Tuesday 17th of December, 2024

Time	Description	Discussant
[10 min]	(0) Introductions and administrivia	Trikalinos
[20 min]	(1) DES as a composition of point processes	Trikalinos
[25 min]	(2) NHPPPs – key properties	Trikalinos
[25 min]	(3) Sampling from NHPPPs	Trikalinos
[40 min]	(4) Guided code review - Implement a simple cancer natural history DES for one person - The many-person case - Packaging	Trikalinos
[15 min]	General Q & A	Trikalinos

Section 3: Sampling

Three important properties for sampling

Memorylessness

You can ignore what happens outside your interval

Composability

You can merge two NHPPPs with intensities λ_1, λ_2 to get a new NHPPP with intensity $\lambda_1 + \lambda_2$.

Transmutability (time warping)

Any one-to-one transformation of the intensity function results in a unique NHPPP in the transformed time axis

Overview of the sampling strategy

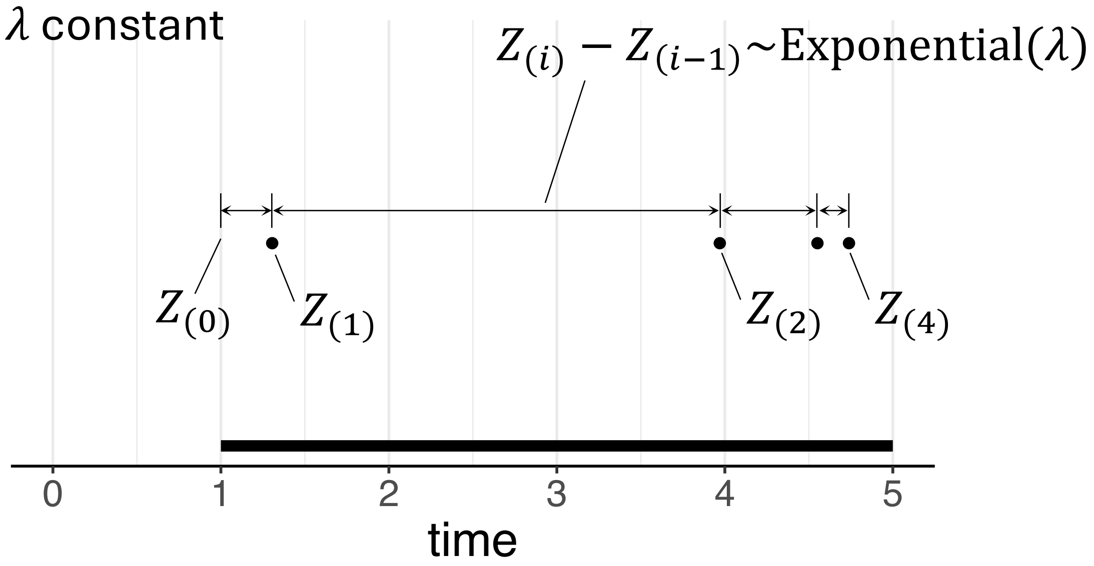
1. Sampling from constant rate PPP is easy	*easy*
2. Memorylessness implies you can treat the piecewise as constant PPPs over disjoint interval	*peasy*
3. Composability motivates an acceptance-rejection algorithm for sampling from any $\lambda(t)$	*almost always practical*
4. Time warping allows efficient sampling if you have (cheap access to) $\Lambda(\mathrm{t}), \Lambda^{-1}(t)$	*sometimes possible, may be worth the hassle to get Λ, Λ^{-1}*

1. Sampling from a PPP is easy

Constant intensity function (homogeneous PPP)

Sampling from a constant intensity function is easy.

The interarrival times have an exponential distribution.

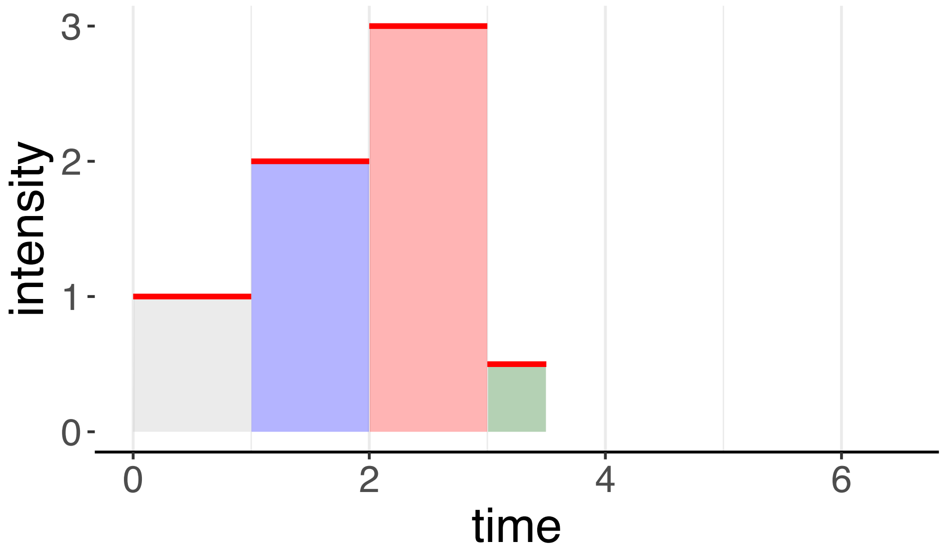


2. Memorylessness: Sampling from piecewise constant NHPPP is peasy

Piecewise constant intensity function (NHPPP)

- Look at each piecewise constant interval separately
- In each interval you have a constant intensity (easy)
- Return the union of all events

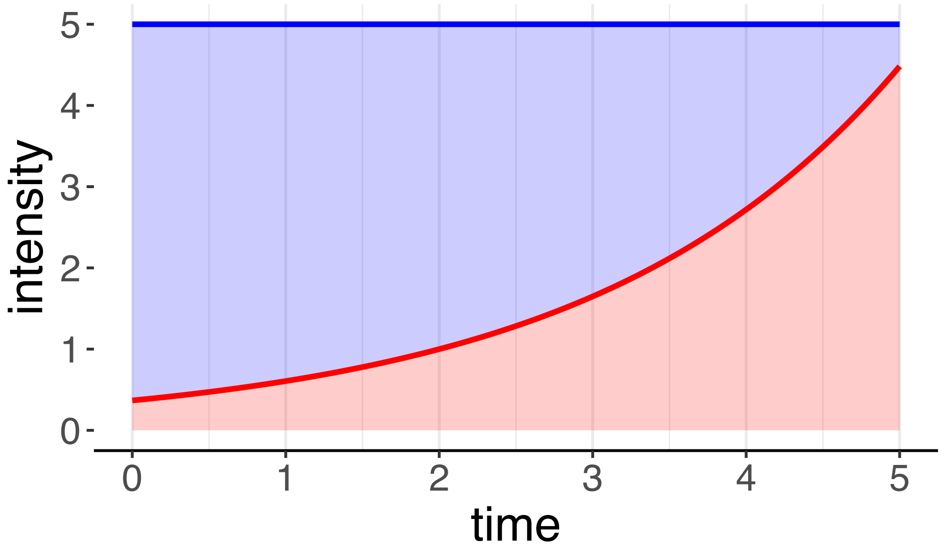
Sampling from piecewise constant intensities is easy (**memorylessness**)



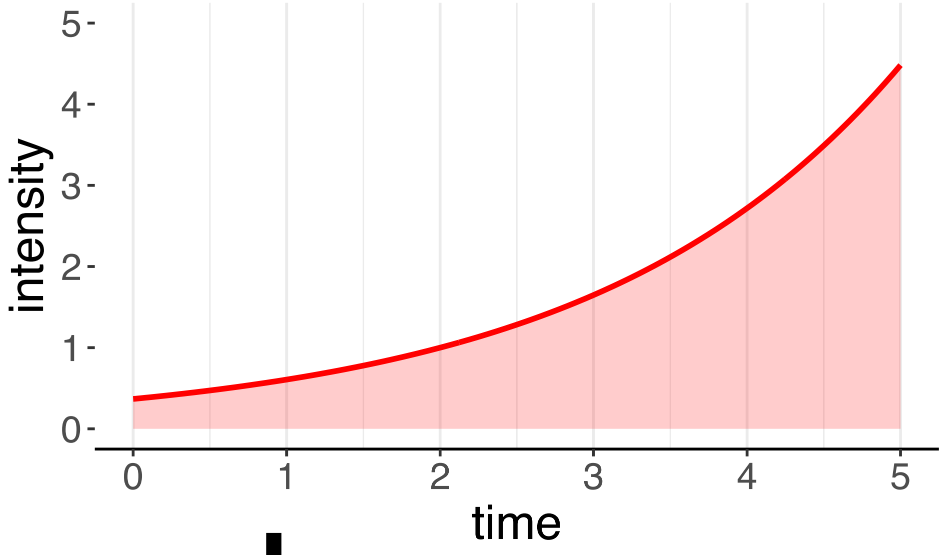
3. Composability: Sampling NHPPPs when you know $\lambda(t)$ reduces to sampling from a PPP (#1) or piecewise constant NHPPP (#2)*

** You still need to find a constant or piecewise constant majorizer $\lambda_*(t)$, whose choice determines your efficiency .*
You cannot get achieve something difficult with zero effort.
You will put in some work.
Other terms and conditions may apply.

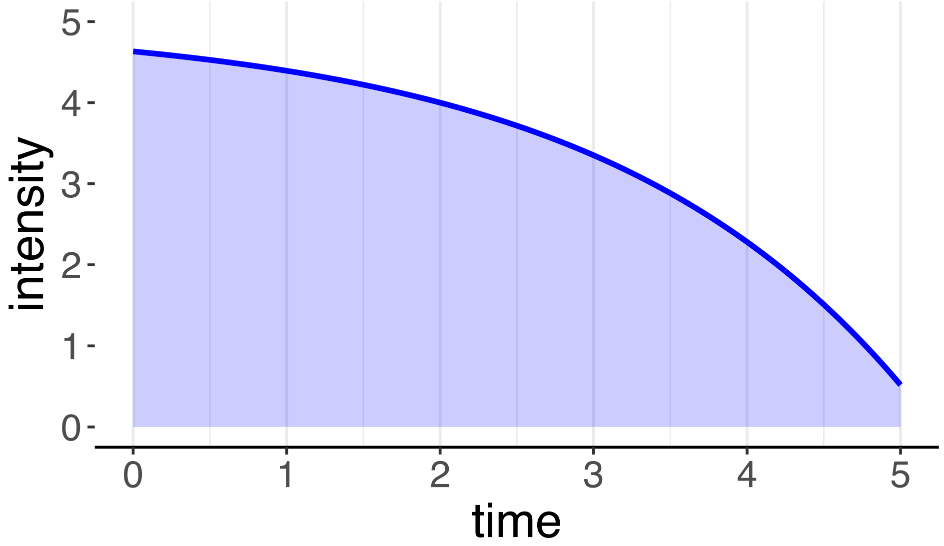
Composability



=

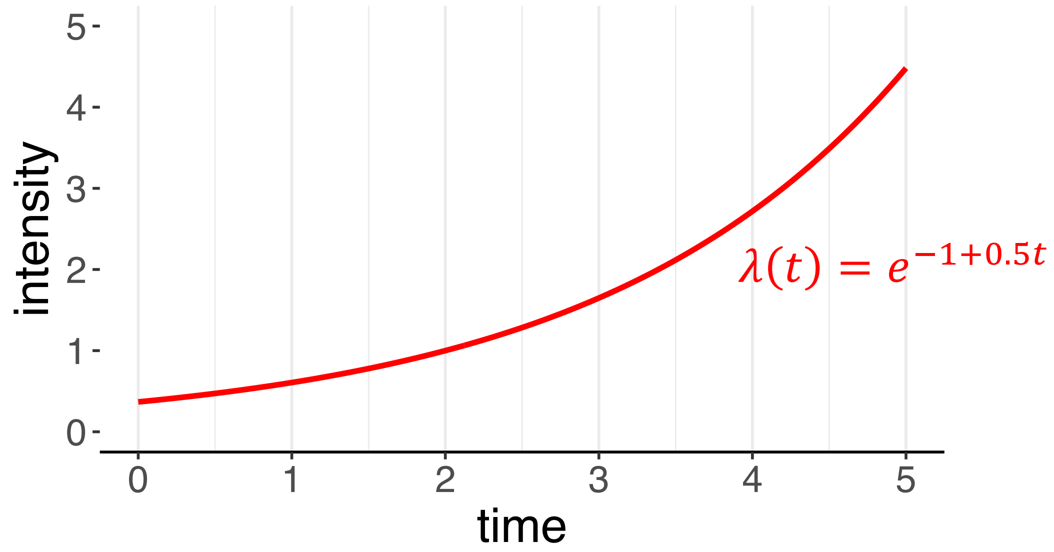


+



NHPPP, where you know $\lambda(t)$: Thinning

The general case is more challenging



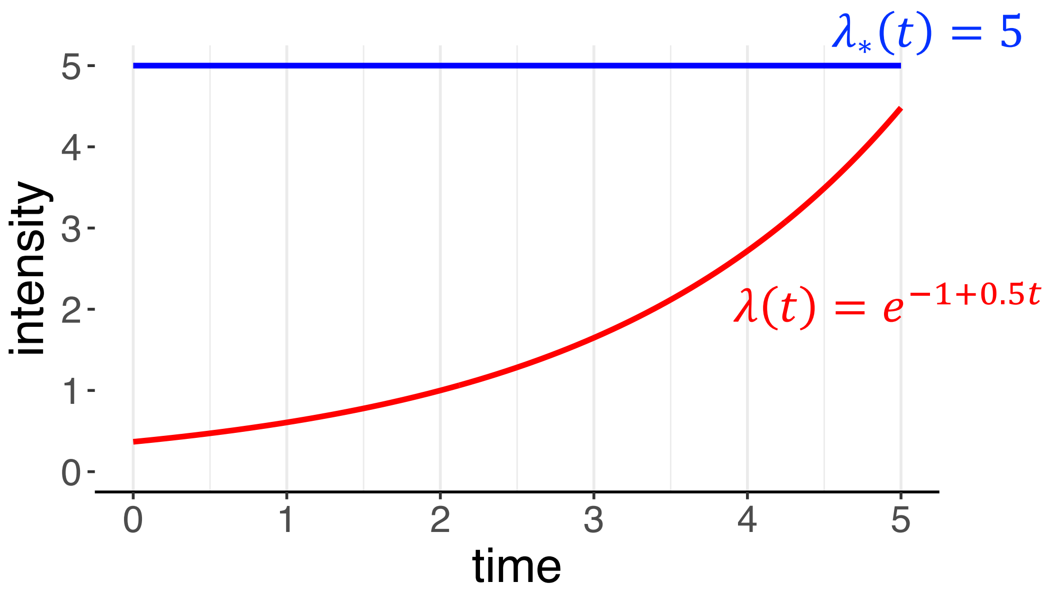
NHPPP, where you know $\lambda(t)$: Thinning

- Find a majorizer function λ_* that's easy to sample

Majorizer: any function that is "taller" that λ

$$\lambda_* \geq \lambda$$

(and has the same support as λ)



NHPPP, where you know $\lambda(t)$: Thinning

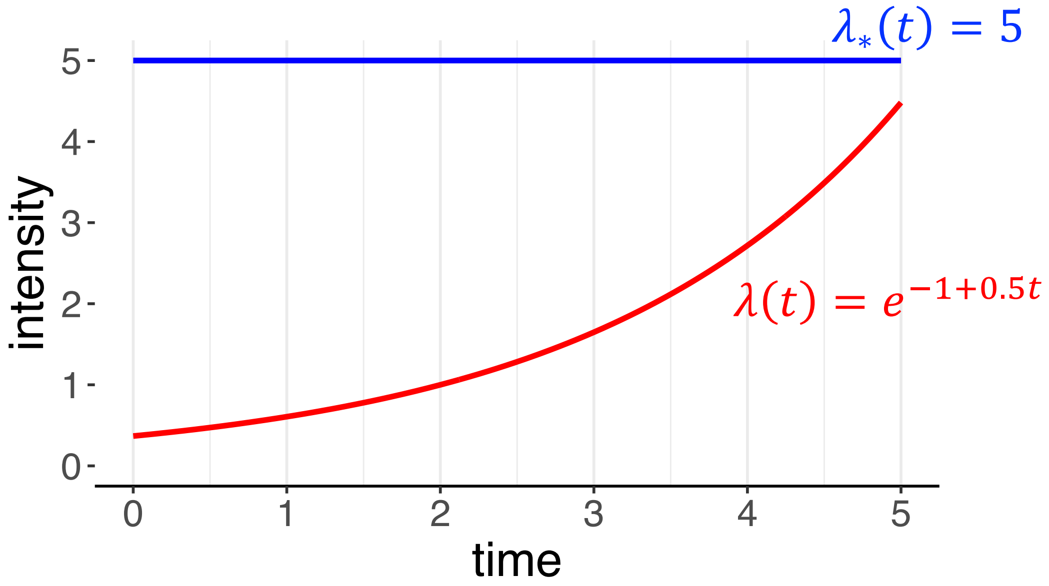
- Find a majorizer function λ_* that's easy to sample

$$\lambda_*(t) = \lambda(t) + [\lambda_*(t) - \lambda(t)]$$

Sample proposals from here

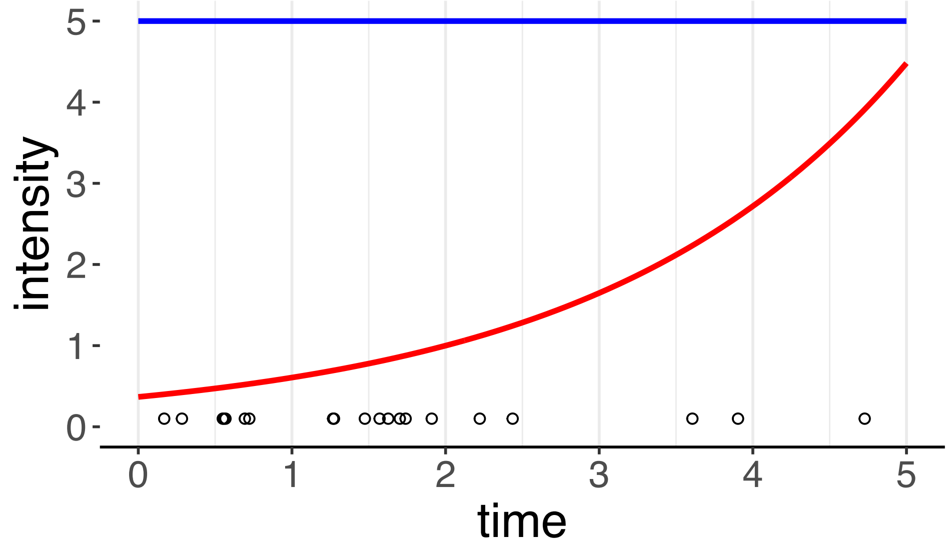
Keep proposals conforming to this part

Reject the rest



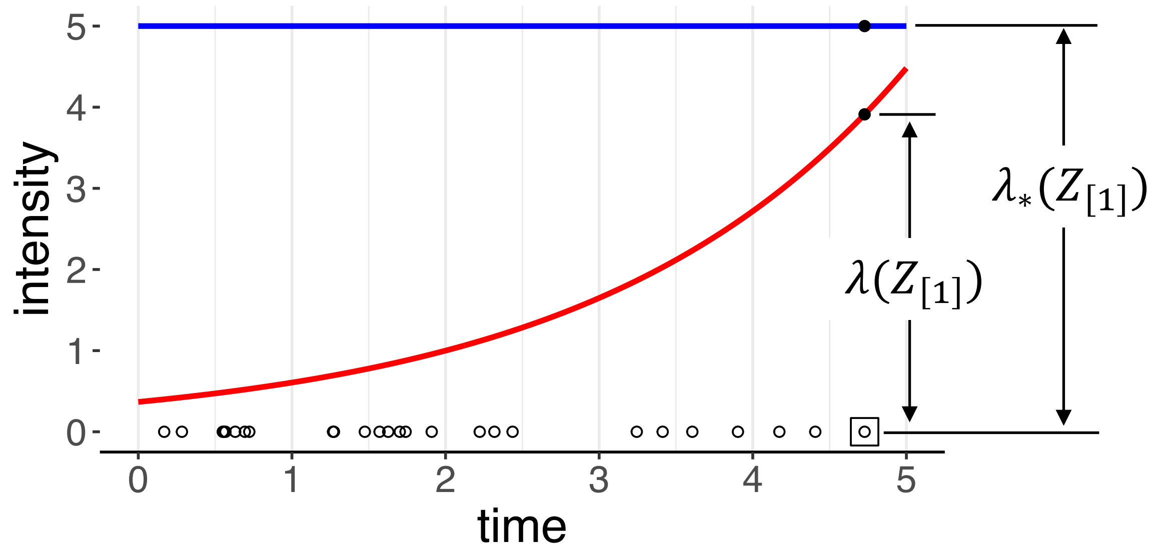
NHPPP, where you know $\lambda(t)$: Thinning

- Find a majorizer function λ_* that's easy to sample
- Draw events $\{Z_{*1}, \dots\}$ from λ_*



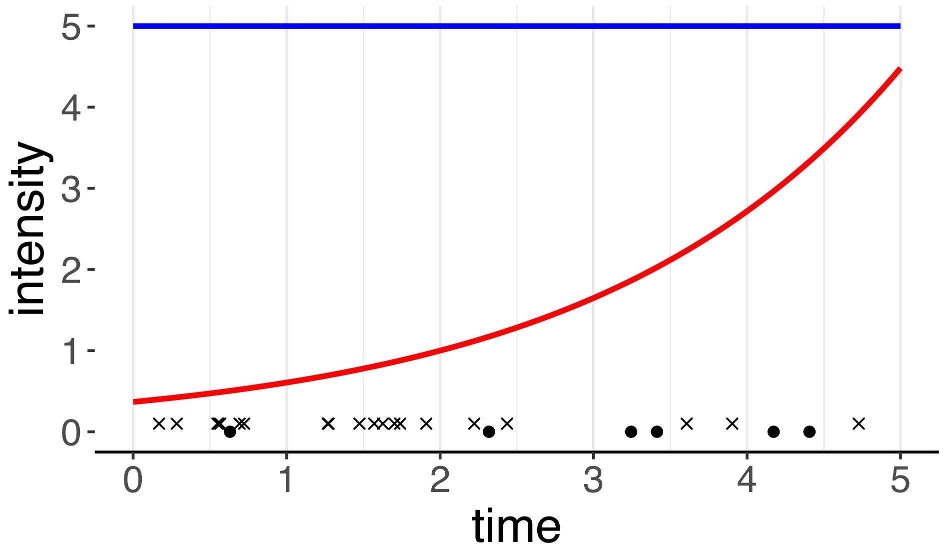
NHPPP, where you know $\lambda(t)$: Thinning

- Find a majorizer function λ_* that's easy to sample
- Draw events $\{Z_1, \ldots\}$ from λ_*
- Accept event i with probability $\frac{\lambda(Z_i)}{\lambda_*(Z_i)}$



NHPPP, where you know $\lambda(t)$: Thinning

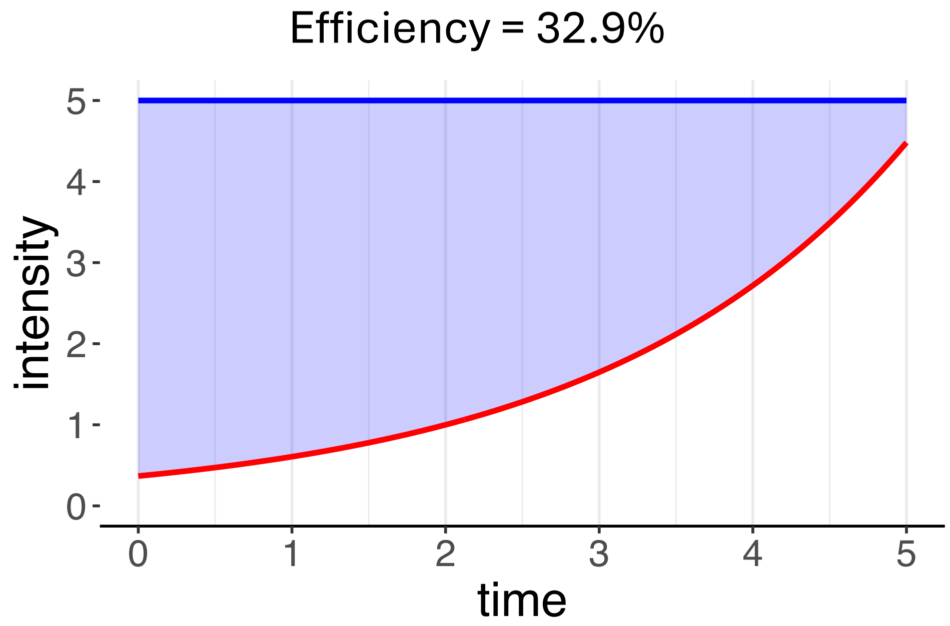
- Find a majorizer function λ_* that's easy to sample
- Draw events $\{Z_{*1}, \ldots\}$ from λ_*
- Accept event i with probability $\frac{\lambda(Z_i)}{\lambda_*(Z_i)}$
- The set of accepted points is an instantiation from $\lambda(t)$



(composability)

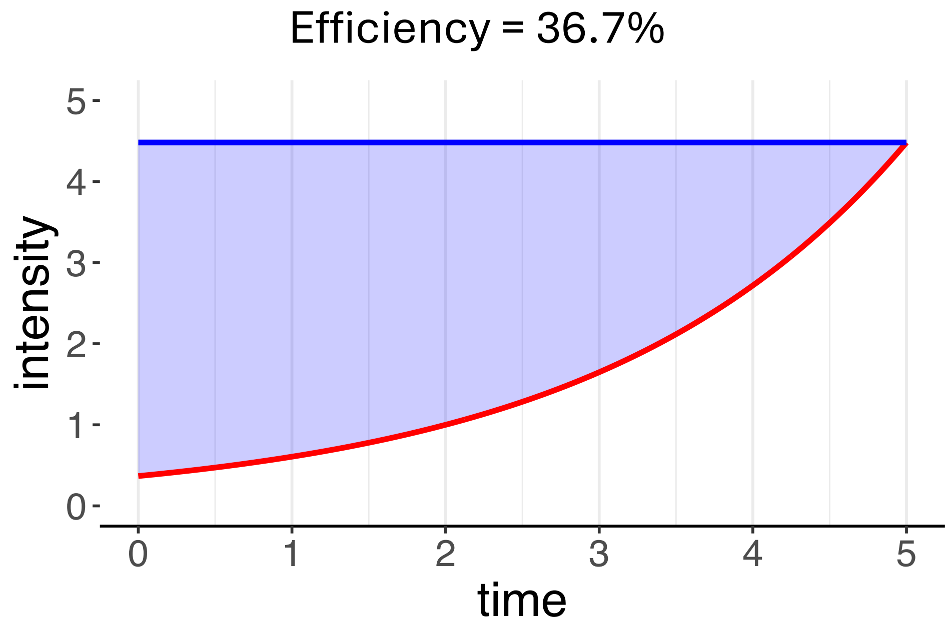
Thinning, efficiency

- Thinning efficiency: average fraction of proposals that are accepted
- Depends on the choice of λ_*
- The smaller the blue area, the better the efficiency



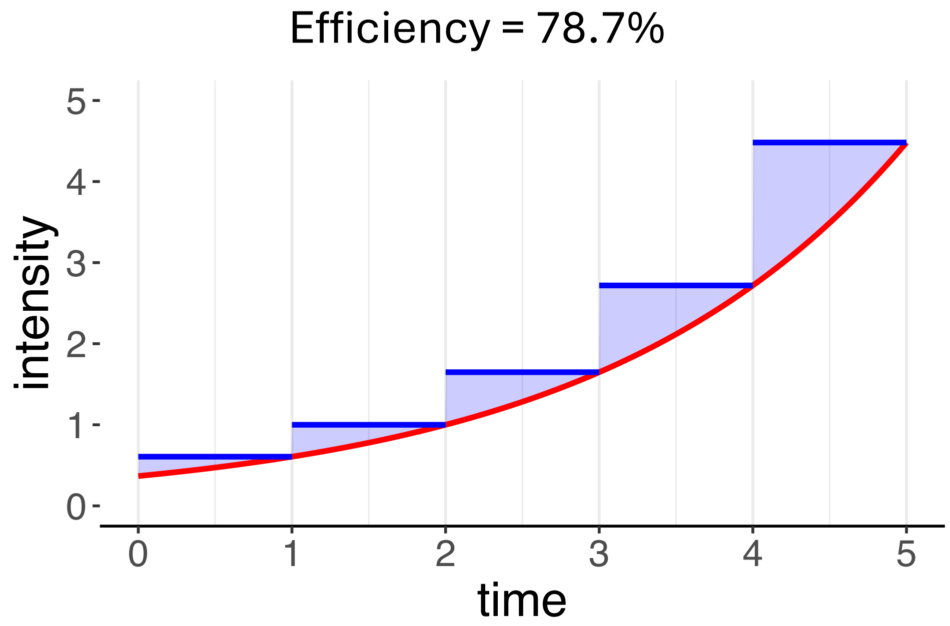
Thinning, efficiency

- Thinning efficiency: average fraction of proposals that are accepted
- Depends on the choice of λ_*
- The smaller the blue area, the better the efficiency



Thinning, efficiency

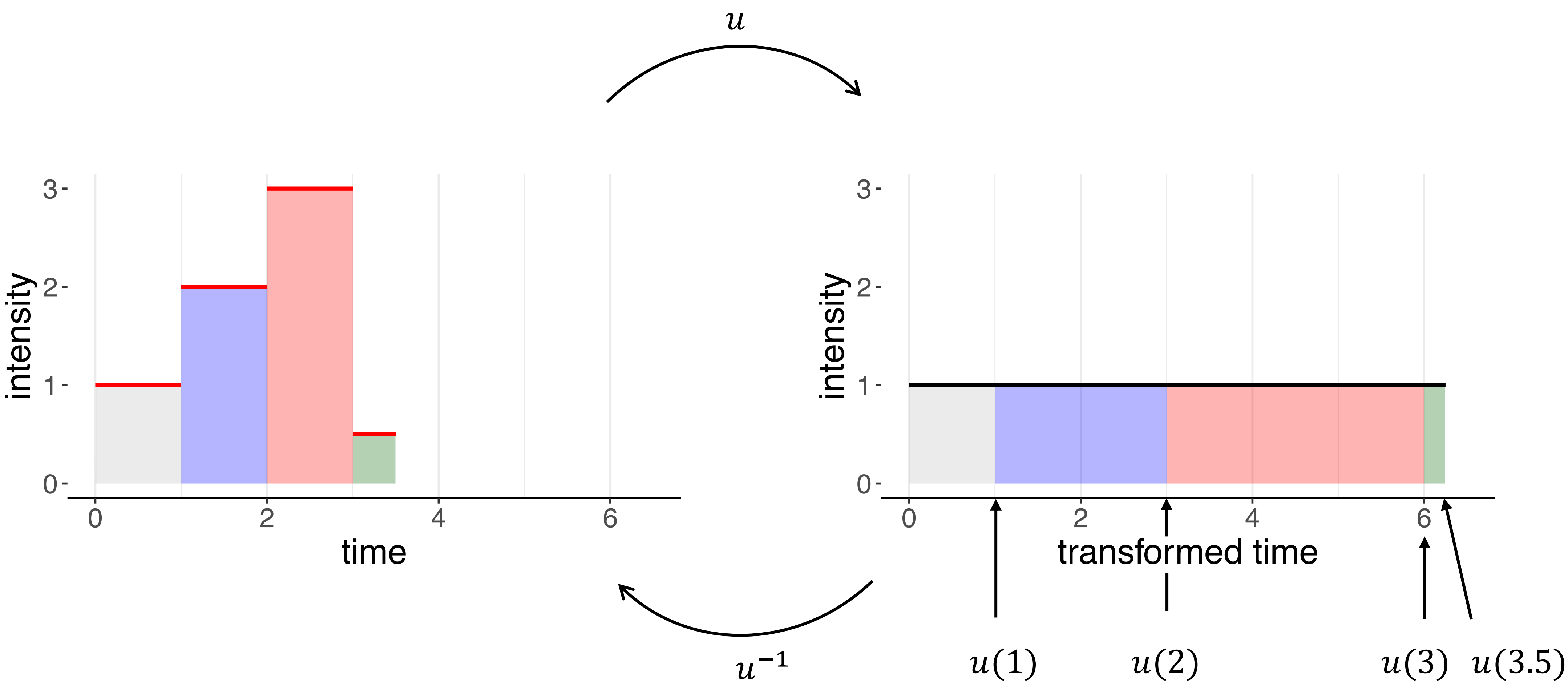
- Thinning efficiency: average fraction of proposals that are accepted
- Depends on the choice of λ_*
- The smaller the blue area, the better the efficiency



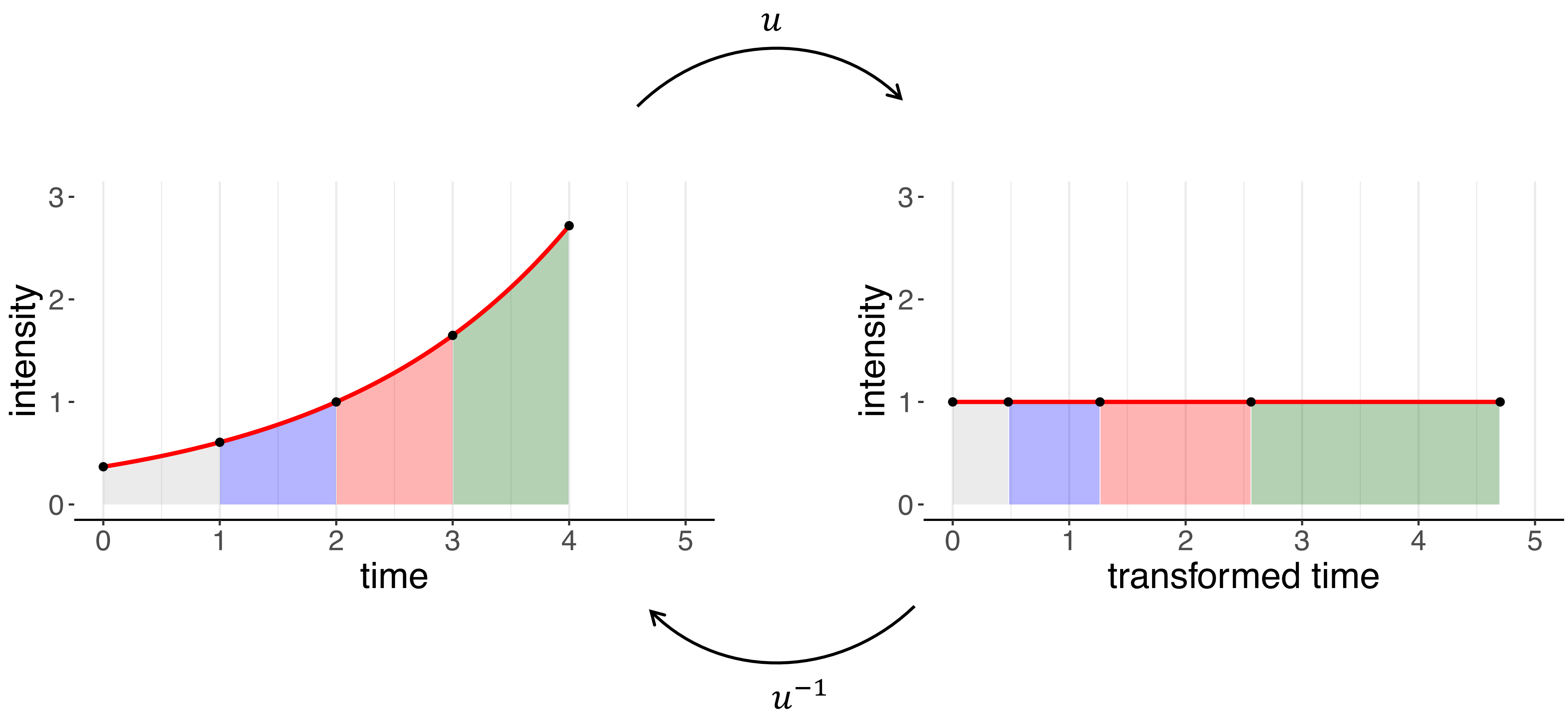
4. Transmutability of time: Sampling NHPPPs when you know Λ, Λ^{-1} reduces to sampling from a PPP with rate one (#1) *

** You will need to do some maths to get Λ, Λ^{-1}. It may not be practical to do so, or even possible. In such a case, back to (#3). Even if you have Λ, you may not have a cheap Λ^{-1}.*
You cannot achieve something difficult with zero effort. You will put in some work. Other terms and conditions may apply.

Transmutability



Transmutability



A nice u is Λ (and then u^{-1} is Λ^{-1})

Change of variable from s to u

$$\Lambda(t) = \int_a^t \lambda(s)\,\mathrm{d}s = \int_{u(a)}^{u(t)} \frac{\lambda(s)}{u'(s)}\,\mathrm{d}u$$

Pick u so that $u' = \lambda$. Any antiderivative of λ works. Using $u \coloneqq \Lambda$, transforms time to scale where the process has constant rate 1,

$$\int_{\Lambda(a)}^{\Lambda(t)} \frac{\lambda(s)}{\Lambda'(s)}\,\mathrm{d}u = \int_{\Lambda(a)}^{\Lambda(t)} 1\,\mathrm{d}u\,.$$

This is a sketch of the formal proof – omitting the rigorous bits

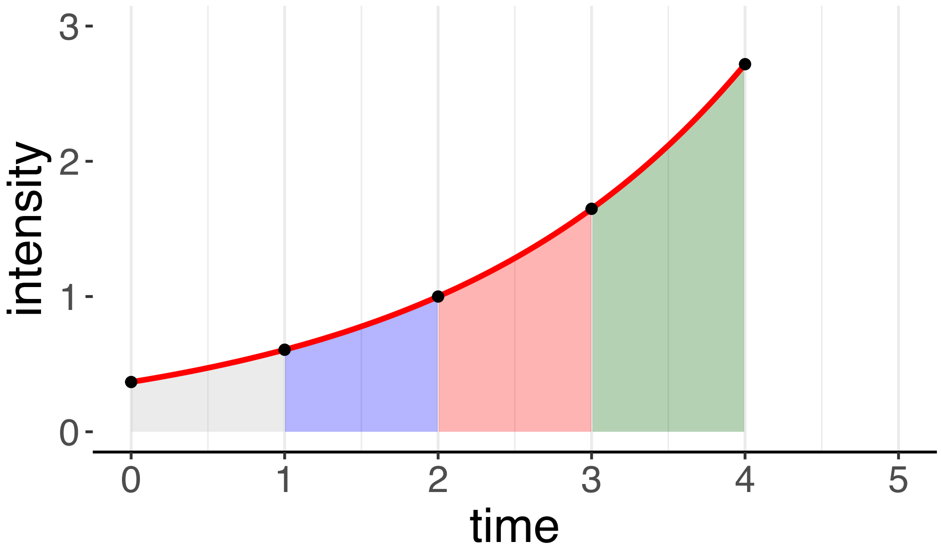
Transmutability

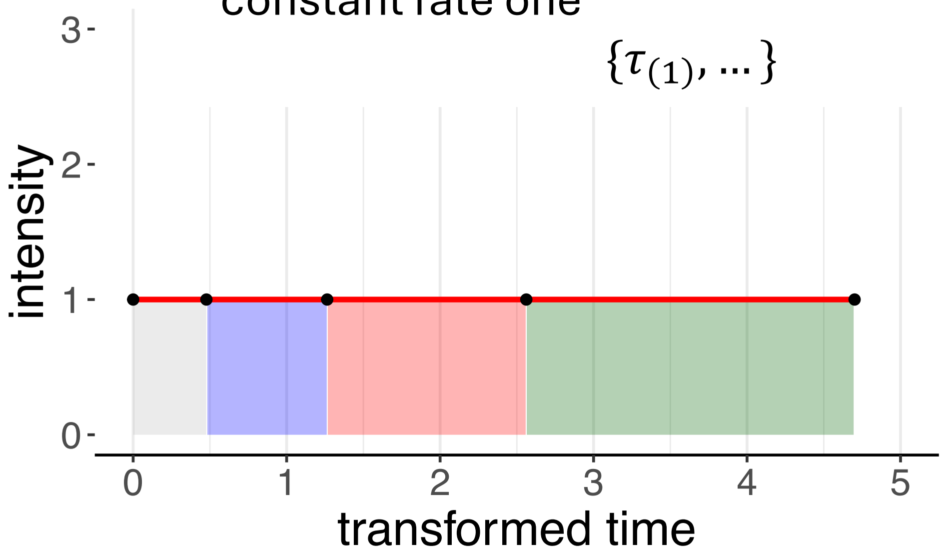
1. Find the start and stop of the transformed time interval
$\tau_{start} = \Lambda(t_{start})$ and $\tau_{stop} = \Lambda(t_{stop})$

Λ

2. Sample transformed times from a PPP with constant rate one

$\{\tau_{(1)}, \dots\}$





Λ^{-1}

3. Back-transform the instantiation to the original time scale

$\{\Lambda^{-1}(\tau_{(1)}), \dots\}$

More in these works...

arXiv > stat > arXiv:2402.00358

Search...

Help | Advar

Statistics > Computation

[Submitted on 1 Feb 2024 (v1), last revised 29 May 2024 (this version, v2)]

nhppp: Simulating Nonhomogeneous Poisson Point Processes in R

Thomas A. Trikalinos, Yuliia Sereda

PLOS ONE

RESEARCH ARTICLE

The nhppp package for simulating non-homogeneous Poisson point processes in R

Thomas A. Trikalinos[1,2,3]*, Yuliia Sereda[1]

medRxiv

THE PREPRINT SERVER FOR HEALTH SCIENCES

CSH Cold Spring Harbor Laboratory BMJ Yale

Follow this p

A Fast Nonparametric Sampling (NPS) Method for Time-to-Event in Individual-Level Simulation Models

David U. Garibay-Treviño, Hawre Jalal, Fernando Alarid-Escudero

doi: https://doi.org/10.1101/2024.04.05.24305356

medRxiv

THE PREPRINT SERVER FOR HEALTH SCIENCES

Next … Section 4: Hands-on example (simple case)

Tuesday 17th of December, 2024

Time	Description	Discussant
[10 min]	(0) Introductions and administrivia	Trikalinos
[20 min]	(1) DES as a composition of point processes	Trikalinos
[25 min]	(2) NHPPPs – key properties	Trikalinos
[25 min]	(3) Sampling from NHPPPs	Trikalinos
[40 min]	(4) Guided code review - Implement a simple cancer natural history DES for one person - The many-person case - Packaging	Trikalinos
[15 min]	General Q & A	Trikalinos