1}^{n} \log \lambda^*(t_i) - \int_0^T \lambda^*(\tau)\, d\tau$$

The max. likelihood is jointly convex in μ and α

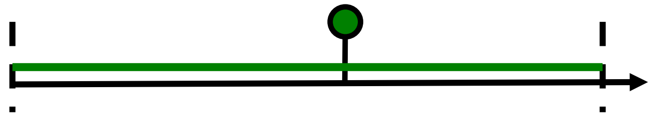
Skoltech

Summary

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

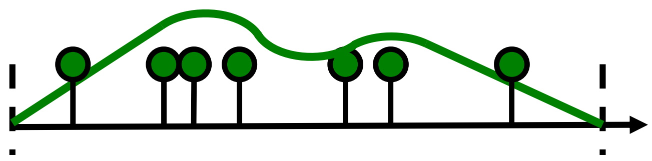
Poisson processes:

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



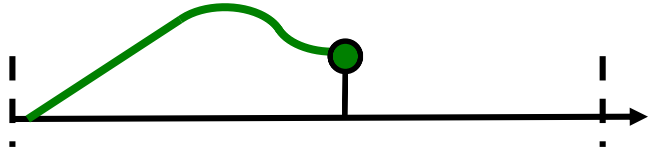
Inhomogeneous Poisson processes:

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



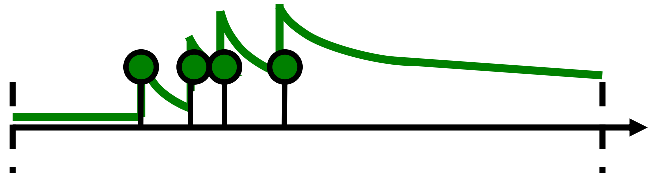
Terminating point processes:

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



Self-exciting point processes:

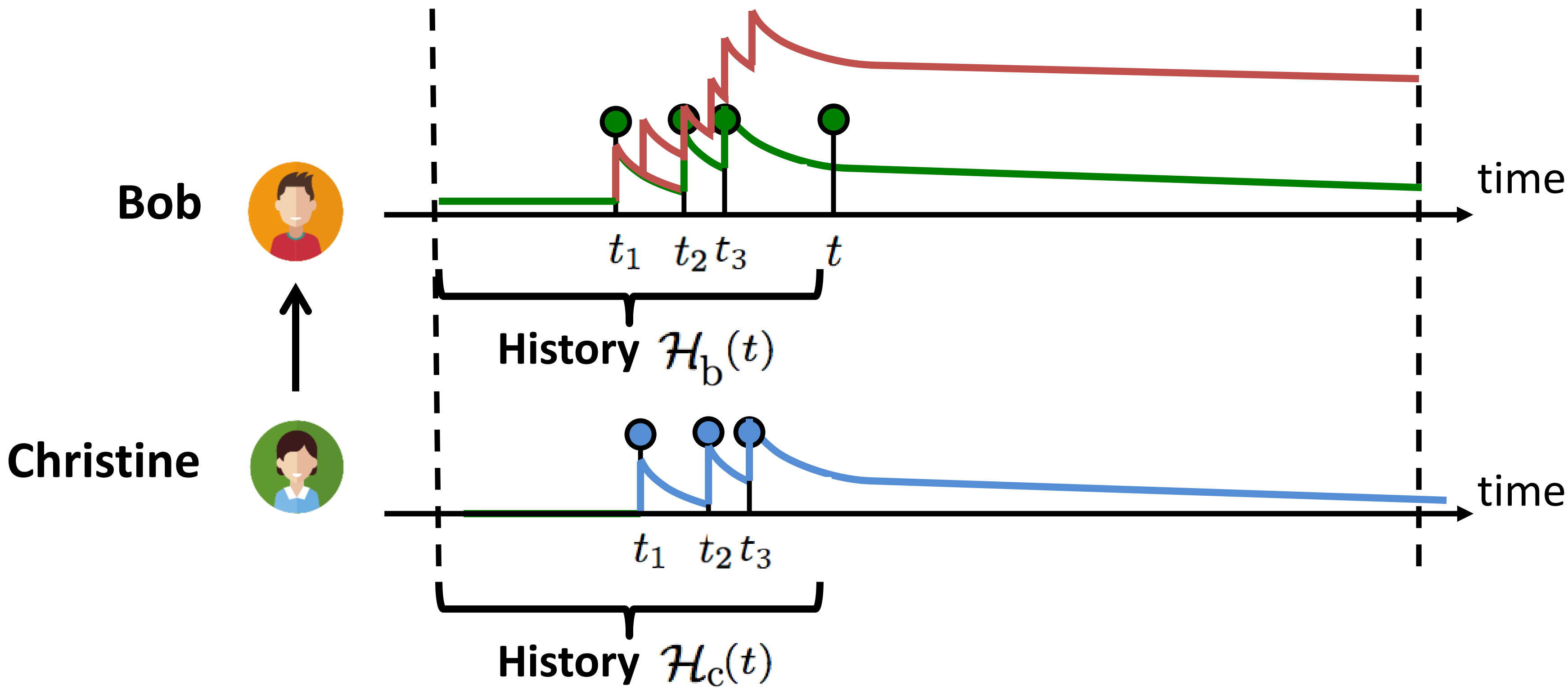
$$\lambda^*(t) = \mu + \alpha \sum_{t_i \in \mathcal{H}(t)} \kappa_\omega(t - t_i)$$



Skoltech

Temporal Point Processes: other ideas

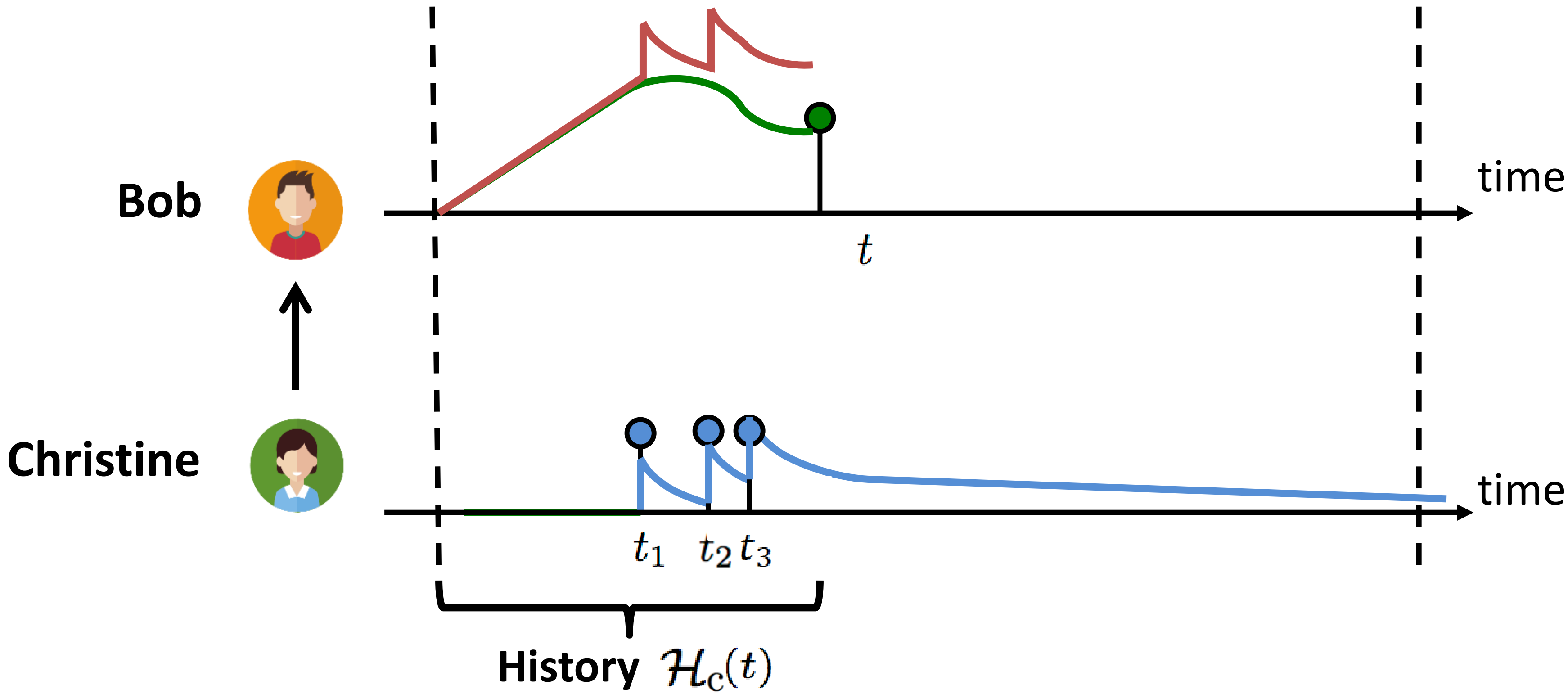
Mutually exciting process



Clustered occurrence affected by neighbors

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

Mutually exciting terminating process



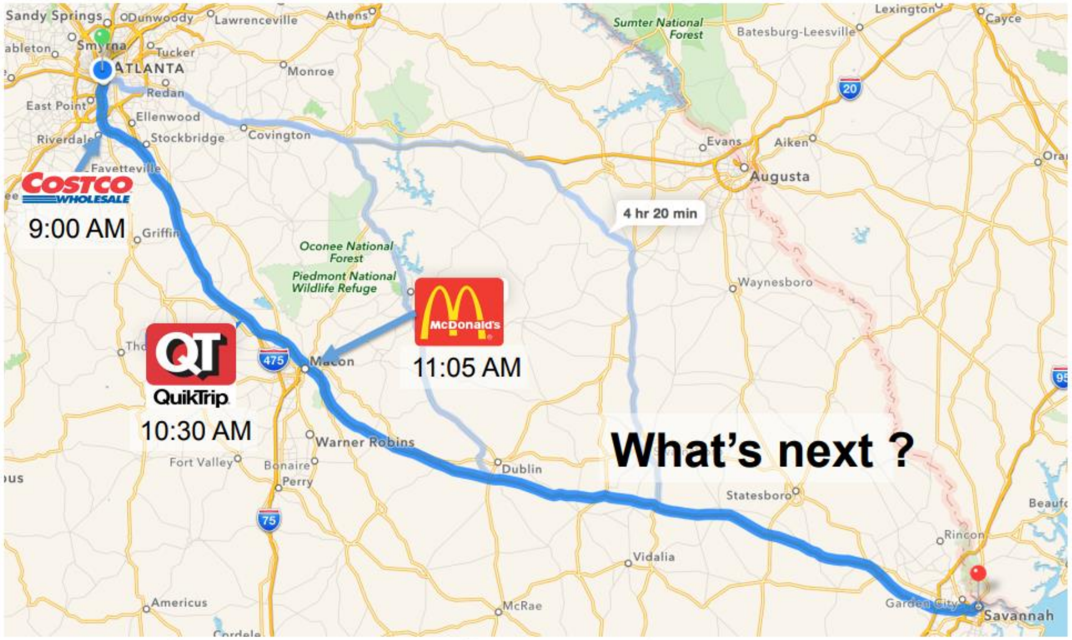
Clustered occurrence affected by neighbors

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

<u>Marked</u> temporal point processes

Marked temporal point process:

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



Given the trace of past locations and time, can we predict the location and time of the next stop?

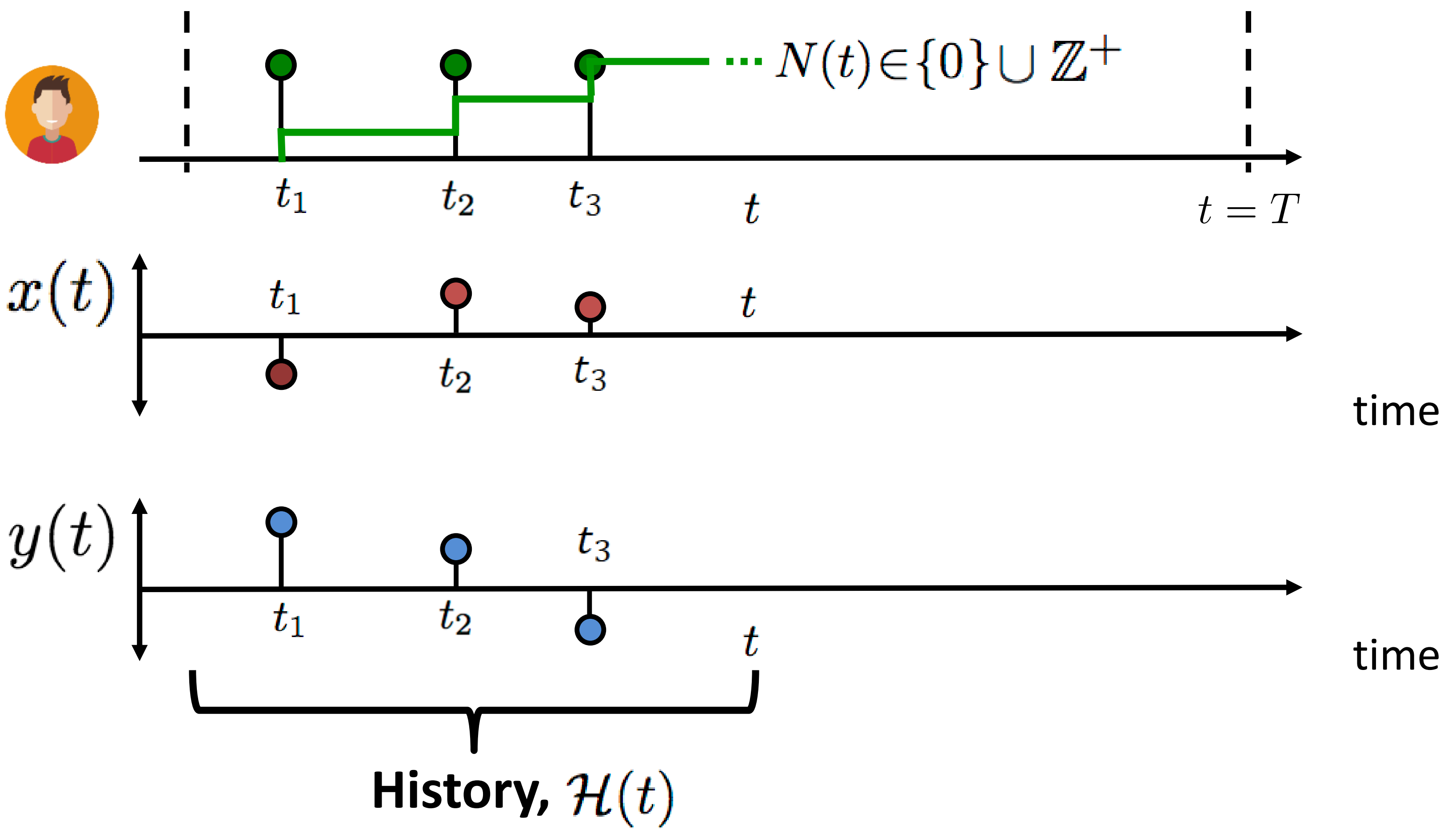
Figure source: Du, Nan, et al. "Recurrent marked temporal point processes: Embedding event history to vector." KDD. 2016.

Skoltech

Marked temporal point processes

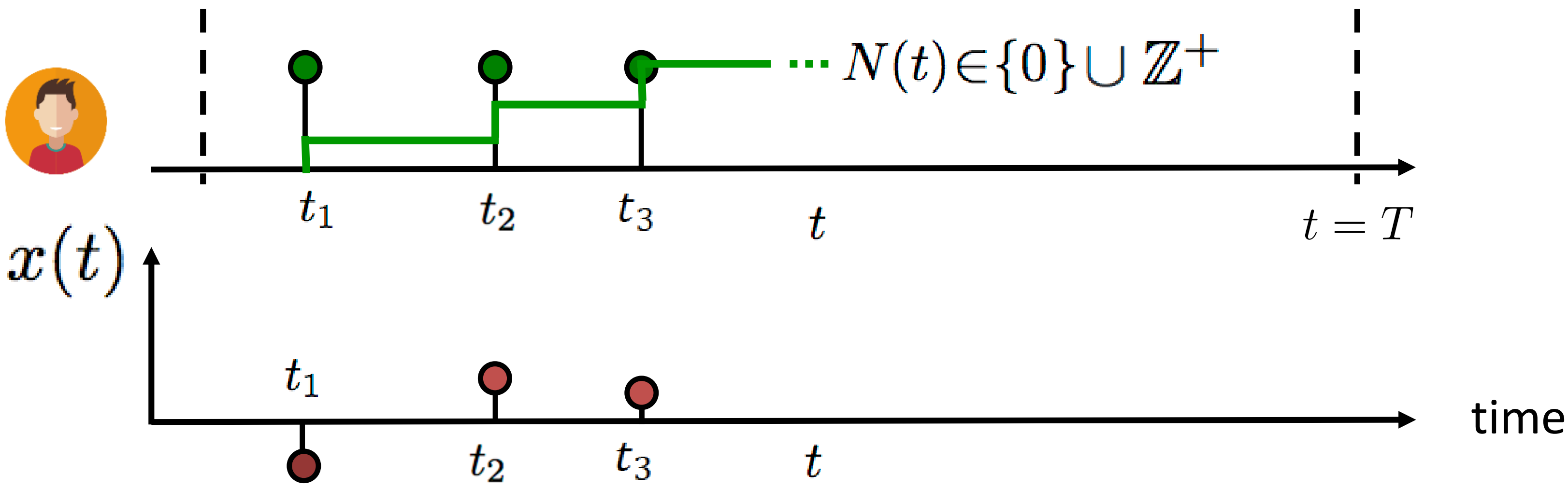
Marked temporal point process:

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



Skoltech

Independent identically distributed marks



Distribution for the marks:

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

Observations:

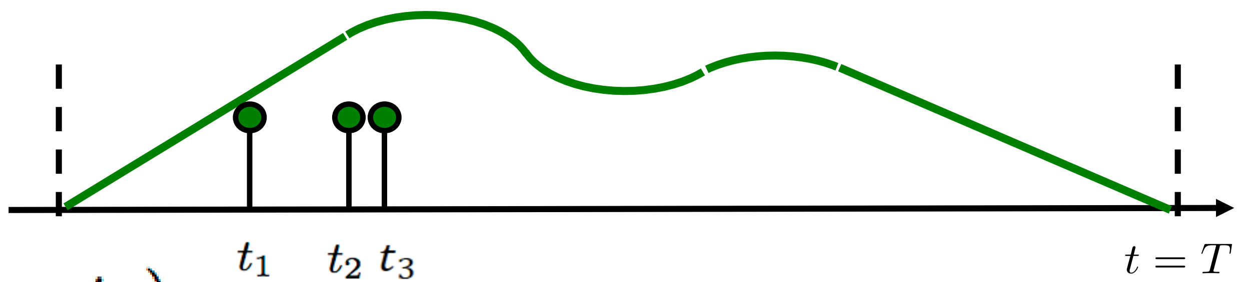
1. Marks independent of the temporal dynamics
2. Independent identically distributed (I.I.D.)

Models & Inference: Neural networks for the win

Neural networks for the win

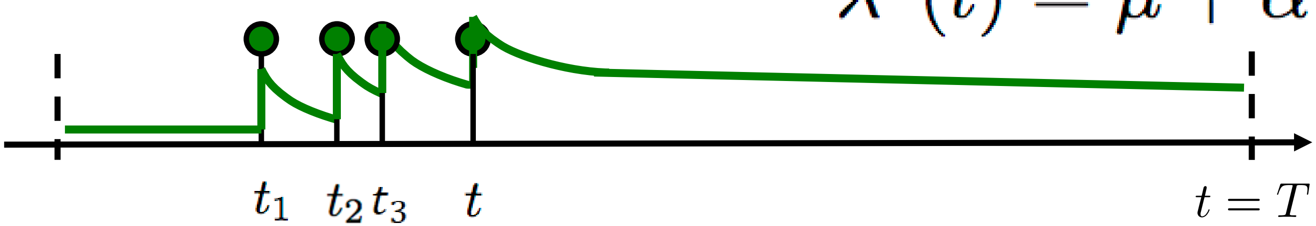
Up to now, we have focused on simple temporal dynamics (and intensity functions):

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



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

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



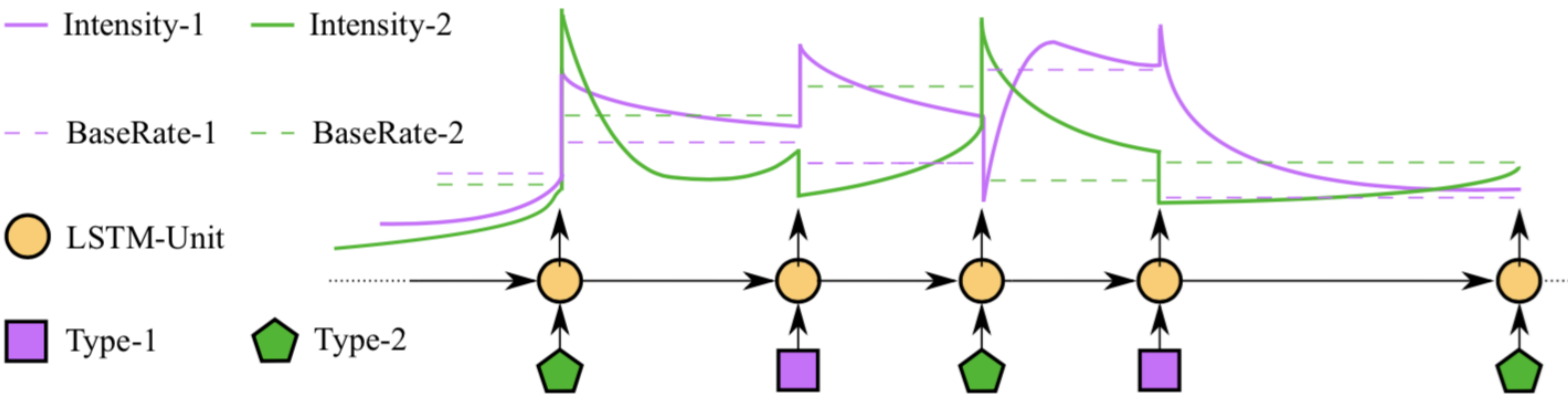
Recent works make use of RNNs to capture more complex dynamics

[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)

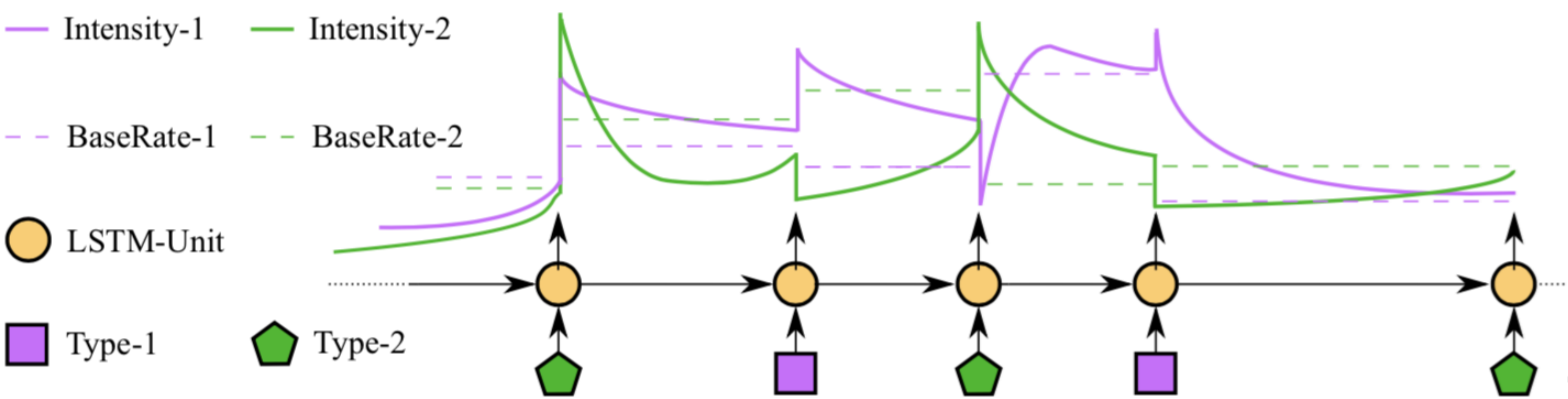


Mei, Hongyuan, and Jason M. Eisner. "The neural Hawkes process: A neurally self-modulating multivariate point process." *NIPS.* 2017.

Neural Hawkes process: RNNs

$$\mathbf{h}(t) = \mathrm{RNN}(\mathcal{H}(t))$$ memory via the continuous-time LSTM

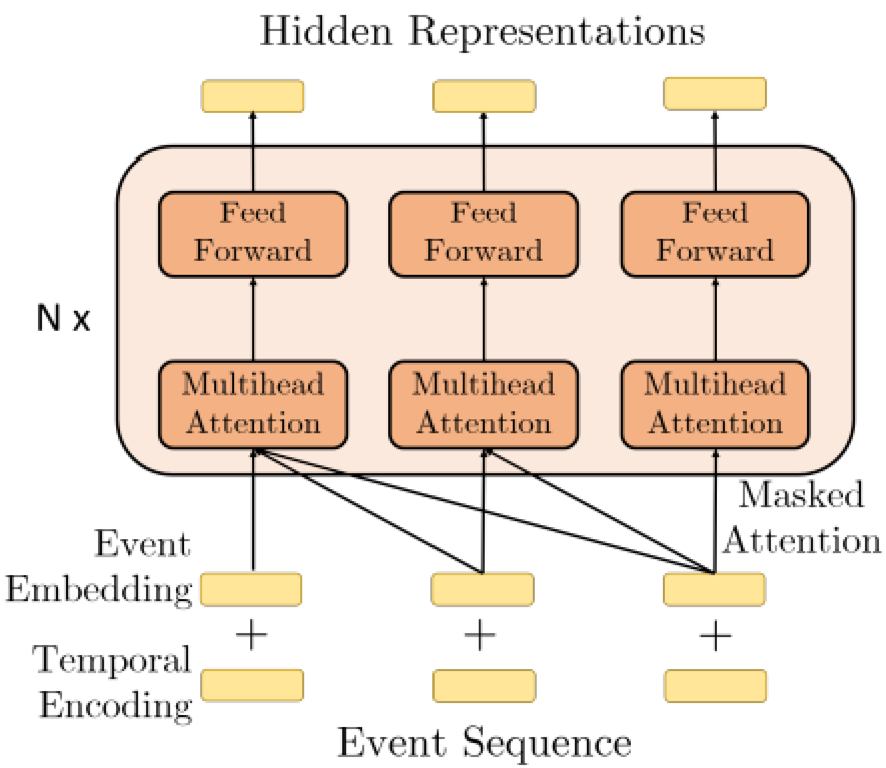
$$\lambda_u(t) = f_u(\mathbf{w}_u^\top \mathbf{h}(t))$$ excitation & inhibition via activation function



Mei, Hongyuan, and Jason M. Eisner. "The neural Hawkes process: A neurally self-modulating multivariate point process." *NIPS.* 2017.

Neural Hawkes process: Transformers

$$\lambda_k(t|\mathcal{H}_t) = f_k\Big(\underbrace{\alpha_k \frac{t - t_j}{t_j}}_{current} + \underbrace{\mathbf{w}_k^\top \mathbf{h}(t_j)}_{history} + \underbrace{b_k}_{base}\Big).$$



$$p(t|\mathcal{H}_t) = \lambda(t|\mathcal{H}_t) \exp\Big(-\int_{t_j}^{t} \lambda(\tau|\mathcal{H}_\tau) d\tau\Big),$$

$$\widehat{t}_{j+1} = \int_{t_j}^{\infty} t \cdot p(t|\mathcal{H}_t) dt,$$

$$\widehat{k}_{j+1} = \operatorname*{argmax}_k \frac{\lambda_k(t_{j+1}|\mathcal{H}_{j+1})}{\lambda(t_{j+1}|\mathcal{H}_{j+1})}.$$

Zuo, Simiao, et al. "Transformer Hawkes process." ICML. 2020.

Skoltech

References

References

- Lifestream library by Sber https://github.com/dllllb/pytorch-lifestream

- Github with exercises https://colab.research.google.com/drive/1Wc6aUNUZpE64egk4XwPT54v3sWDS495u?usp=sharing

- Competition to predict gender based on transactions https://www.kaggle.com/competitions/transactions/overview https://colab.research.google.com/drive/1tAqT4B5H9mkCiA3AilqUdIPOLpjEsXIK?usp=sharing

- ICML tutorial on temporal point processes https://learning.mpi-sws.org/tpp-icml18/

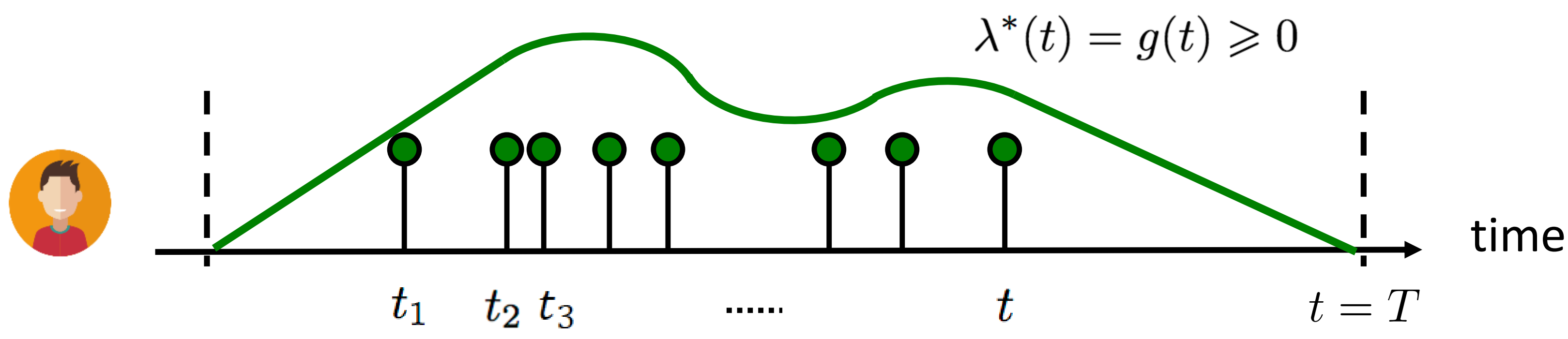
- Lecture Notes: Temporal Point Processes and the Conditional Intensity Function by J. Rasmussen https://arxiv.org/pdf/1806.00221.pdf

- Zuo, Simiao, et al. "Transformer Hawkes process." *ICML*, 2020.

Skoltech

Bonus II: Sampling

Fitting & sampling from inhomogeneous Poisson



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

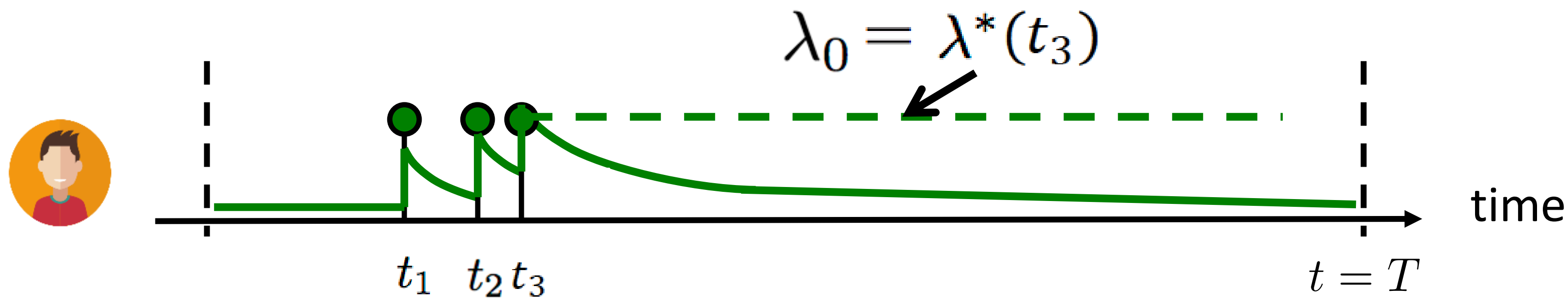
Sampling using thinning (reject. sampling) + inverse sampling*:

1. **Sample t from Poisson process with intensity μ 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. $g(t)/\mu$

Skoltech

Fitting a Hawkes process from a recorded timeline



Fitting by maximum likelihood:

$$\underset{\mu,\alpha}{\text{maximize}} \sum_{i=1}^{n} \log \lambda^*(t_i) - \int_0^T \lambda^*(\tau)\, d\tau$$

The max. likelihood is jointly convex in μ and α

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

Key idea: the maximum of the intensity λ_0 changes over time

Skoltech