Analysis of RNA dynamics

using

Alexey Uvarovskii, Christoph Dieterich

University Hospital Heidelberg

Acknowledgement

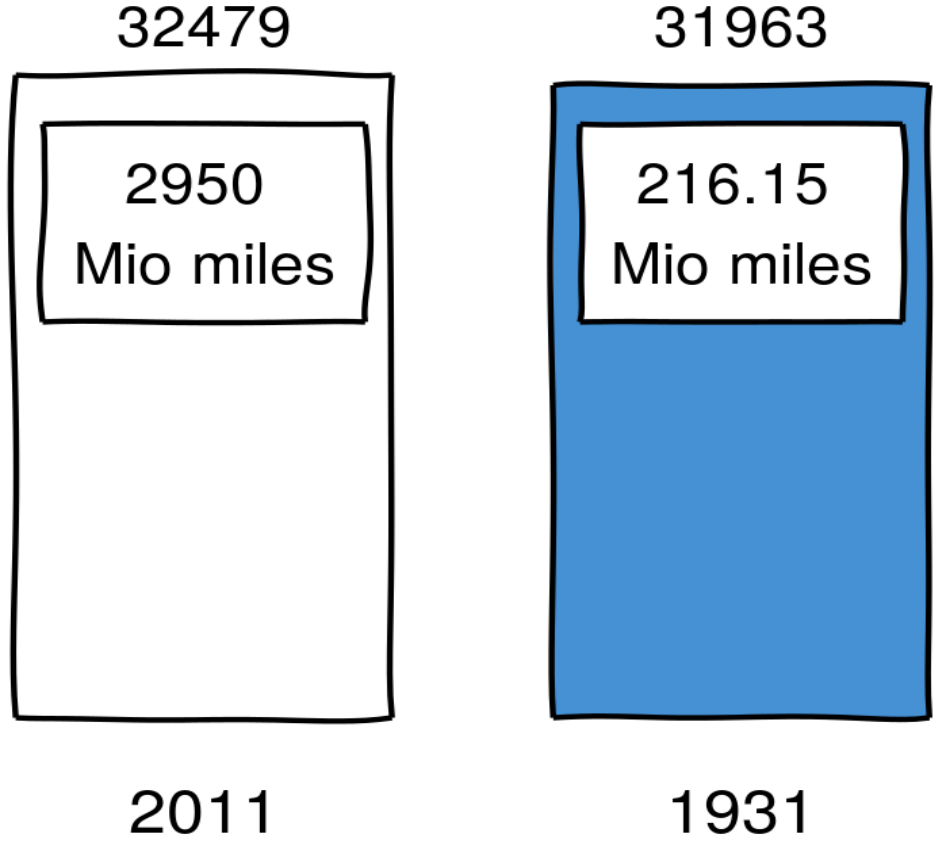
Heidelberg University Hospital

Federal Ministry
of Education
and Research

DZHK
DEUTSCHES ZENTRUM FÜR
HERZ-KREISLAUF-FORSCHUNG E.V.

Do you see a difference?

Number of motor vehicle deaths in the US (wiki)



Same number, different context

I bet you think about

Rates

RNA level is a balance

$$\frac{d[RNA]}{dt} = +[synthesis] - [degradation] \cdot [RNA]$$

$$[steady\ state\ RNA] = \frac{[synthesis]}{[degradation\ ]}$$

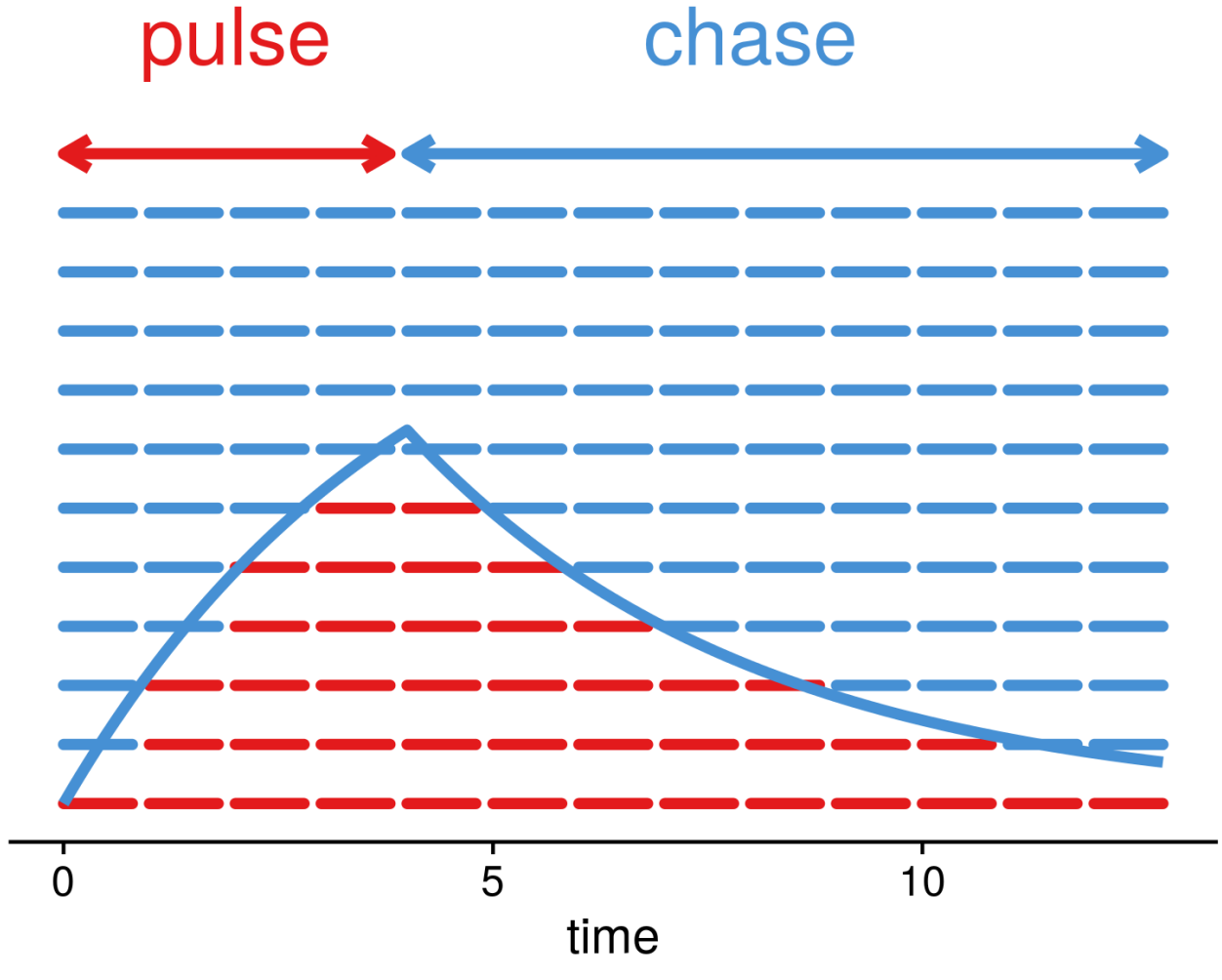
Nascent RNA can be traced

- eU
- 4sU

usual RNA-seq after then

Pulse-chase experiment

a way to measure RNA kinetics



Background

- rates are also interesting
- new RNA can be traced
- pulse-chase RNA-seq

Analysis

- kinetic model
- stat model
- normalisation

`pulseR` to help

Math model

defined by the setup, e.g. pulse labelling is

$$[\text{total}] = T \equiv \text{const}$$

$$[\text{pull down}] = T \cdot \left(1 - e^{-dt}\right)$$

$$\text{t} = 0, 1, 2, 4\,\text{hr}$$

Stat model

Count model for count data:

Negative binomial distribution

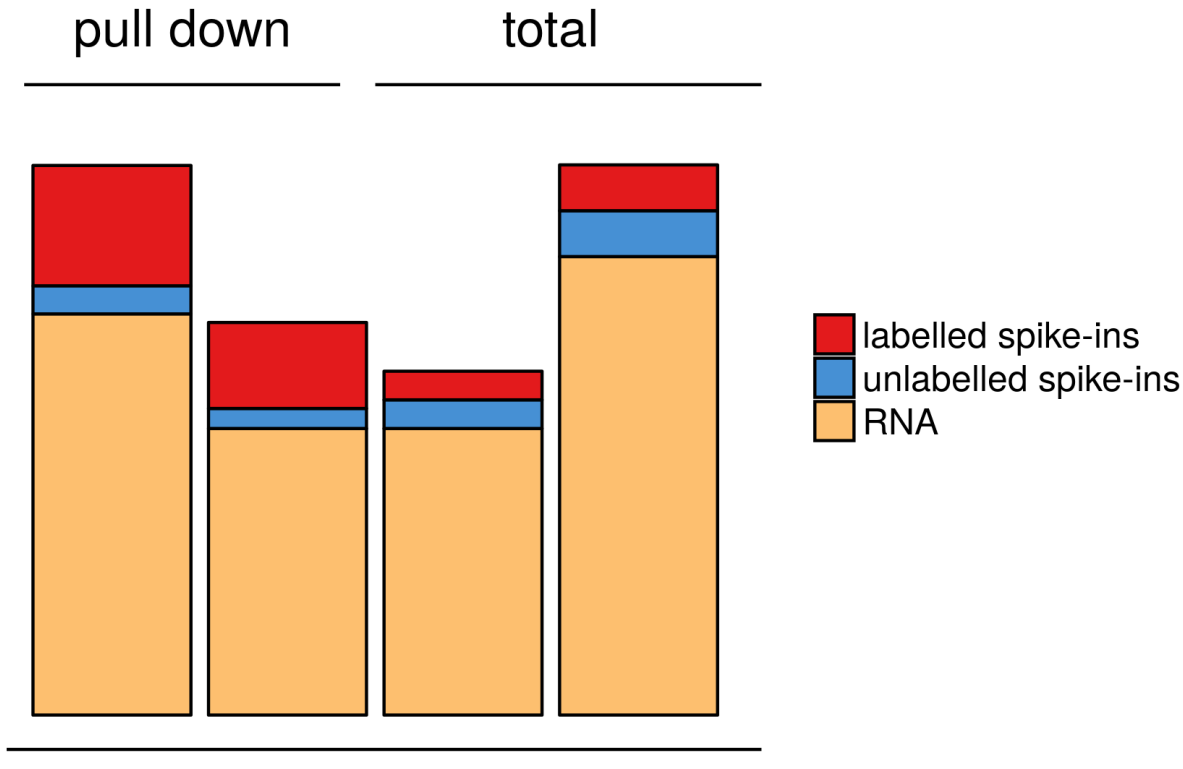
Fitting by maximum likelihood estimation (MLE)

Normalisation

$$[\text{total}] = [\text{labelled}] + [\text{unlabelled}]$$

$$[\text{pull down}] = ?[\text{labelled}] + ?[\text{unlabelled}]$$

Normalisation using spike-ins



In `pulseR`

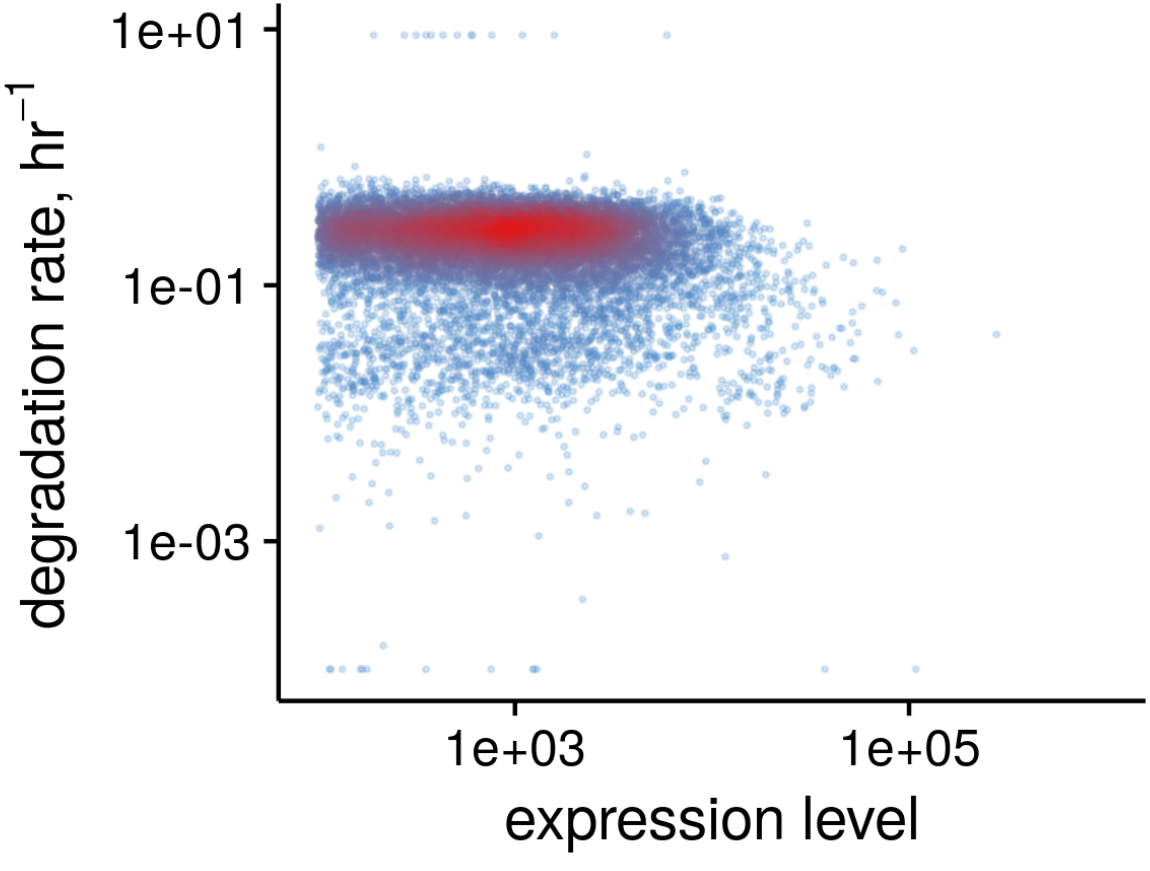
there are two options for normalisation:

using spike-ins (DESeq) absolute synthesis rate

by MLE fitting no spike-ins needed

EU pulse-chase on H9 cells

pulse (22hr) — chase (0, 4, 8hr)

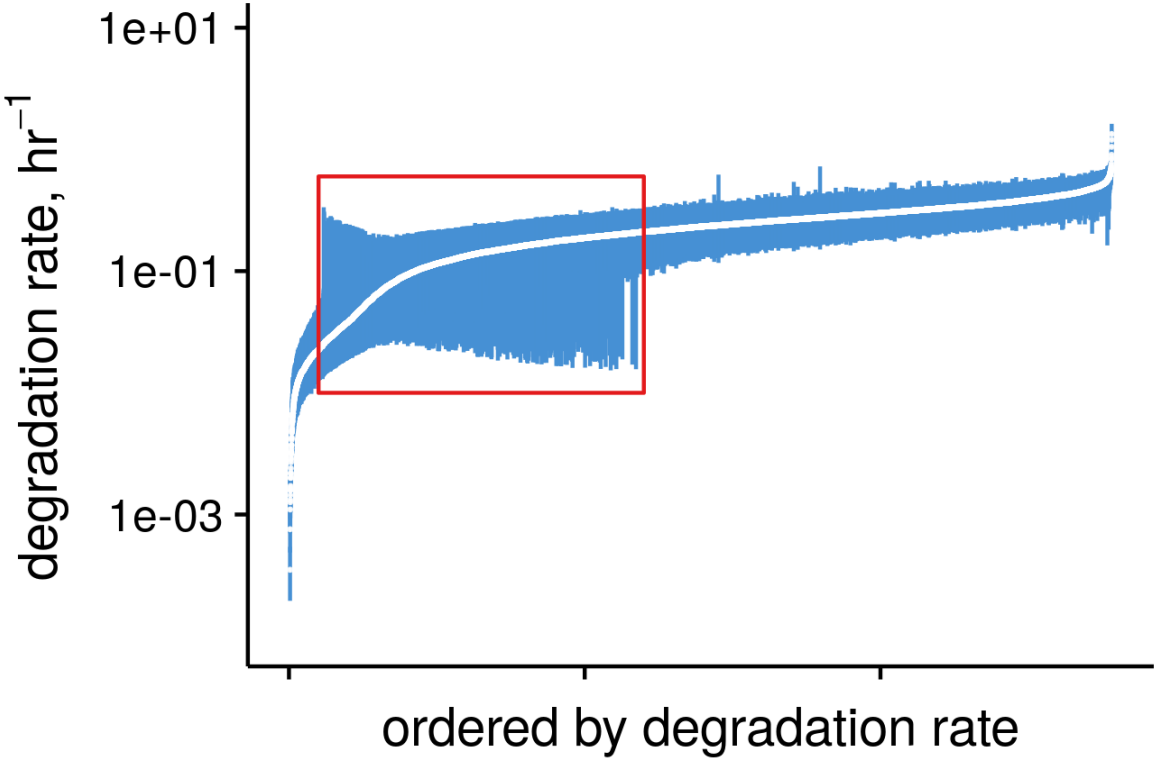


A number is not enough

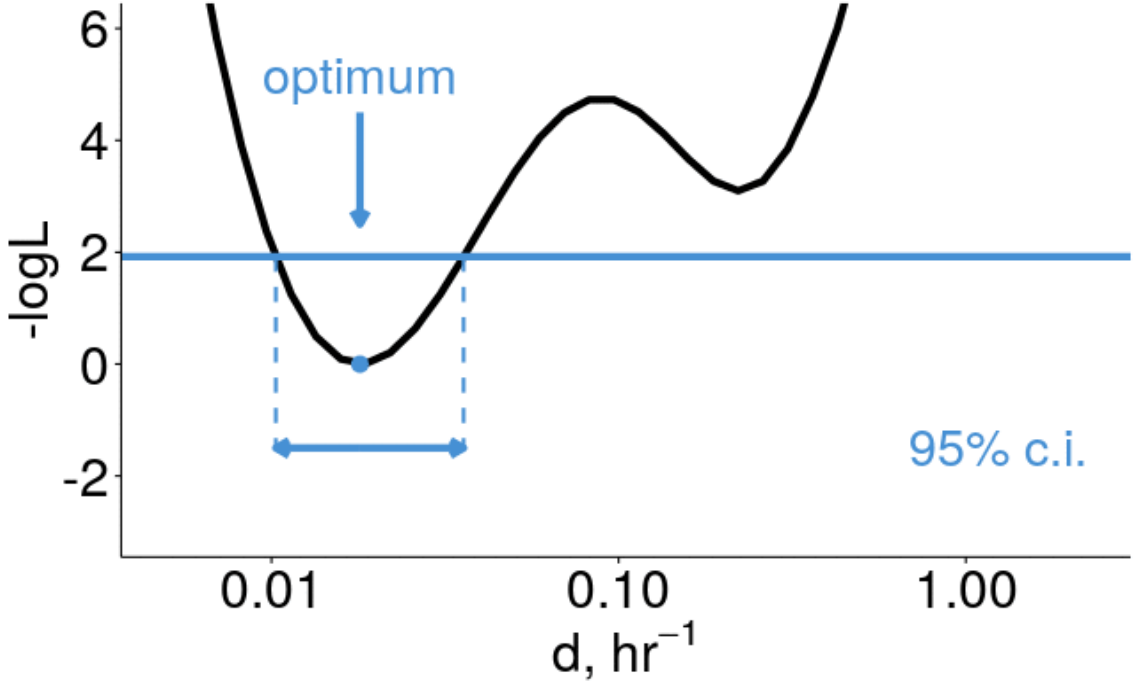
`pulseR` can estimate confidence intervals for you

Good for: diagnostics and comparisons

Confidence intervals



Uncertainty in fit



The workflow

```
library(pulseR)
# put math here
formulas <- MeanFormulas(
  total = mu,
  labelled = mu * (1 - exp(-d*22)) * exp(-d*time),
  unlabelled = mu * (1 - exp(-d*time) * (1 - exp(-d * 22)))
)
# define the fractions
formulaIndexes <- list(
  total_fraction = 'total',
  pull_down      = c('labelled', 'unlabelled'))
```

```
pd <- PulseData(counts, conditions, formulas, formulaIndexes,
  groups = ~ fraction + time)
result <- fitModel(pd, ipar, opts)
```

`pulseR` allows to

- estimate kinetic rates from RNA-seq
- flexible analysis (spike-ins, cross-contamination, etc.)
- diagnostics with profile likelihood

dieterichlab.org

github.com/dieterich-lab/pulseR

alexey.mipt@gmail.com