

Fitting mathematical models to time series of infectious disease cases

Sebastian Funk (@sbfnk)

Centre for the Mathematical Modelling of Infectious Diseases
London School of Hygiene & Tropical Medicine



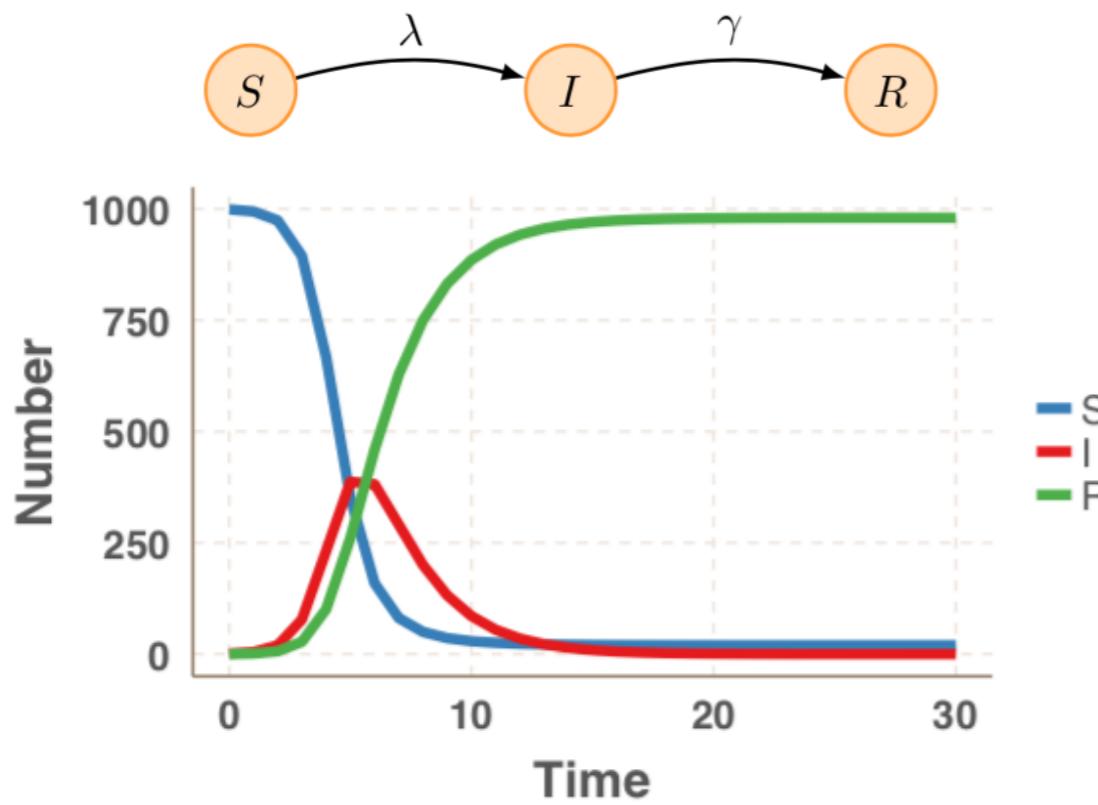
*centre for the
mathematical
modelling of
infectious diseases*

LONDON
SCHOOL *of*
HYGIENE
& TROPICAL
MEDICINE



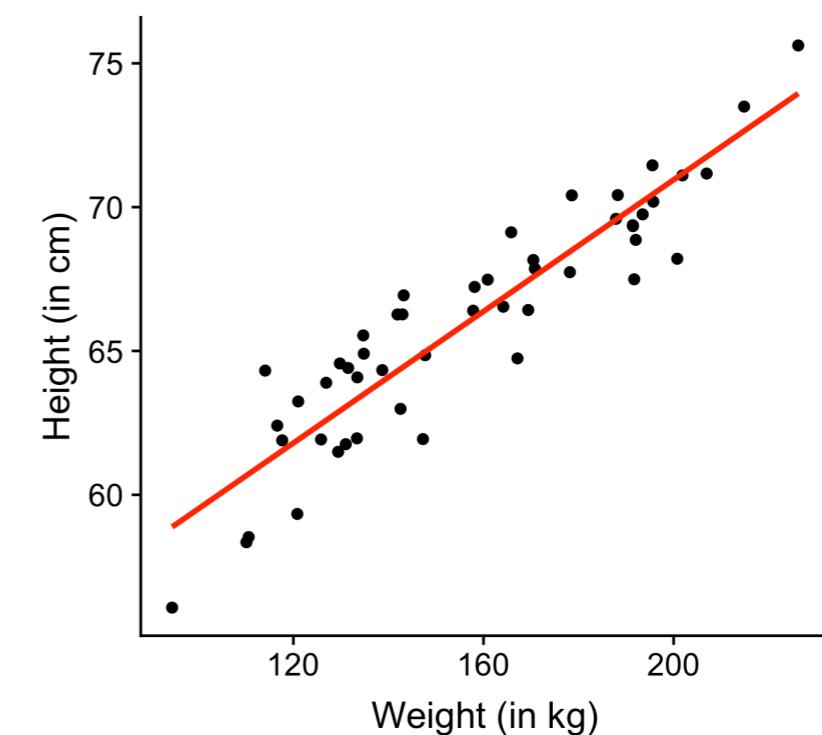
Mechanistic approach

- Design model from first principles
- Focus on model behaviour under different scenarios / parameter ranges

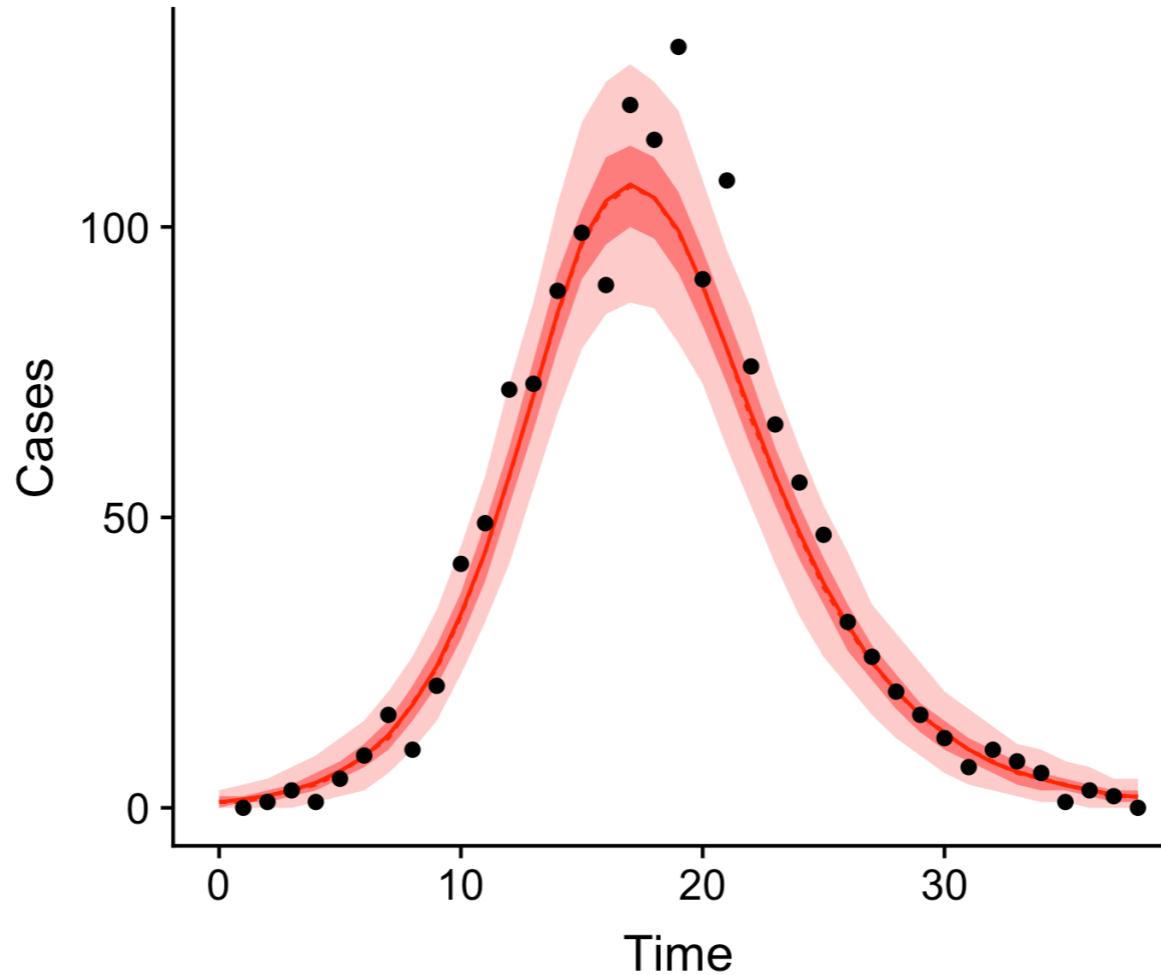


Statistical approach

- Models as a tool to investigate data
- Model choice driven by data and hypotheses about relationships between variables



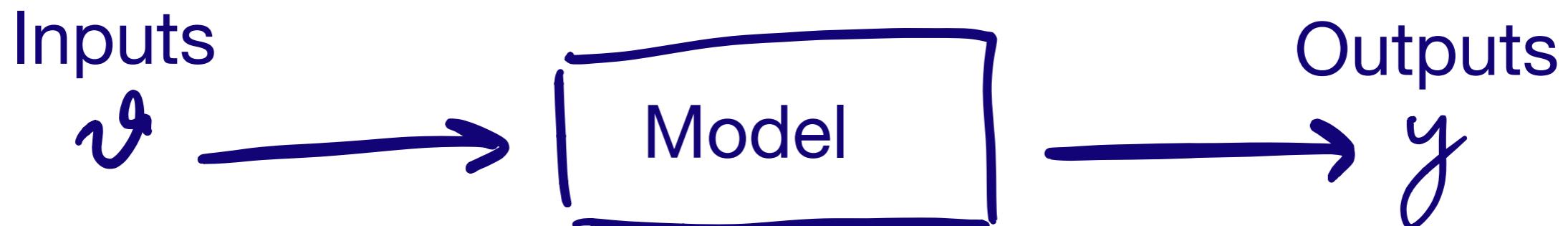
Fitting mathematical models



Combine mechanistic and statistical approach for

- Parameter estimation
- Prediction

A model takes input values θ (parameters) and returns output values y (observations, data).

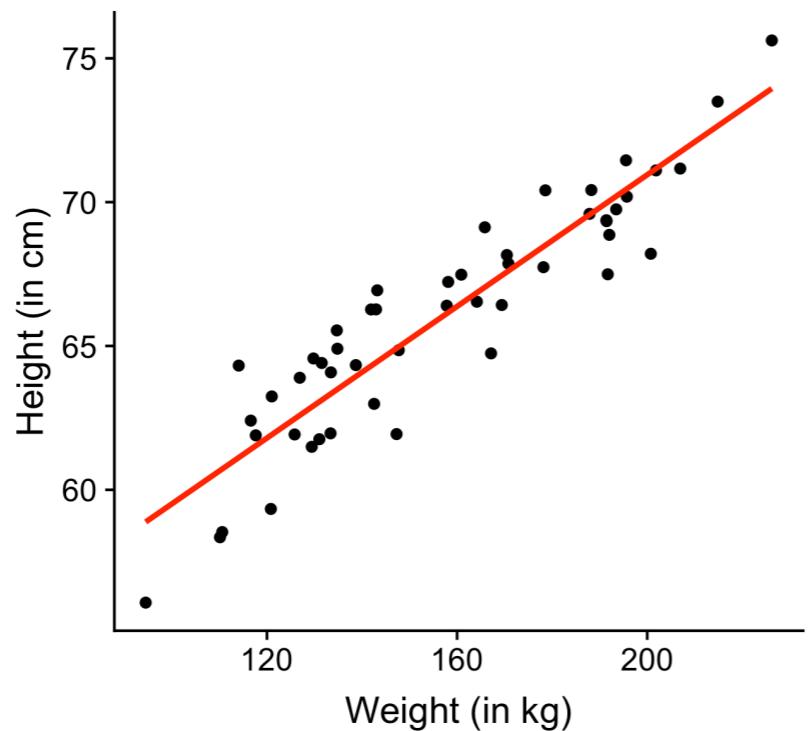


probability $p(y|\theta)$ that y is the output, given inputs θ .

Linear model

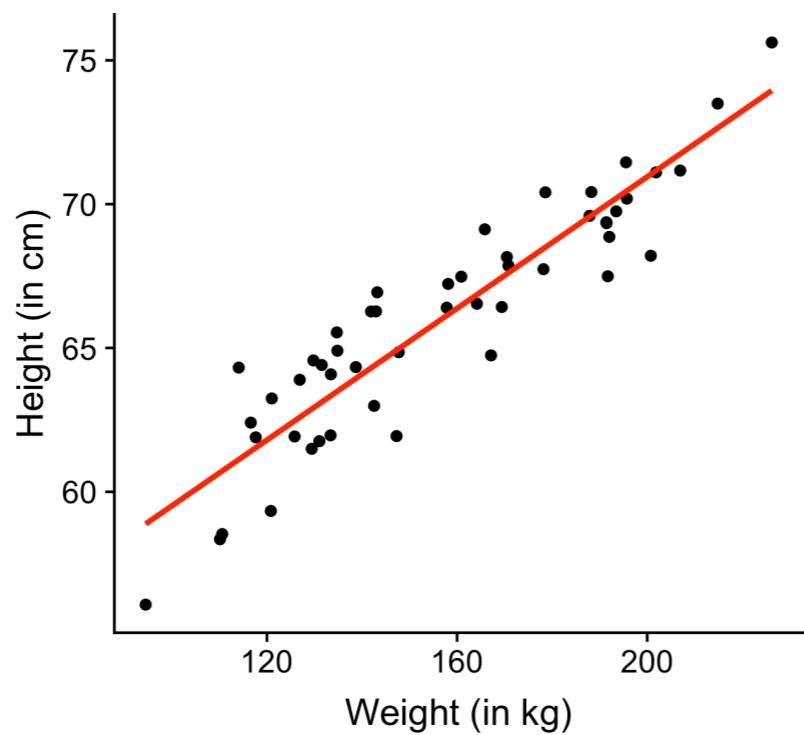
$$y = \beta_0 + \beta_1 x + \epsilon$$

$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$



Linear model

$$y \sim \mathcal{N}(\beta_0 + \beta_1 x, \sigma^2)$$

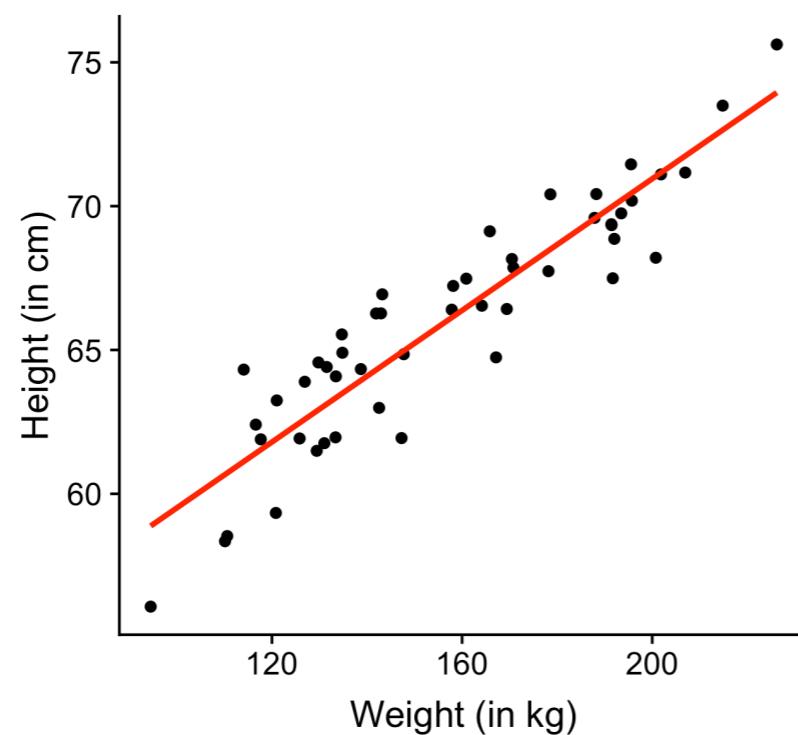


Normal
probability density

Linear model



$$p(y|\theta = \{\beta_0, \beta_1, \sigma\}) = f(y|\beta_0 + \beta_1 x, \sigma^2)$$



Normal
probability density

Linear model



$$p(y|\theta = \{\beta_0, \beta_1, \sigma\}) = f(y|\beta_0 + \beta_1 x, \sigma^2)$$

Given n data points (x_i, y_i) , $i=1..n$

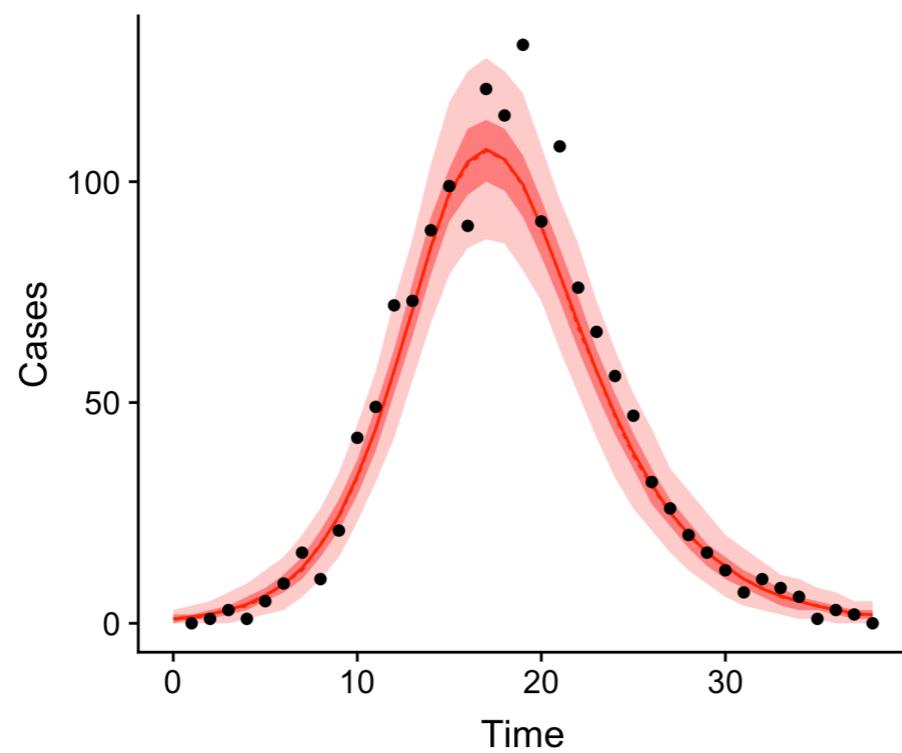
$$\begin{aligned} p(y_i|\theta) &= \prod_{i=1}^n p((y_i, x_i) | \beta_0, \beta_1, \sigma) \\ &= (2\pi\sigma^2)^{-\frac{n}{2}} \prod_{i=1}^n e^{-\frac{y_i - (\beta_0 + \beta_1 x_i)}{2\sigma^2}} \end{aligned}$$

Likelihood

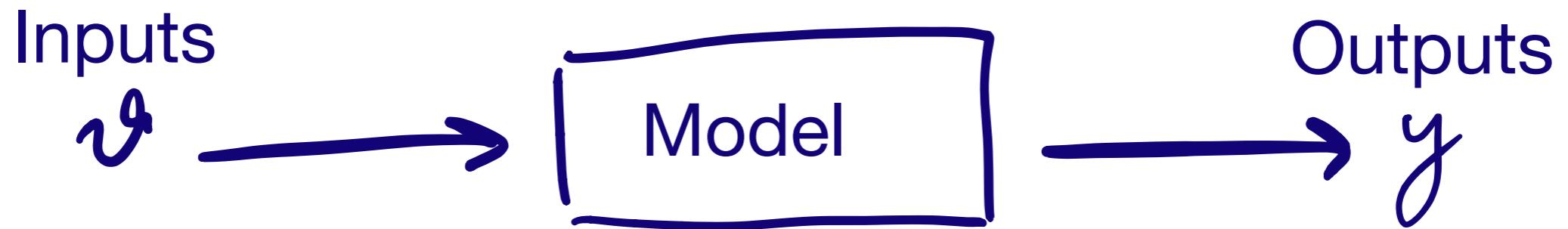
$$p(y = \text{Data} | \theta)$$

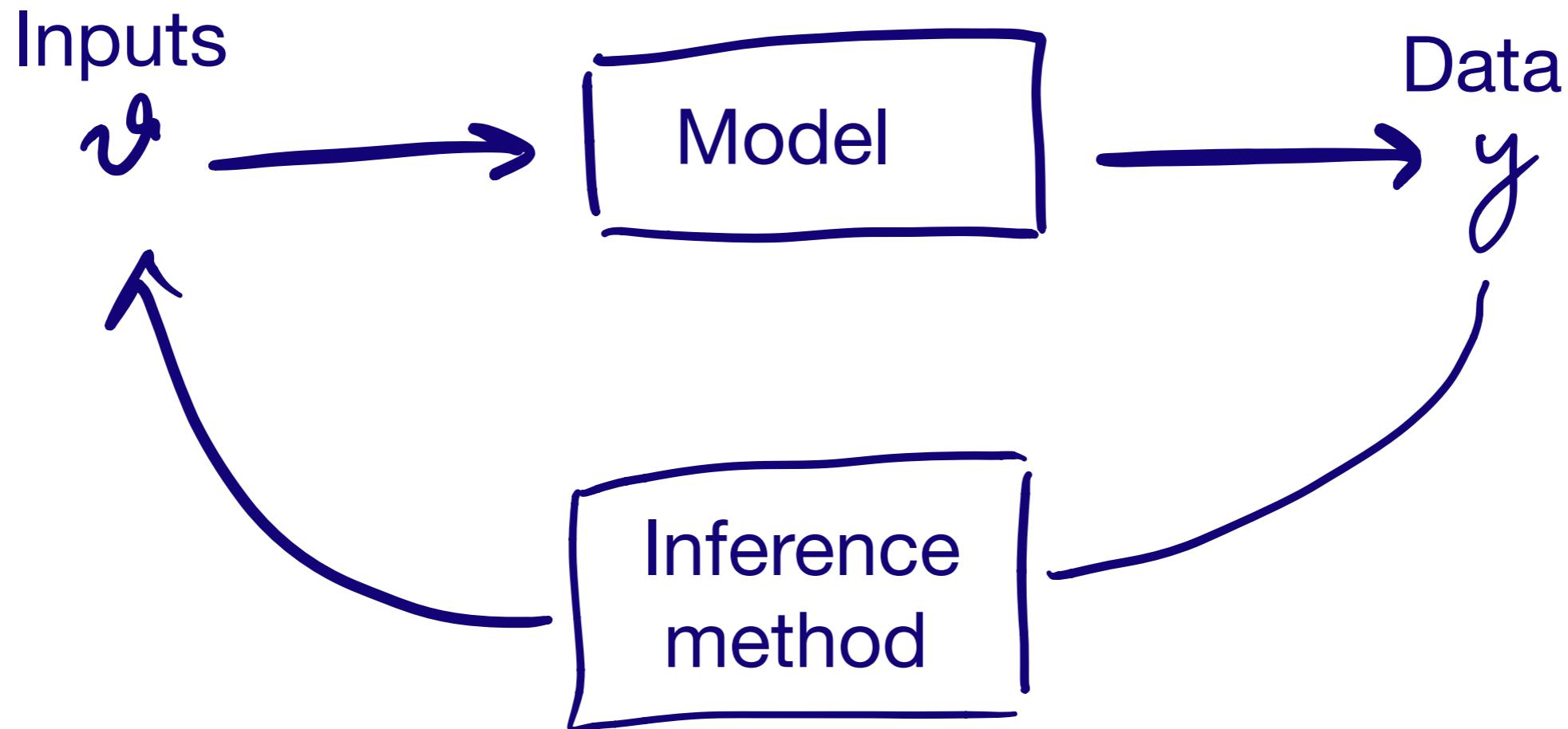
Infectious disease model

$$p(y|\theta) = ???$$

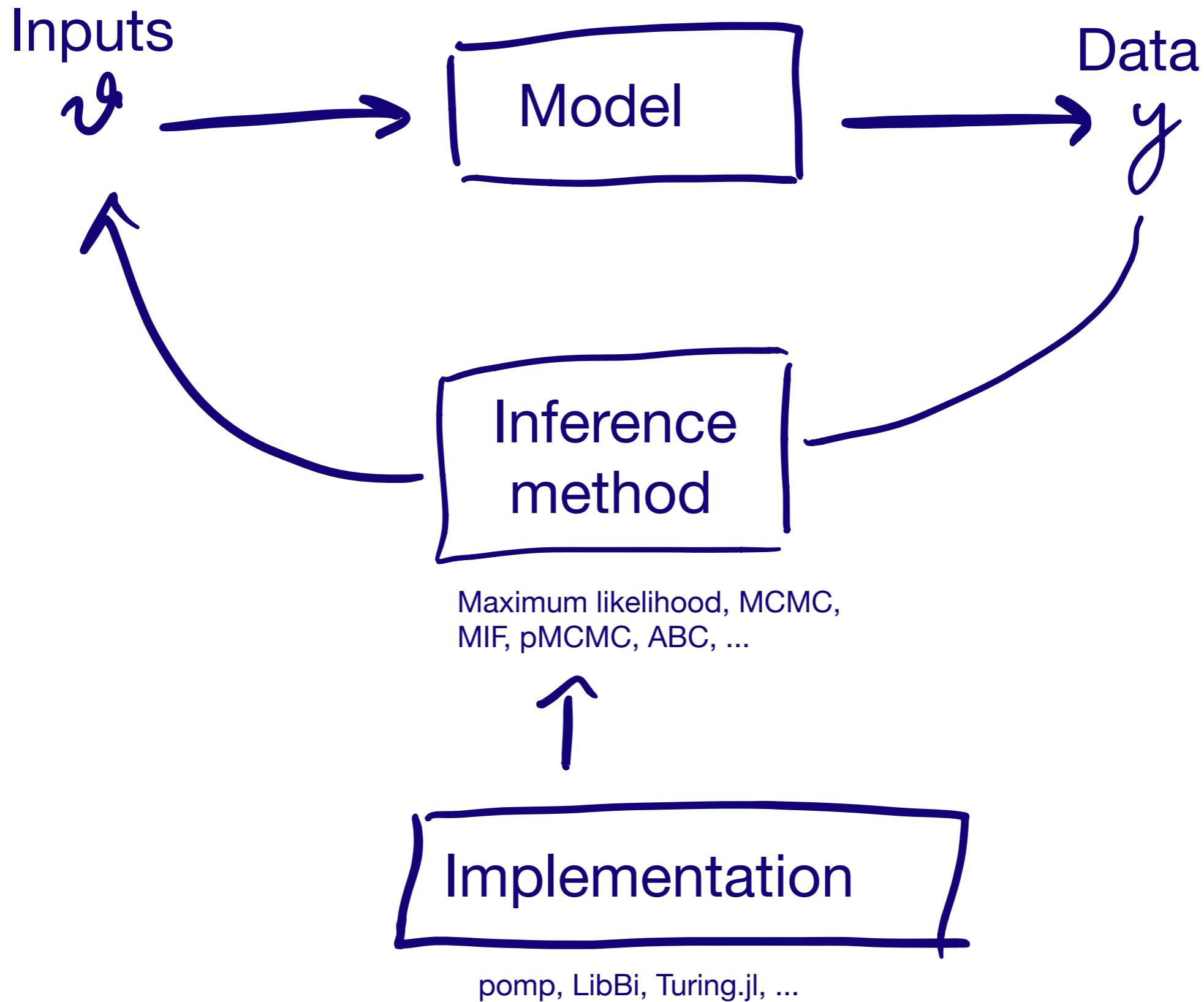


$$\frac{dS}{dt} = -\beta I \frac{S}{N}$$
$$\frac{dI}{dt} = \beta I \frac{S}{N} - \gamma I$$
$$\frac{dR}{dt} = \gamma I$$





Maximum likelihood, MCMC,
MIF, pMCMC, ABC, ...





Bayesian vs Frequentist



- Quantity of interest:
posterior probability

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}$$

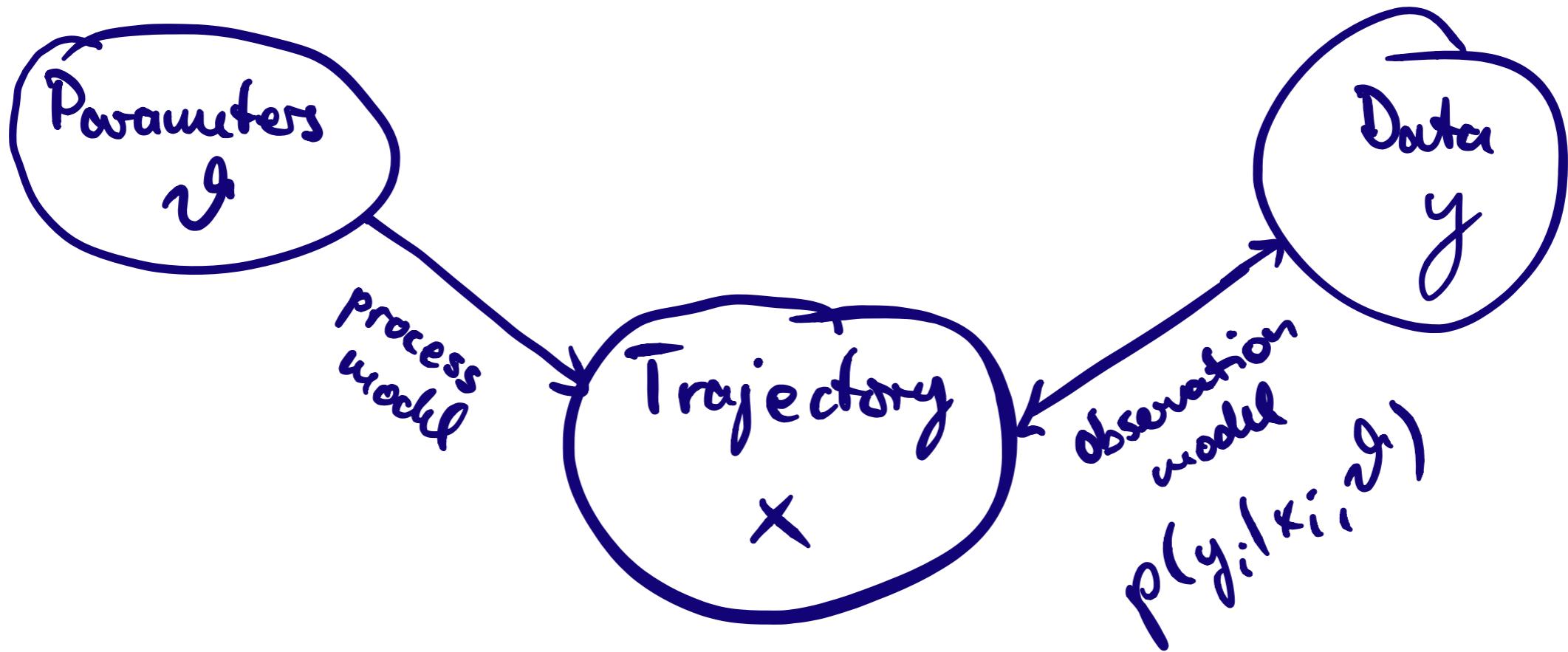
- $p(\theta)$: prior belief
- $p(\theta|y)$: updated belief after confronting model with data

- Quantity of interest:
value of θ that
maximises likelihood

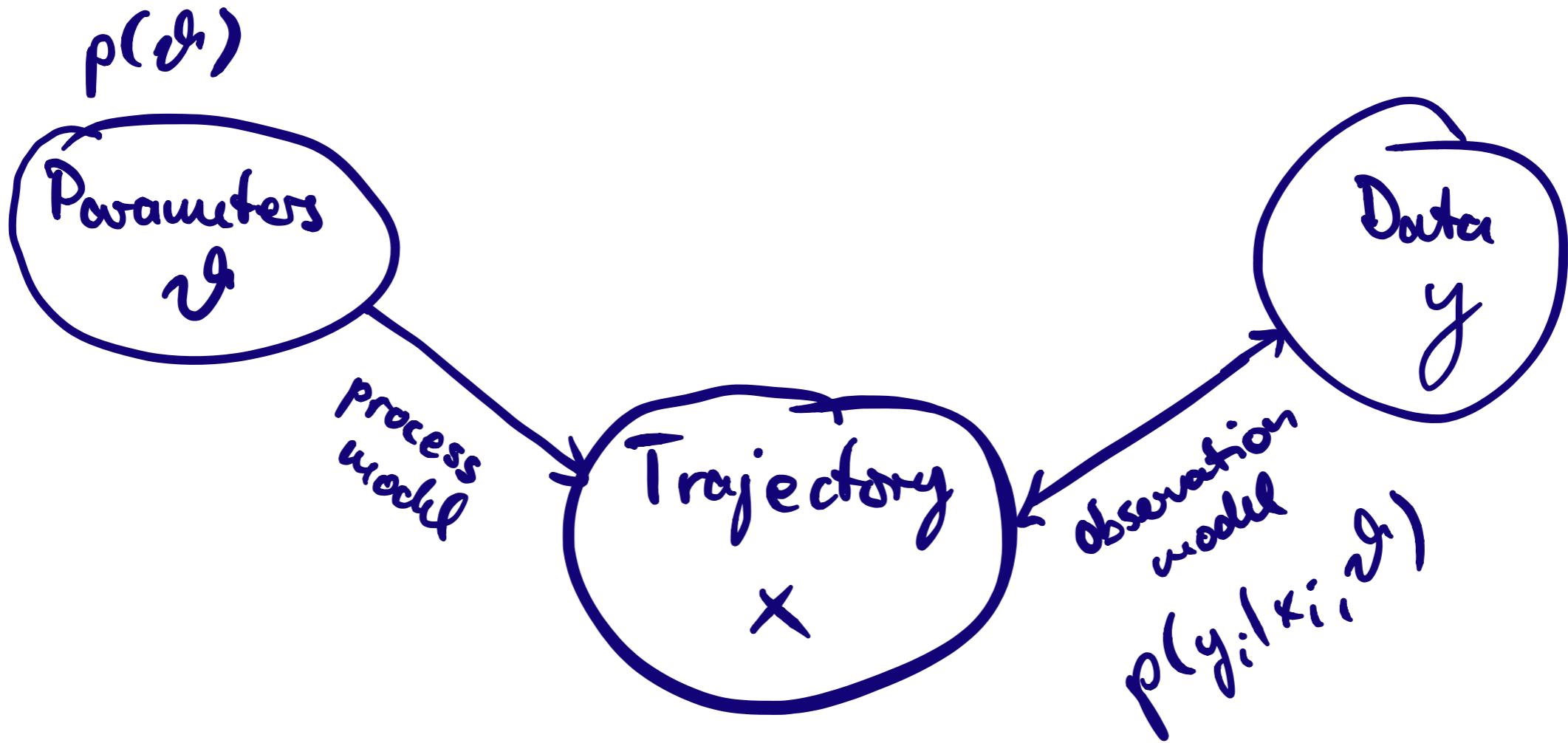
$$L(\theta) = p(y|\theta)$$

Methods for model fitting

1. Deterministic models

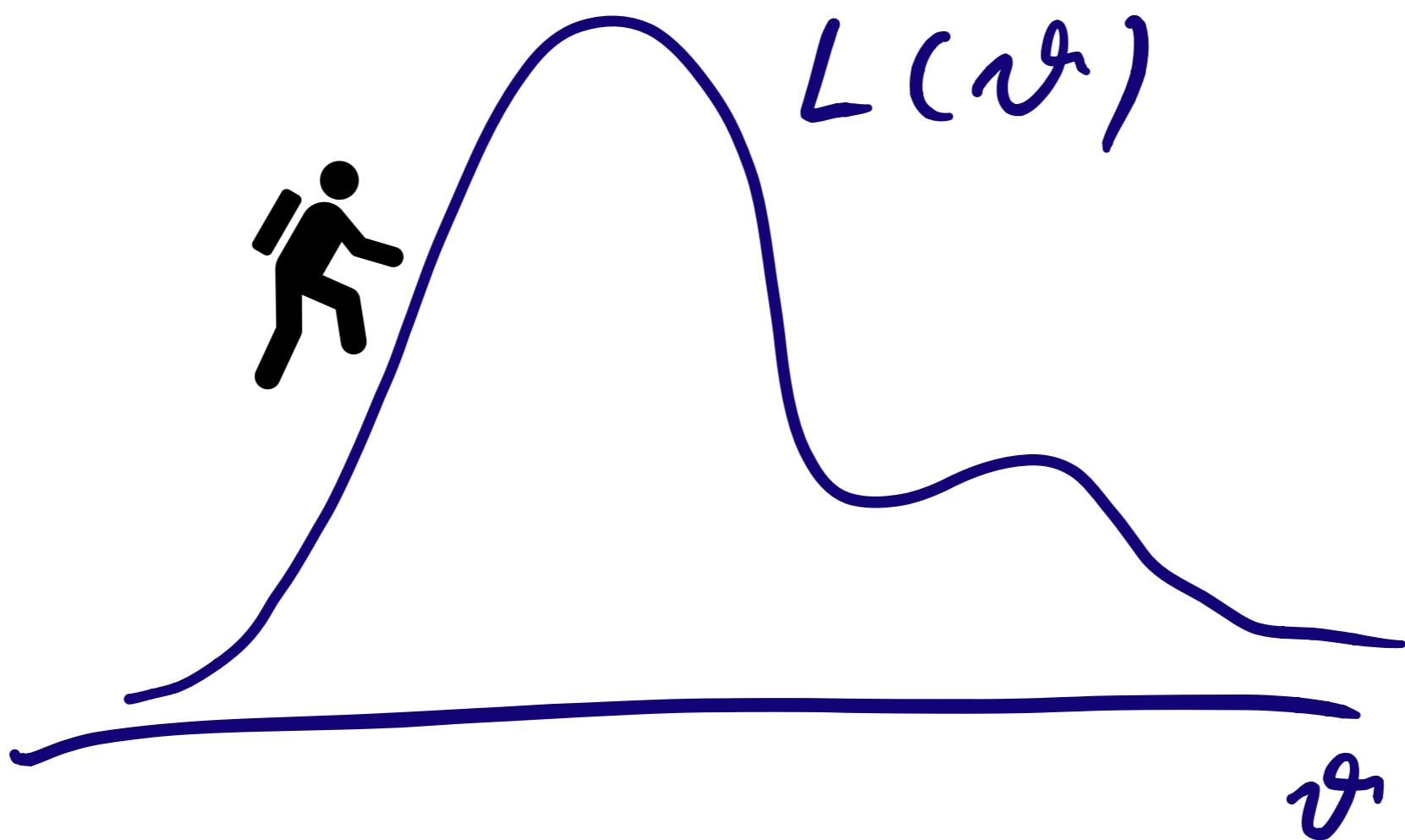


Can compute $p(\text{Data}|\theta)$ for each θ .



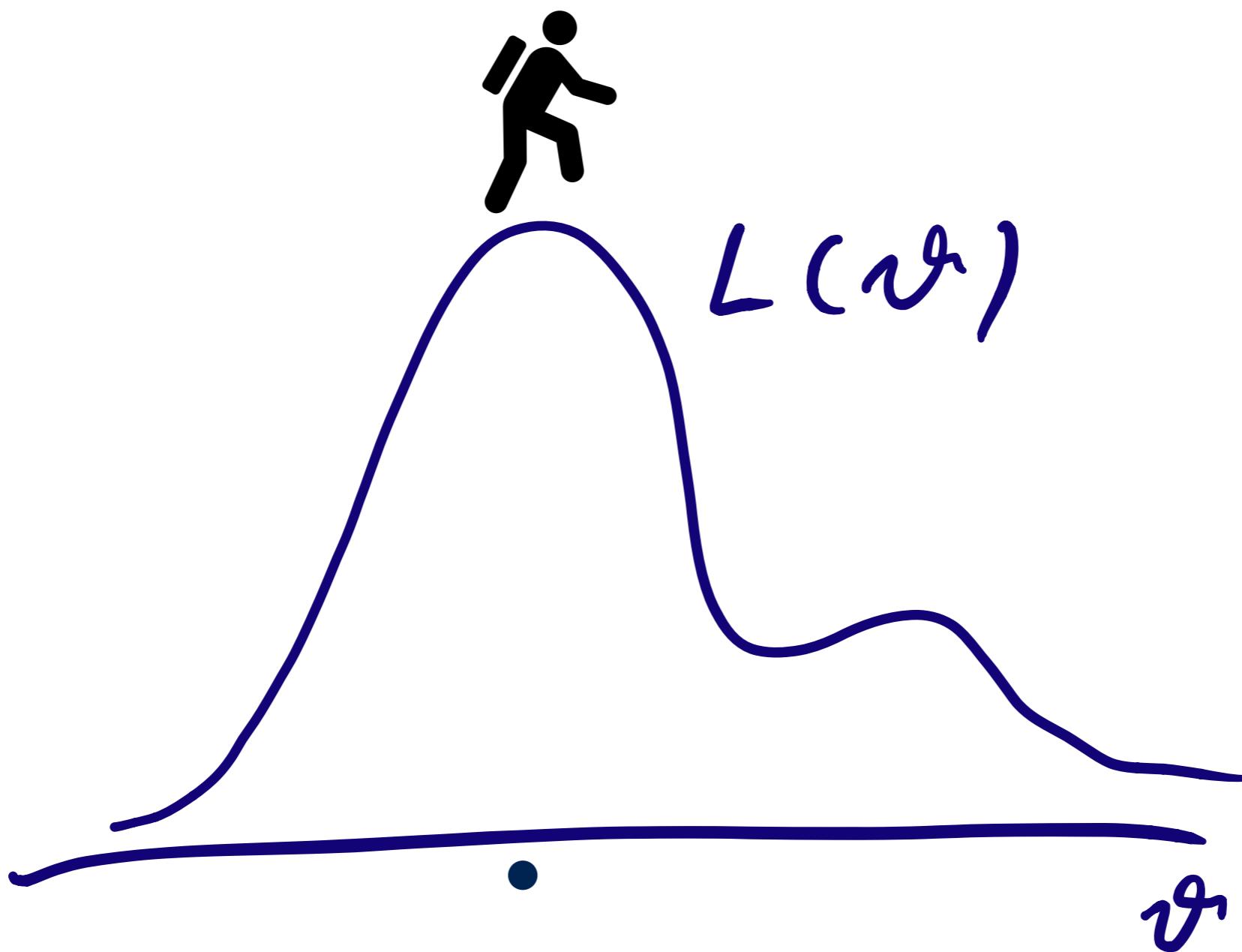
Can compute $p(\theta|\text{Data})$ for each θ .

1. Maximum likelihood via numerical optimisation



Find θ that maximises $L(\theta)=p(\text{Data}|\theta)$.

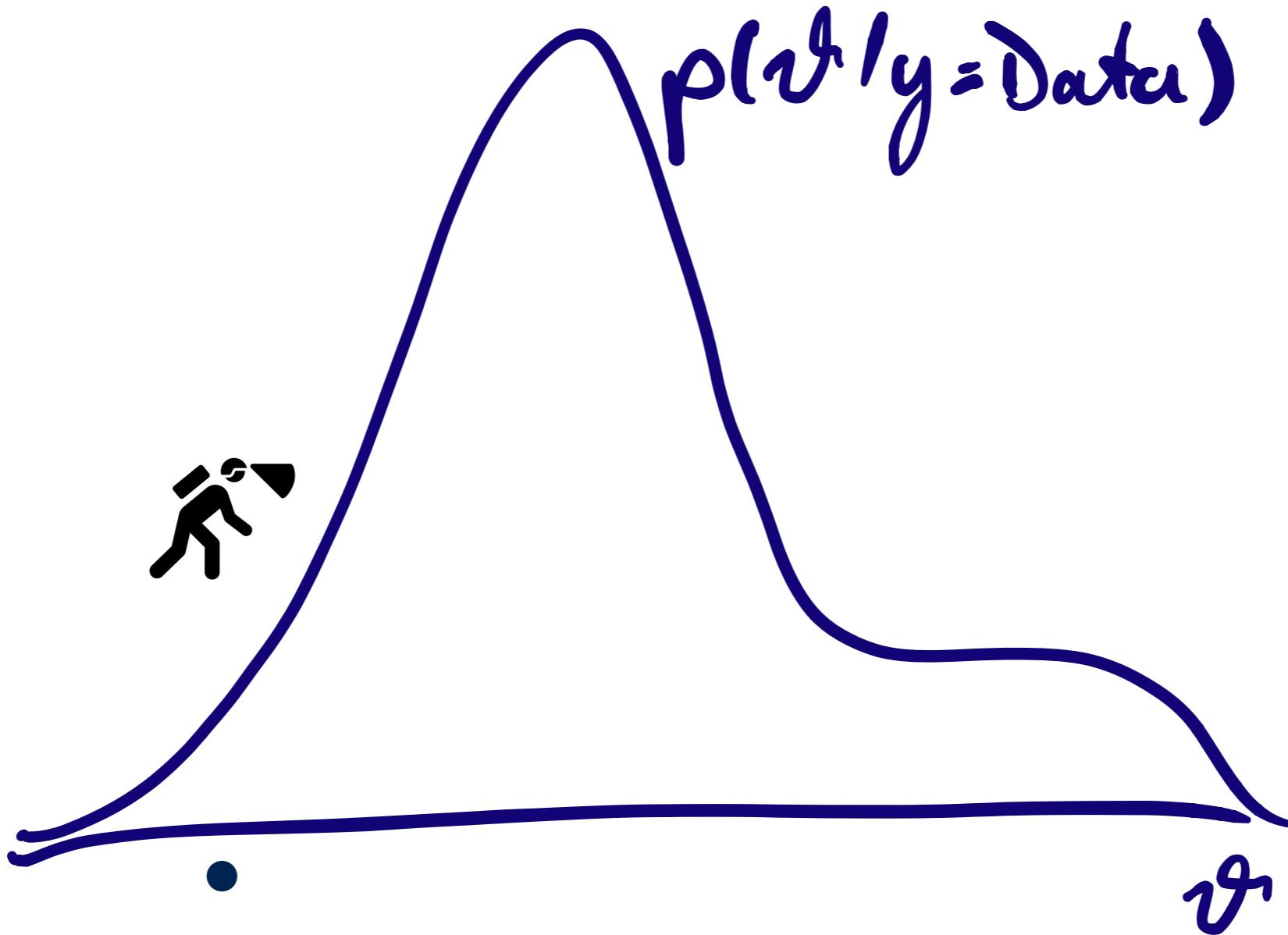
1. Maximum likelihood via numerical optimisation



Find θ that maximises $L(\theta) = p(\text{Data}|\theta)$.



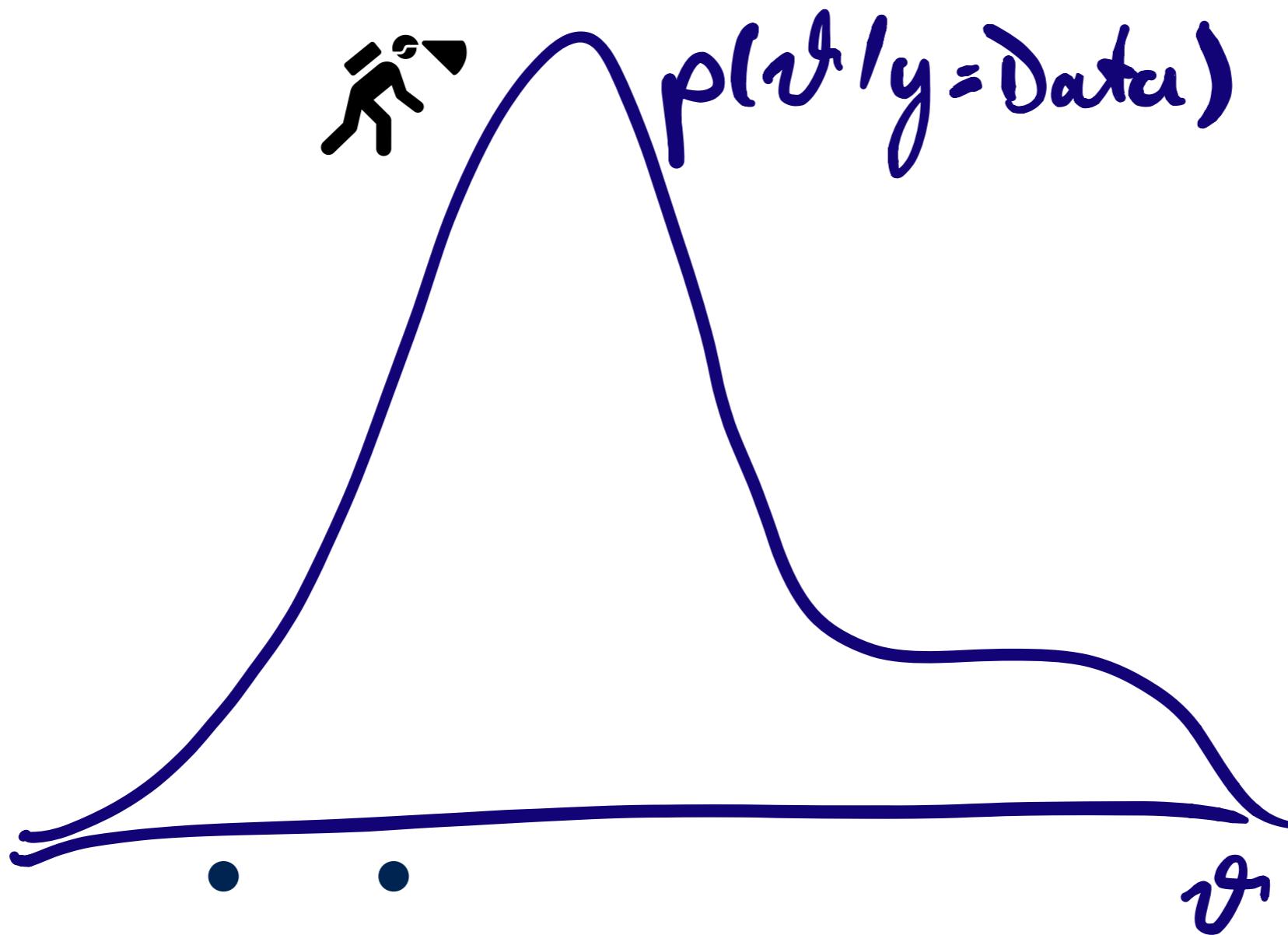
2. Markov-chain Monte Carlo



Sample from $p(\theta | \text{Data})$.



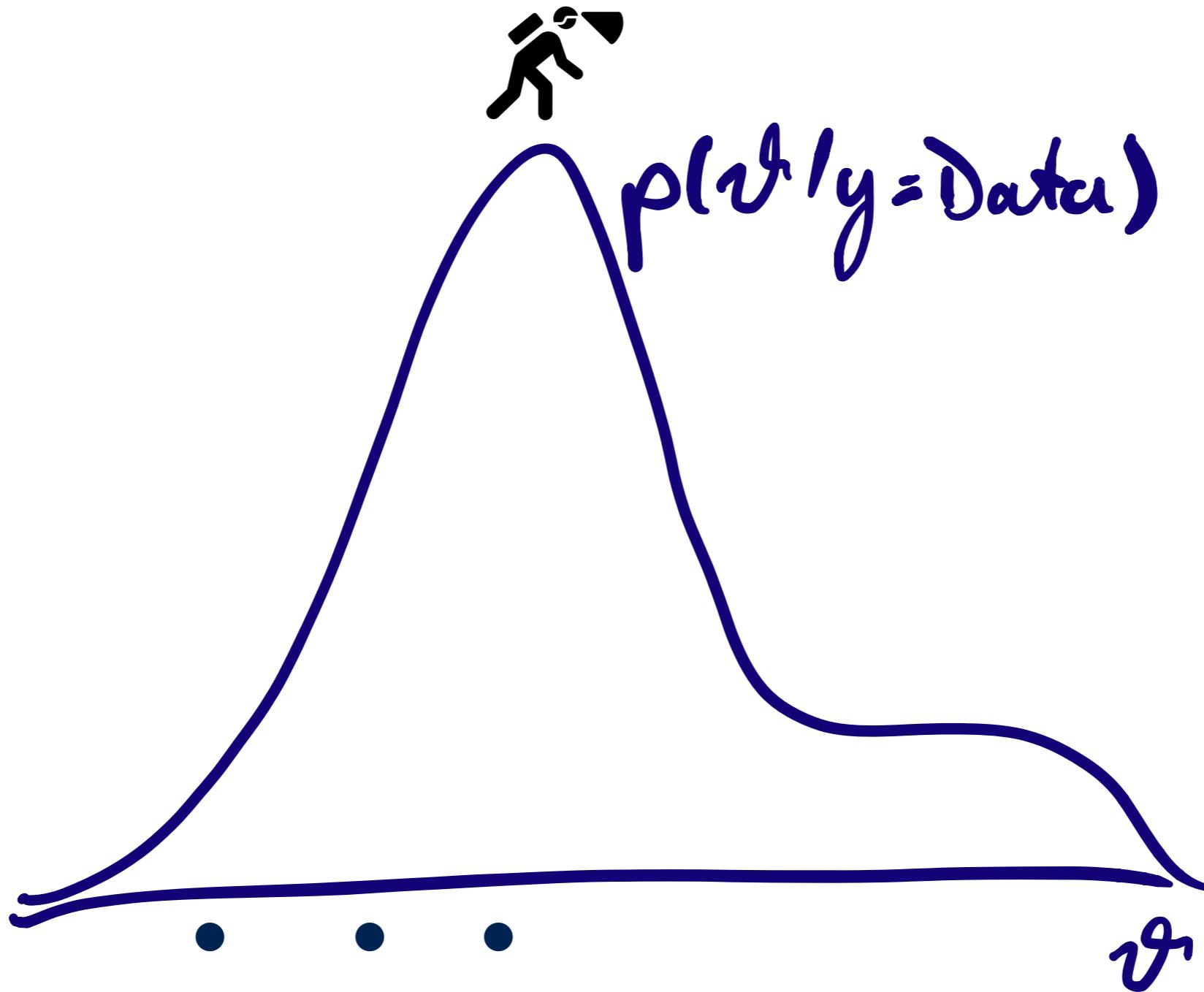
2. Markov-chain Monte Carlo



Sample from $p(\theta | \text{Data})$.



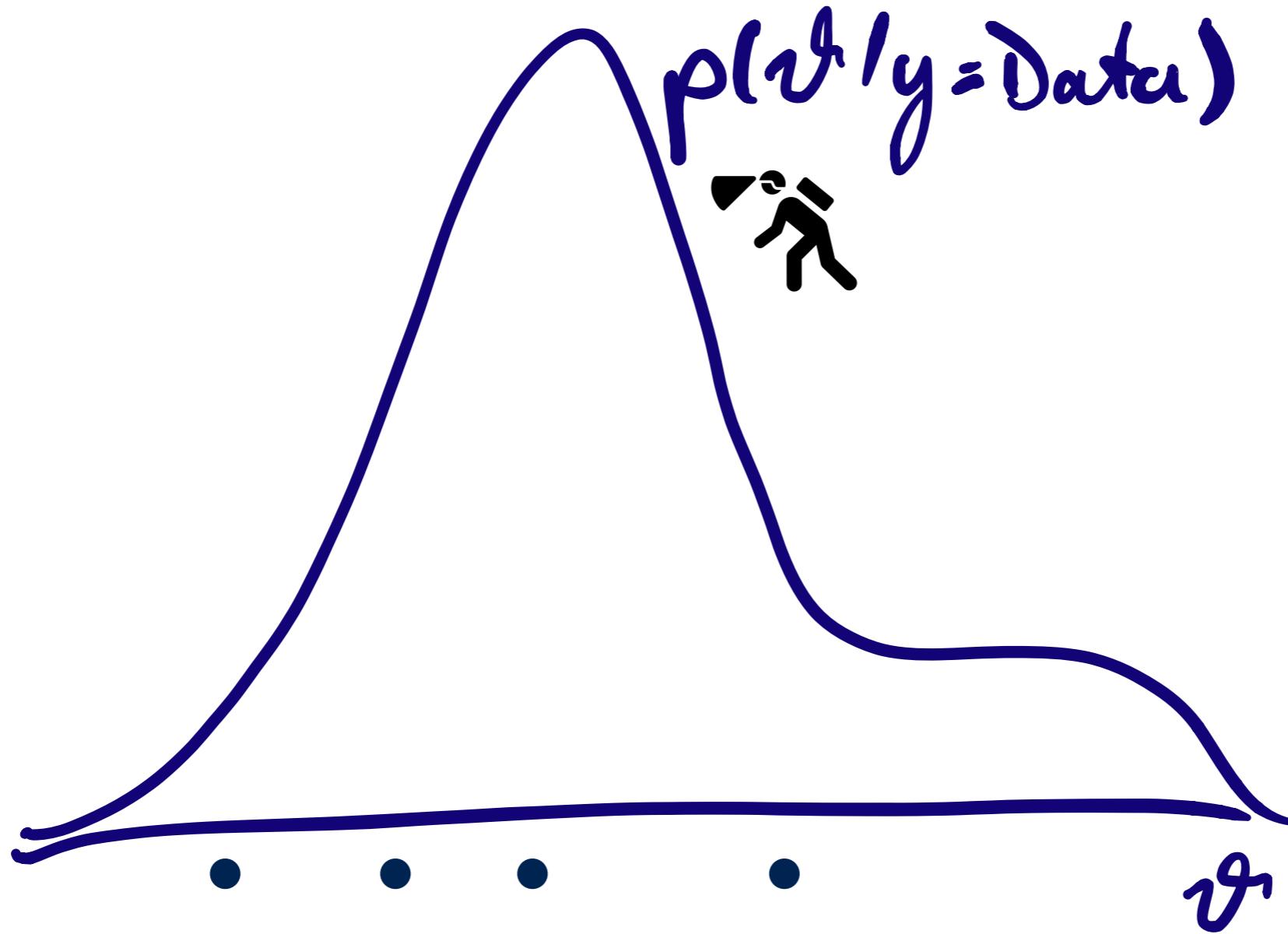
2. Markov-chain Monte Carlo



Sample from $p(\theta | \text{Data})$.



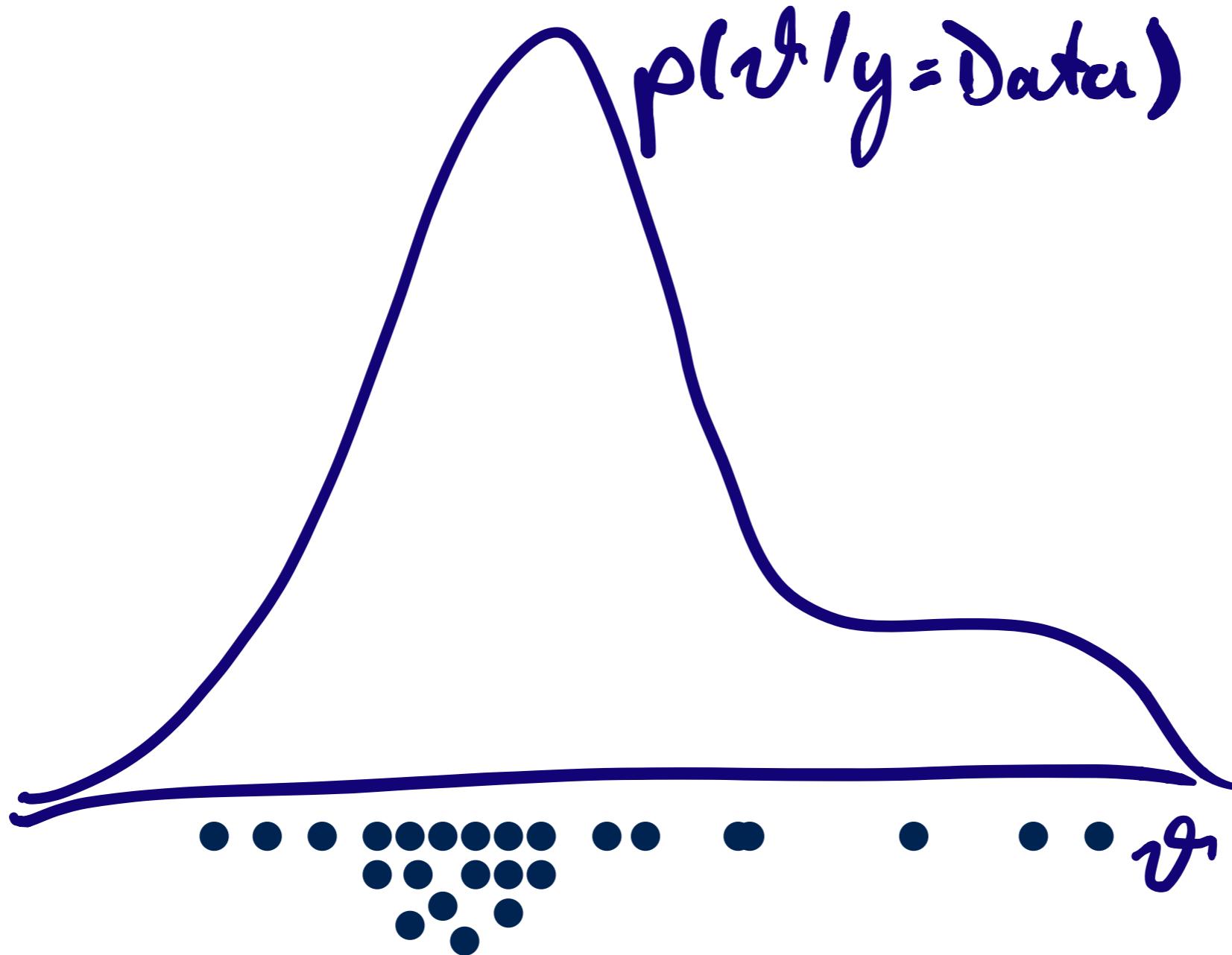
2. Markov-chain Monte Carlo



Sample from $p(\theta | \text{Data})$.

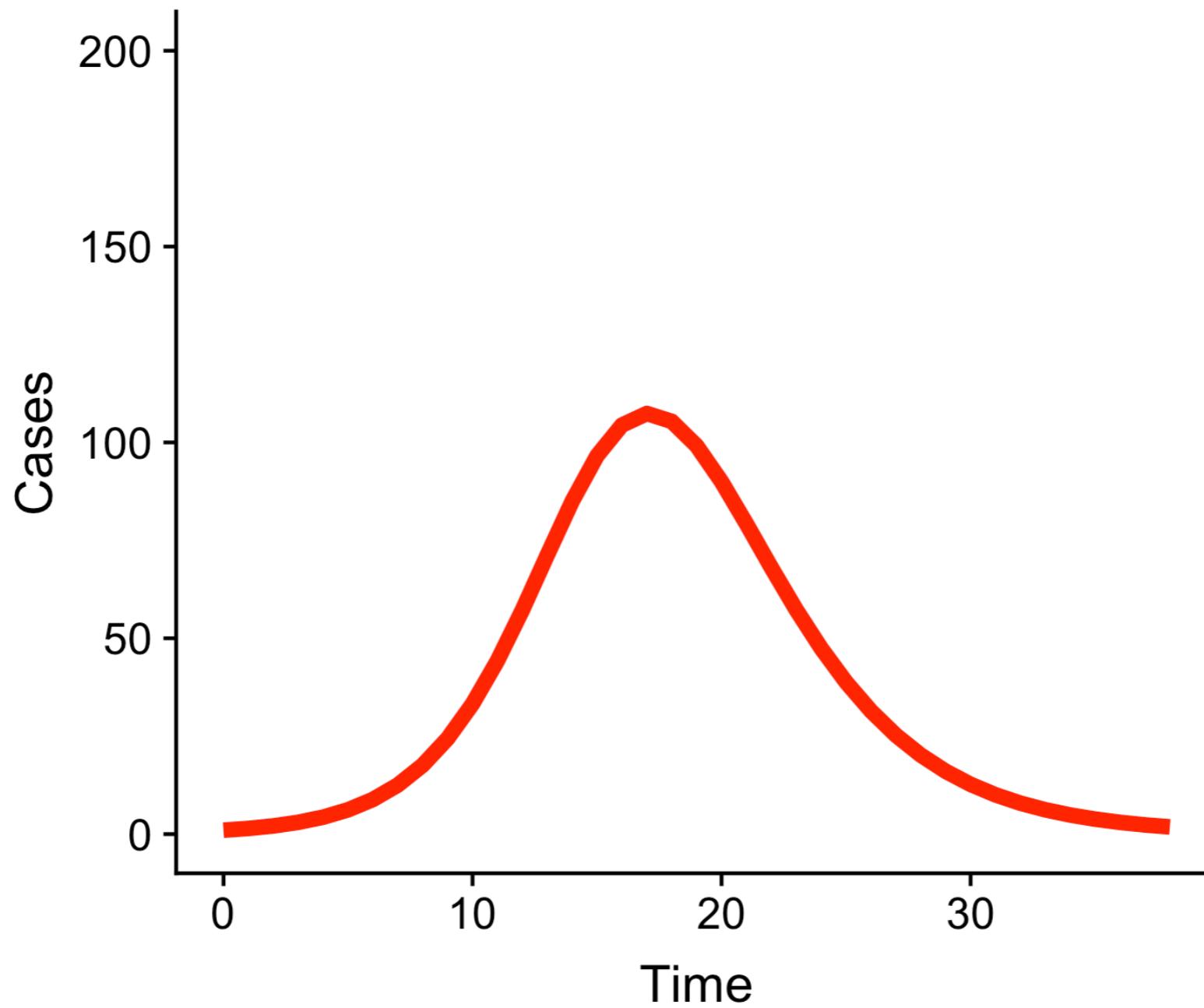


2. Markov-chain Monte Carlo



Sample from $p(\theta | \text{Data})$.

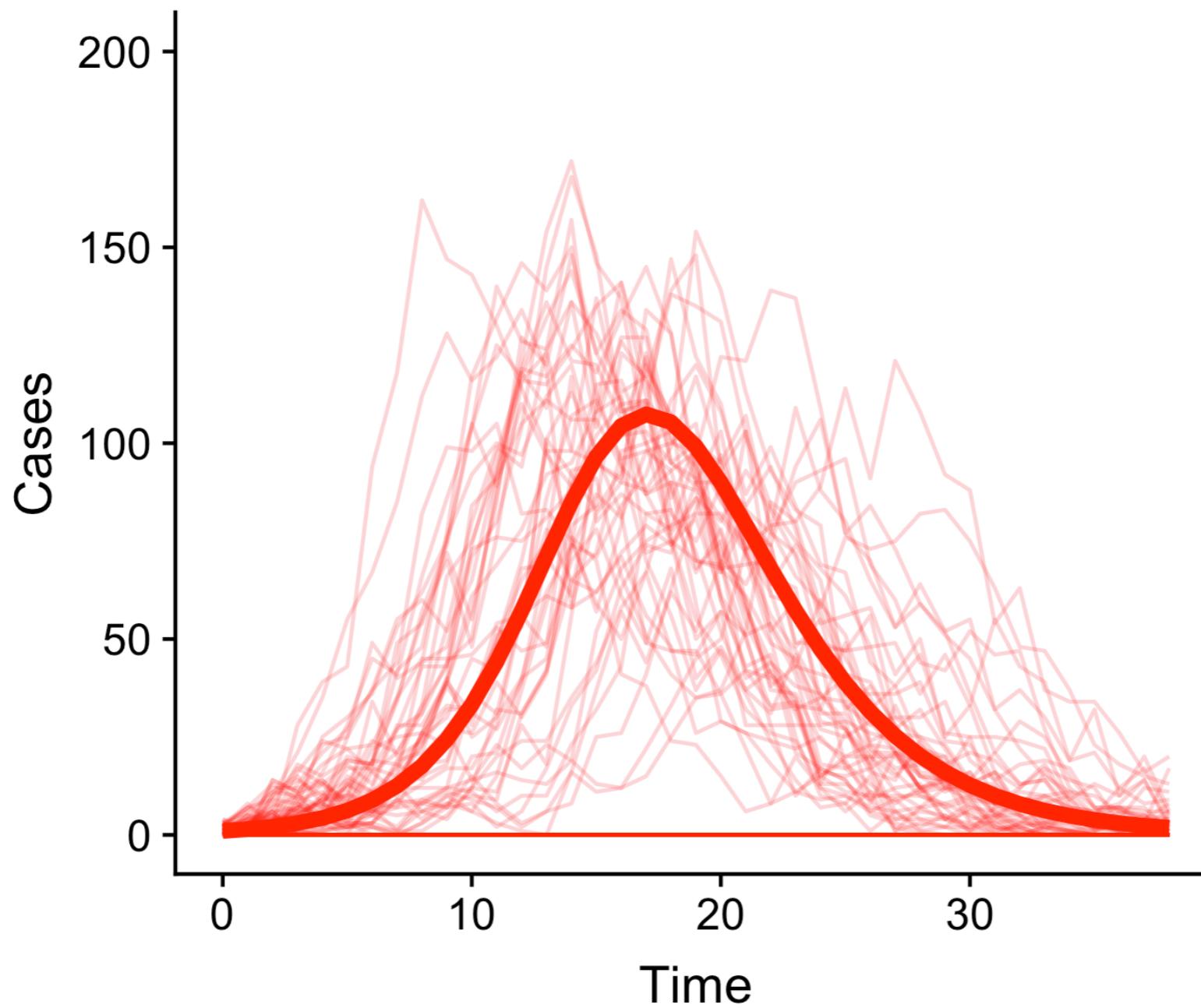
Fitting deterministic vs. stochastic models



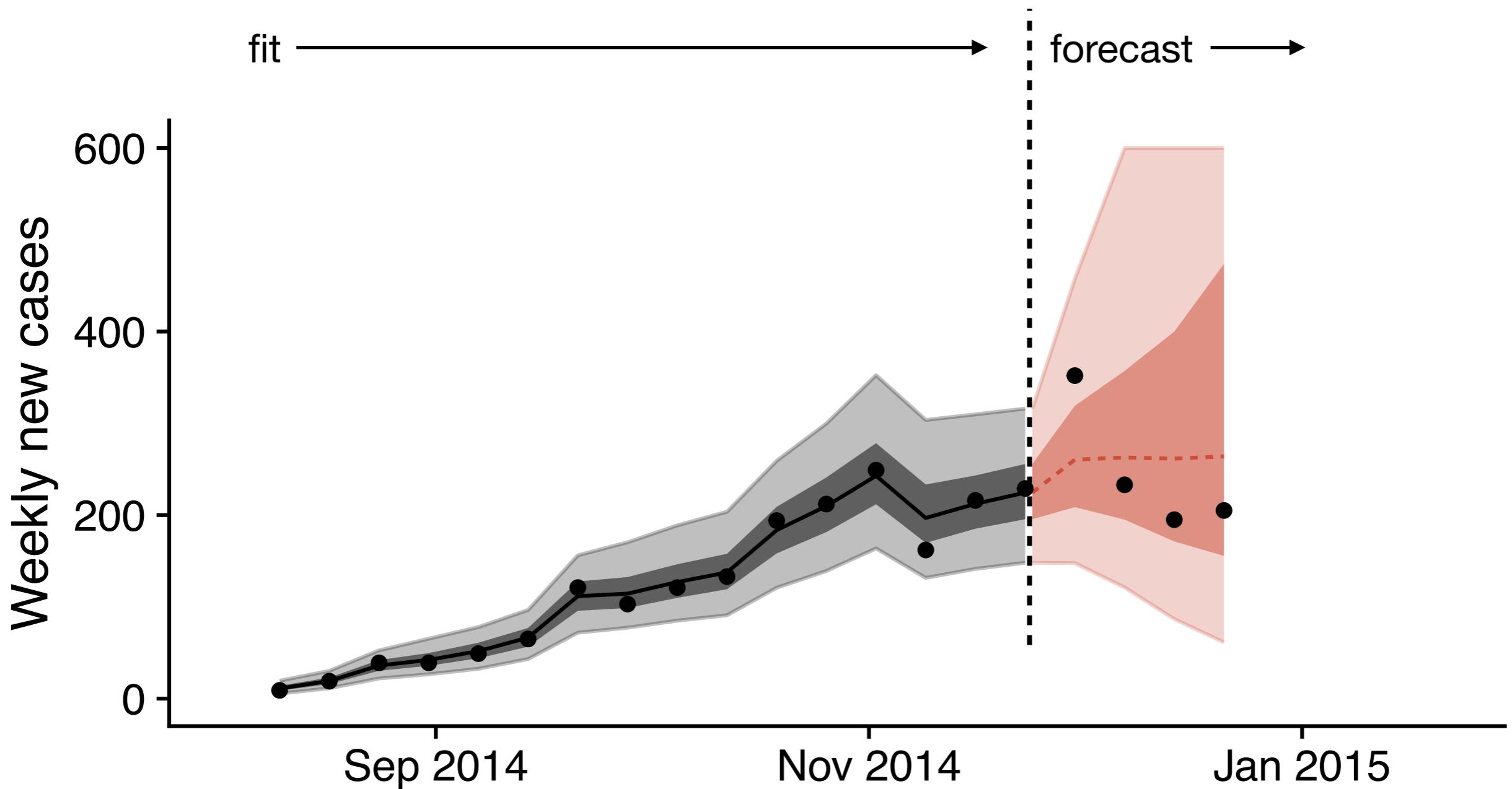
Methods for model fitting

2. Stochastic models

Fitting deterministic vs. stochastic models



Example: real-time modelling and prediction





Research Fellow In Real-Time Modelling Of Infectious Disease Outbreaks x2

Department of Infectious Disease Epidemiology

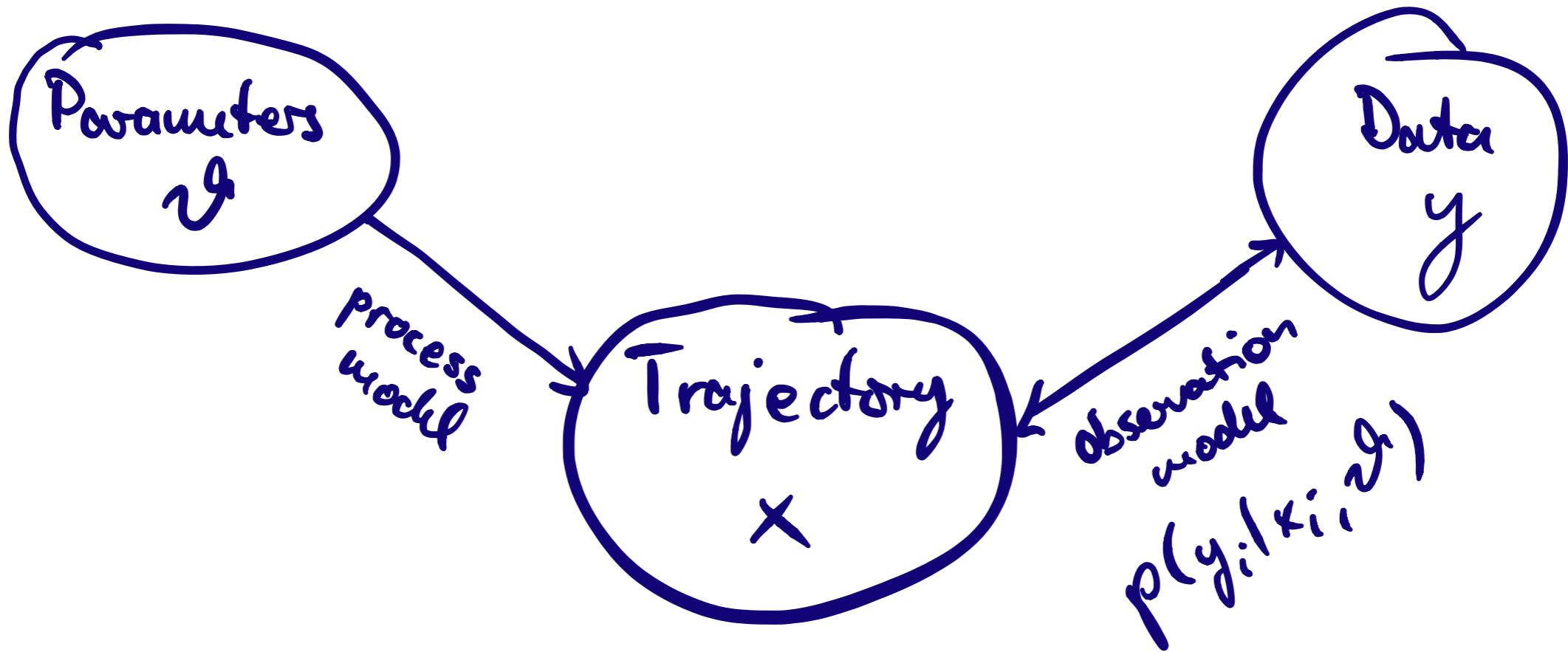
Salary: £39,304 to £44,634 per annum, inclusive.

Closing Date: Wednesday 14 November 2018

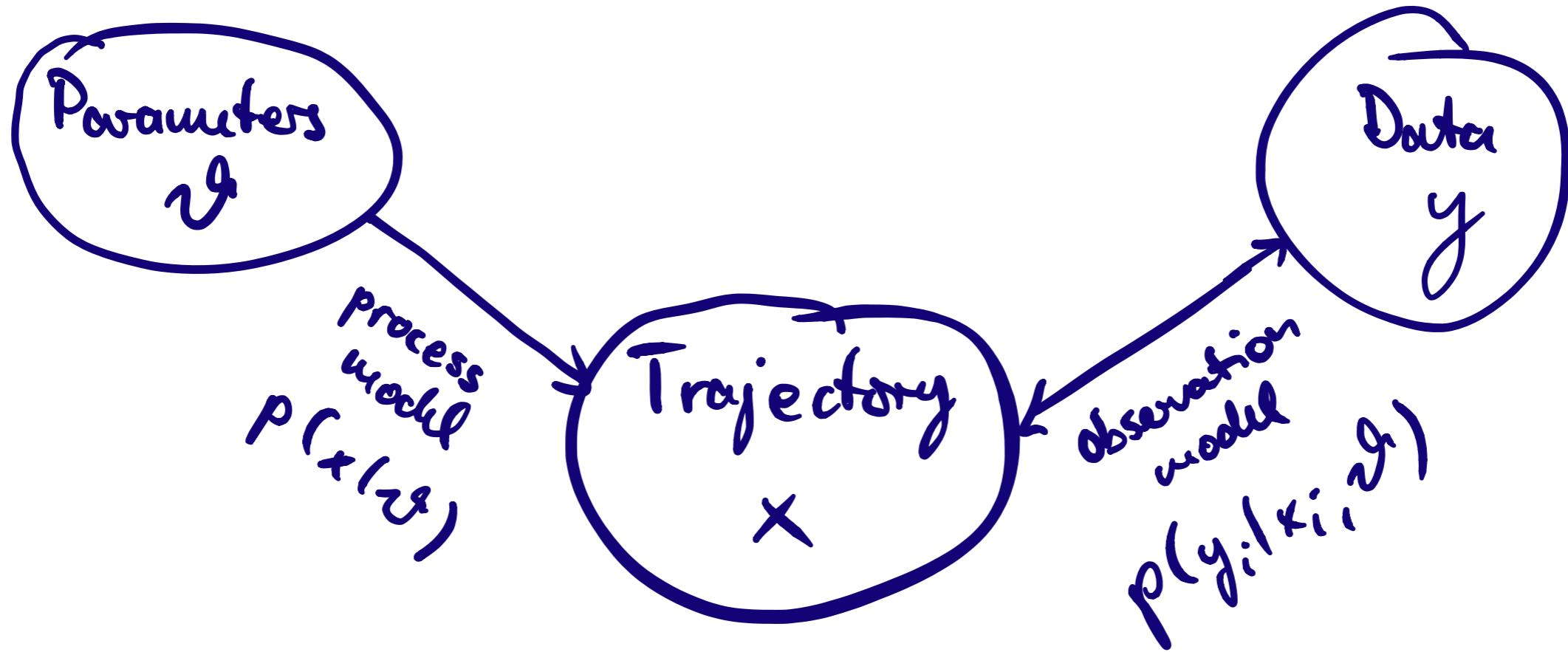
Reference: EPH-IDE-2018-49

There are two posts available to join a team doing research on mathematical models to inform decision making during infectious disease outbreaks. The posts will be based in London at the London School of Hygiene & Tropical Medicine. The post-holders will be supervised by Dr Sebastian Funk and will join a highly successful and supportive group at the Centre for the Mathematical Modelling (CMMID).

The posts will be part of a new Wellcome Trust funded project aiming to improve the use of real-time modelling for meaningful decision support in infectious disease outbreaks. The main task of the post-holders will be to develop methods for model-based forecasting that can be readily deployed in an outbreak situation and updated when new data become available.



Can compute $p(\text{Data}|\theta)$ for each θ .



Cannot compute $p(\text{Data}|\theta)$ for given θ .

Marginal likelihood

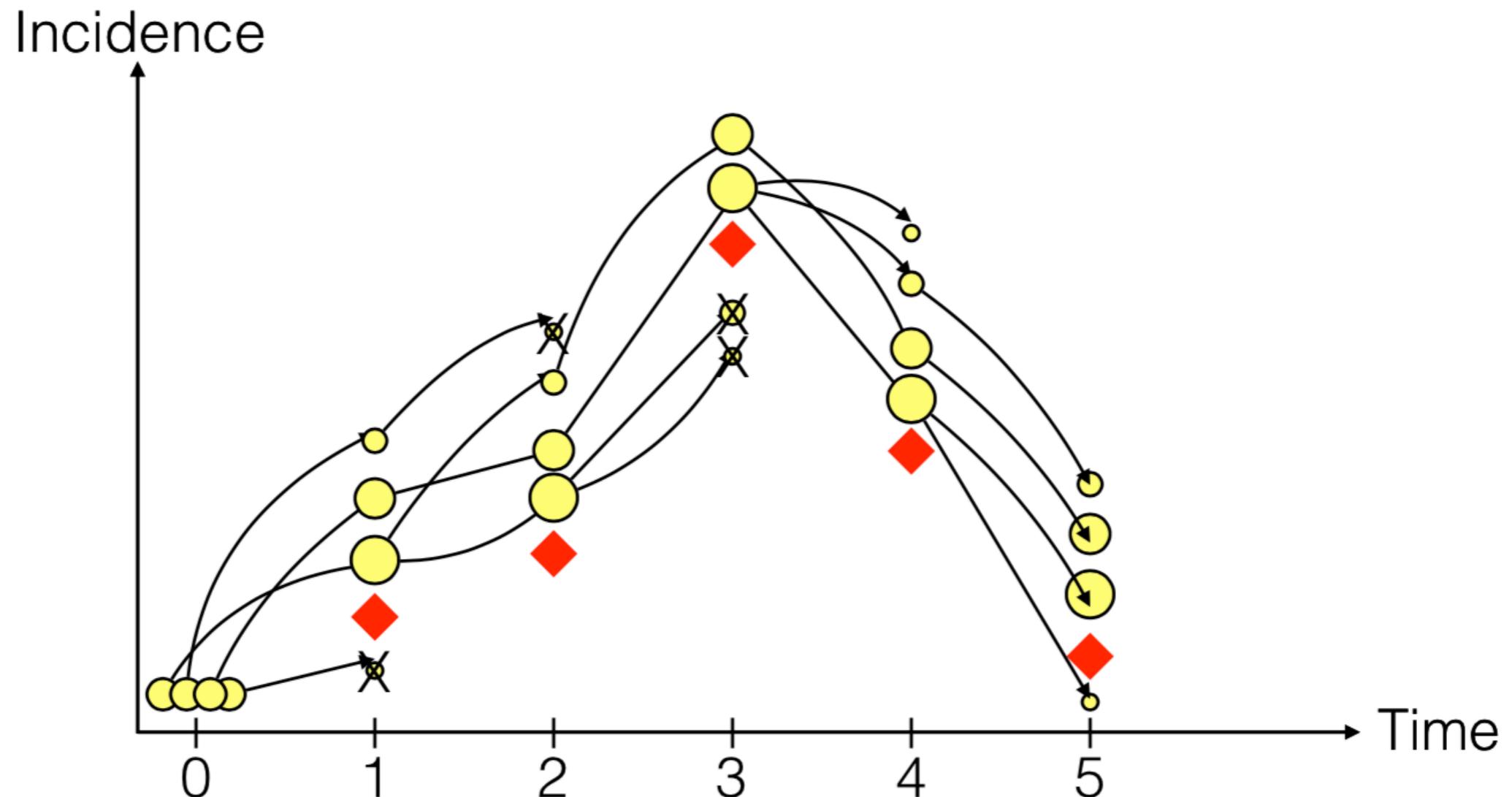
$$p(y|\theta) = \sum p(y|x, \theta)p(x|\theta)$$

x



All possible trajectories, given θ

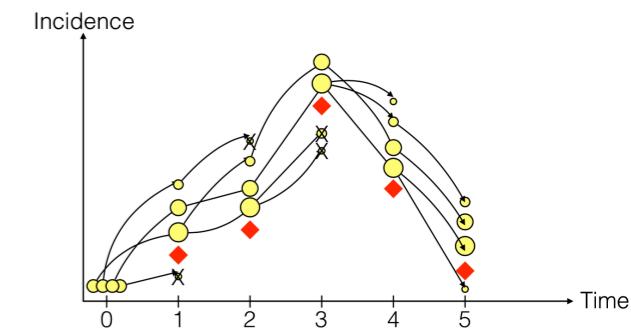
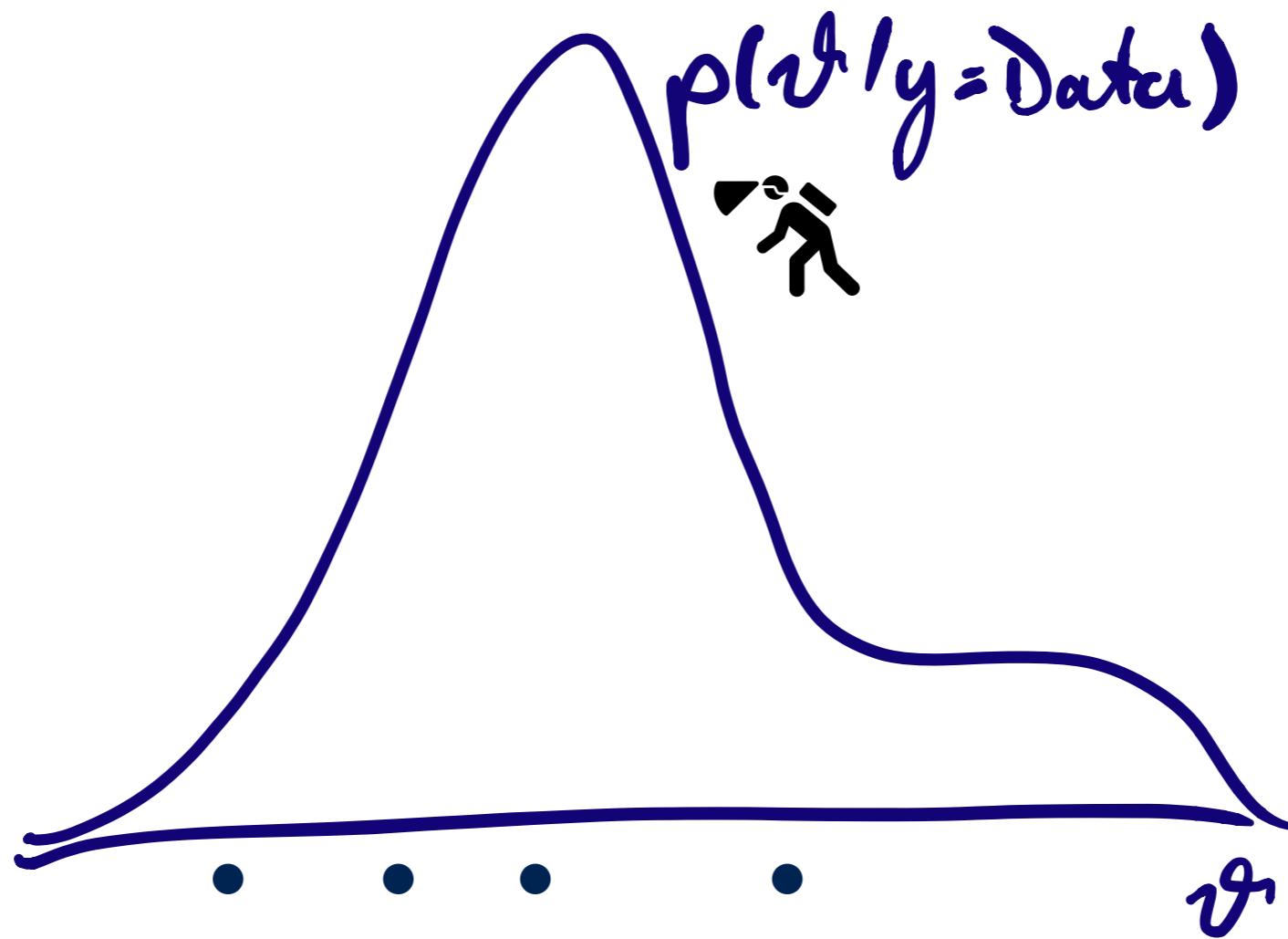
The particle filter



Estimate $p(\text{Data}|\theta)$ for given θ .

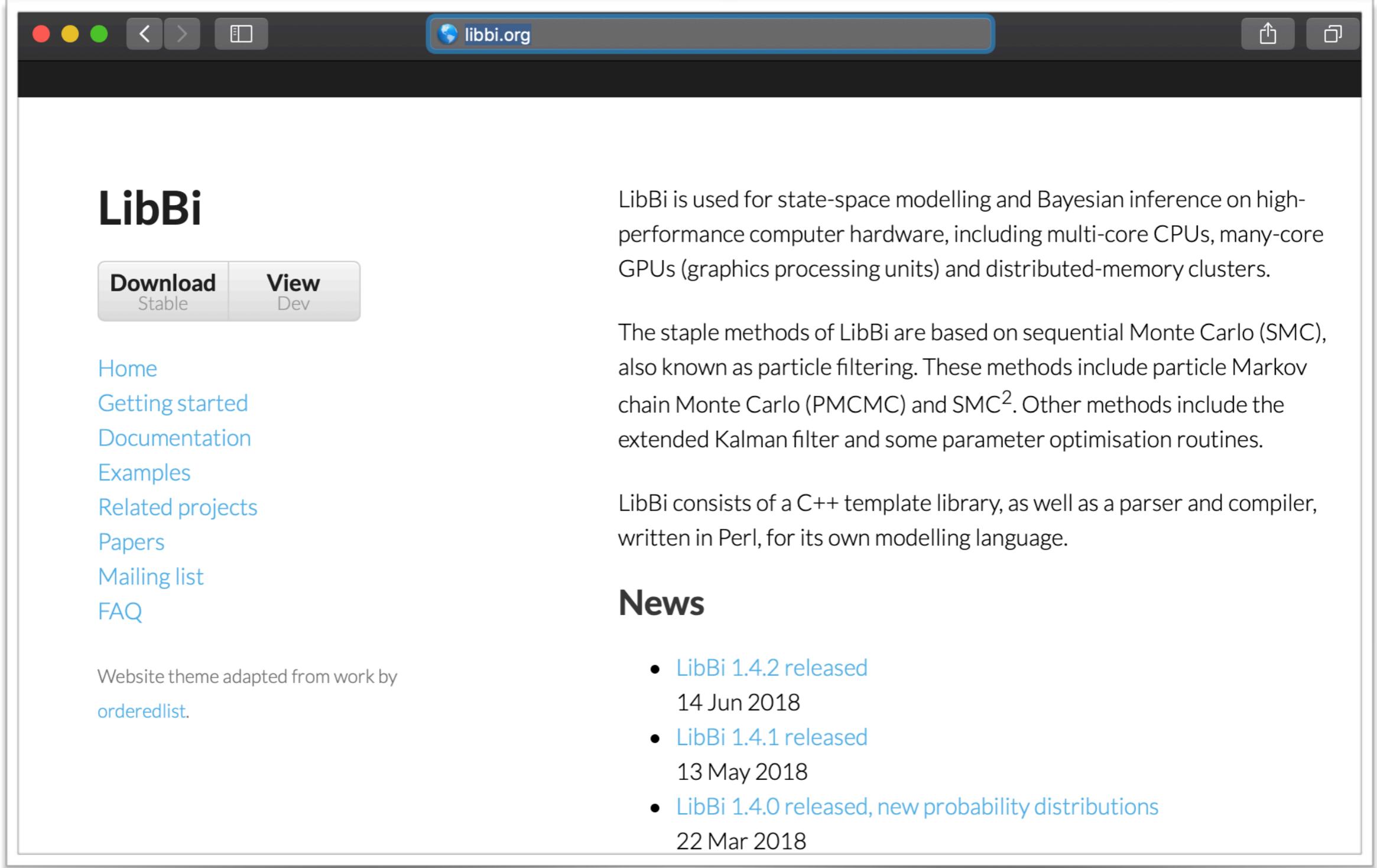


3. Particle Markov-chain Monte Carlo (pMCMC)



MCMC with $p(\text{Data}|\theta)$ replaced with particle filter estimate

LibBi



The screenshot shows a Mac OS X desktop environment with a window for the LibBi website. The window title bar says "libbi.org". The main content area of the website features the LibBi logo at the top left. Below it are two buttons: "Download Stable" and "View Dev". To the right of these buttons is a large text block about LibBi's use cases. Further down, there are sections for "News" (listing recent releases) and a sidebar with links to "Home", "Getting started", "Documentation", "Examples", "Related projects", "Papers", "Mailing list", and "FAQ". At the bottom left, there is a note about the website theme.

LibBi is used for state-space modelling and Bayesian inference on high-performance computer hardware, including multi-core CPUs, many-core GPUs (graphics processing units) and distributed-memory clusters.

The staple methods of LibBi are based on sequential Monte Carlo (SMC), also known as particle filtering. These methods include particle Markov chain Monte Carlo (PMCMC) and SMC². Other methods include the extended Kalman filter and some parameter optimisation routines.

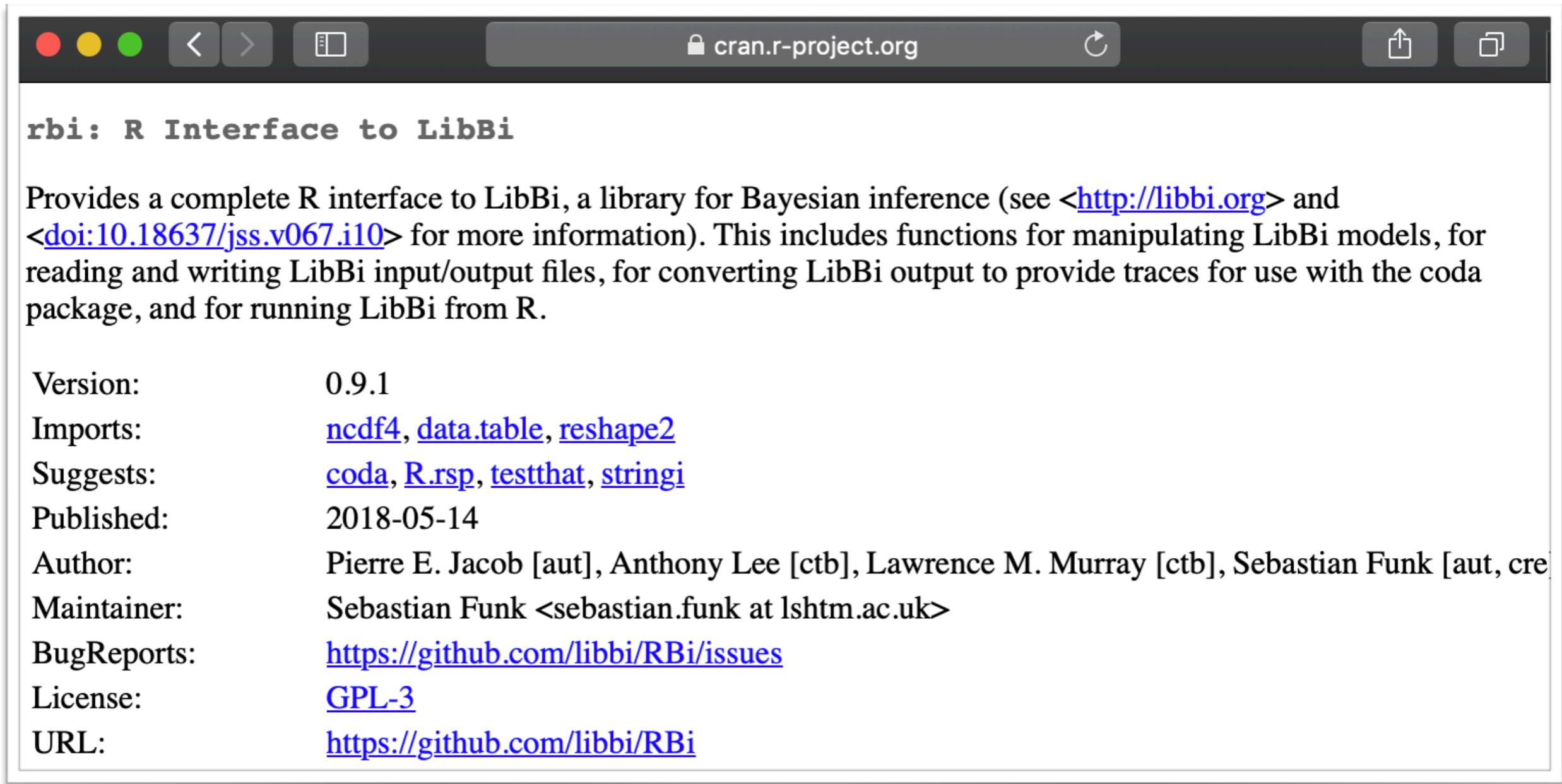
LibBi consists of a C++ template library, as well as a parser and compiler, written in Perl, for its own modelling language.

News

- [LibBi 1.4.2 released](#)
14 Jun 2018
- [LibBi 1.4.1 released](#)
13 May 2018
- [LibBi 1.4.0 released, new probability distributions](#)
22 Mar 2018

Website theme adapted from work by [orderedlist](#).

rbi



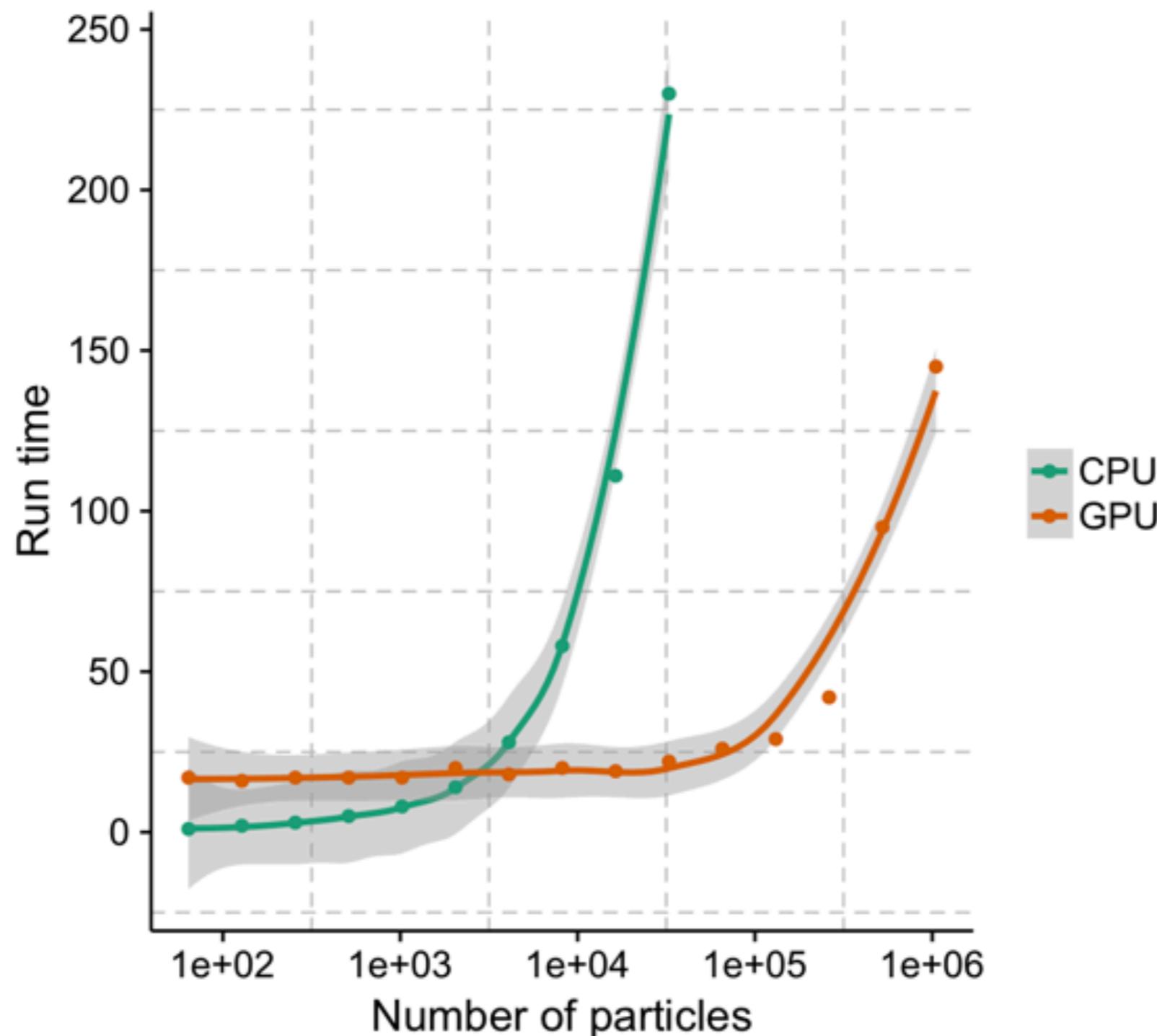
The screenshot shows a web browser window with the URL cran.r-project.org in the address bar. The page content is about the **rbi** package, which provides an R interface to LibBi. The package version is 0.9.1, published on 2018-05-14. It imports `ncdf4`, `data.table`, and `reshape2`. It suggests `coda`, `R.rsp`, `testthat`, and `stringi`. The author is Pierre E. Jacob [aut], Anthony Lee [ctb], Lawrence M. Murray [ctb], and Sebastian Funk [aut, cre]. The maintainer is Sebastian Funk <sebastian.funk@lshtm.ac.uk>. Bug reports can be filed at <https://github.com/libbi/RBi/issues>. The license is [GPL-3](#). The URL for the GitHub repository is <https://github.com/libbi/RBi>.

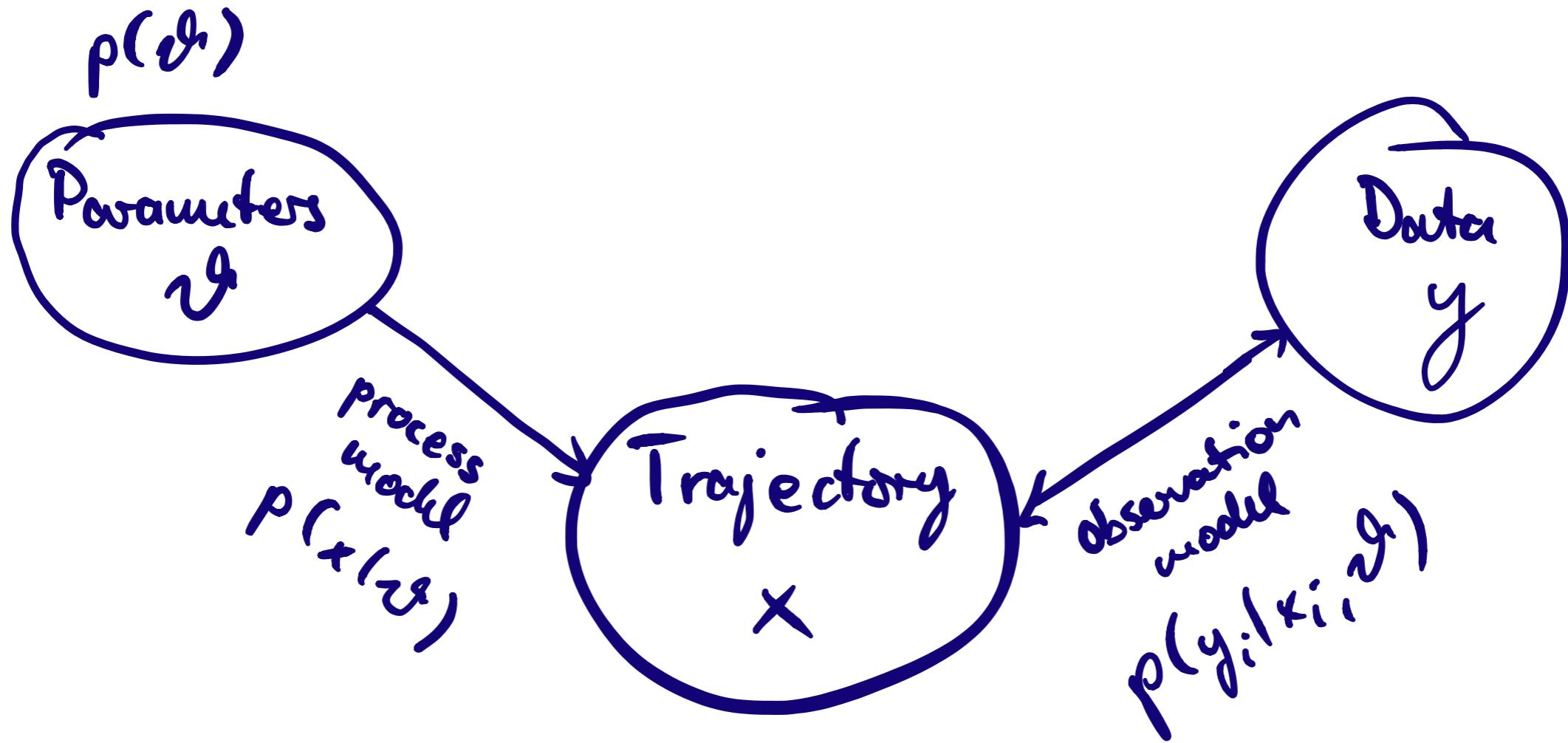
rbi: R Interface to LibBi

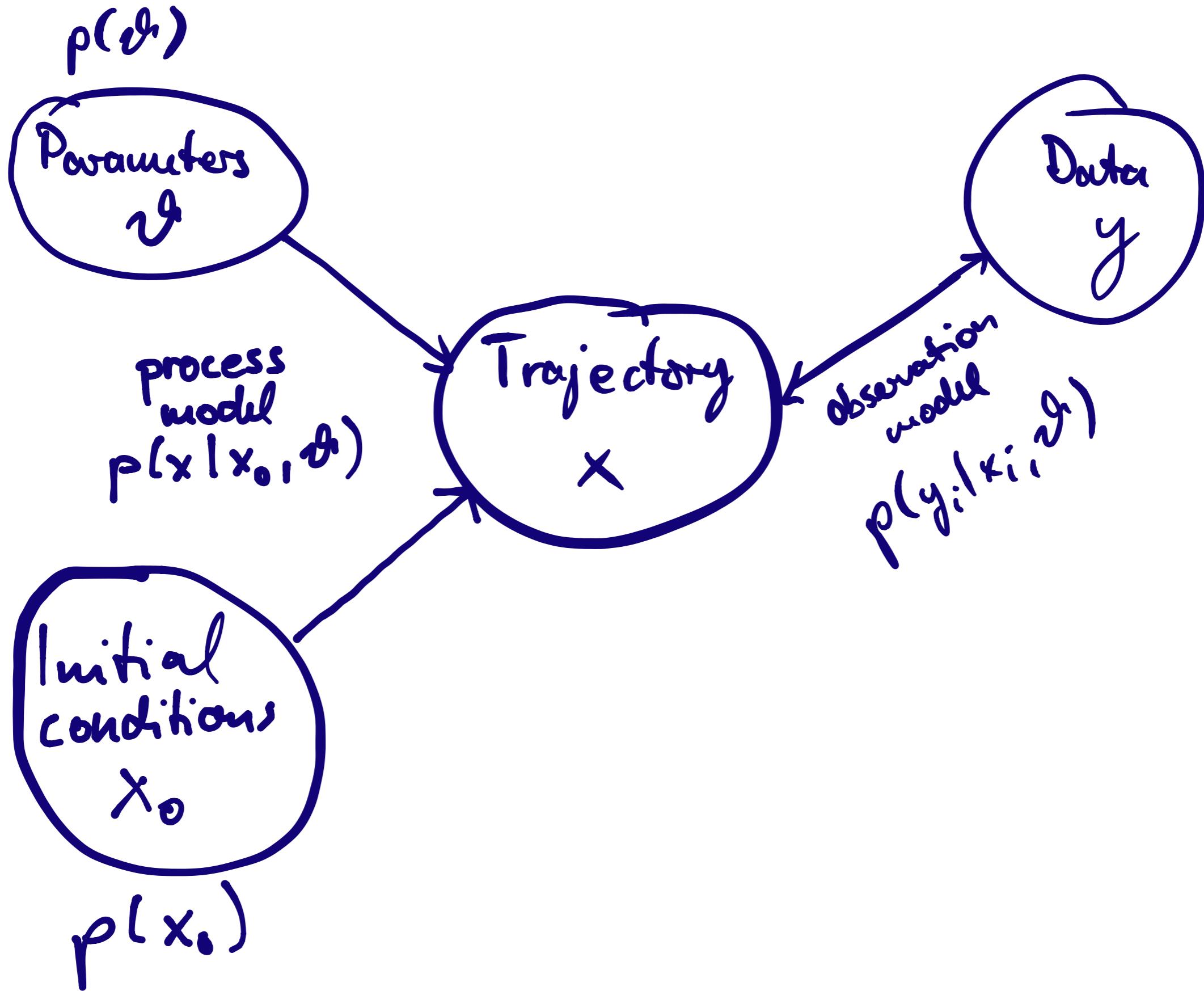
Provides a complete R interface to LibBi, a library for Bayesian inference (see <http://libbi.org> and [doi:10.18637/jss.v067.i10](https://doi.org/10.18637/jss.v067.i10) for more information). This includes functions for manipulating LibBi models, for reading and writing LibBi input/output files, for converting LibBi output to provide traces for use with the `coda` package, and for running LibBi from R.

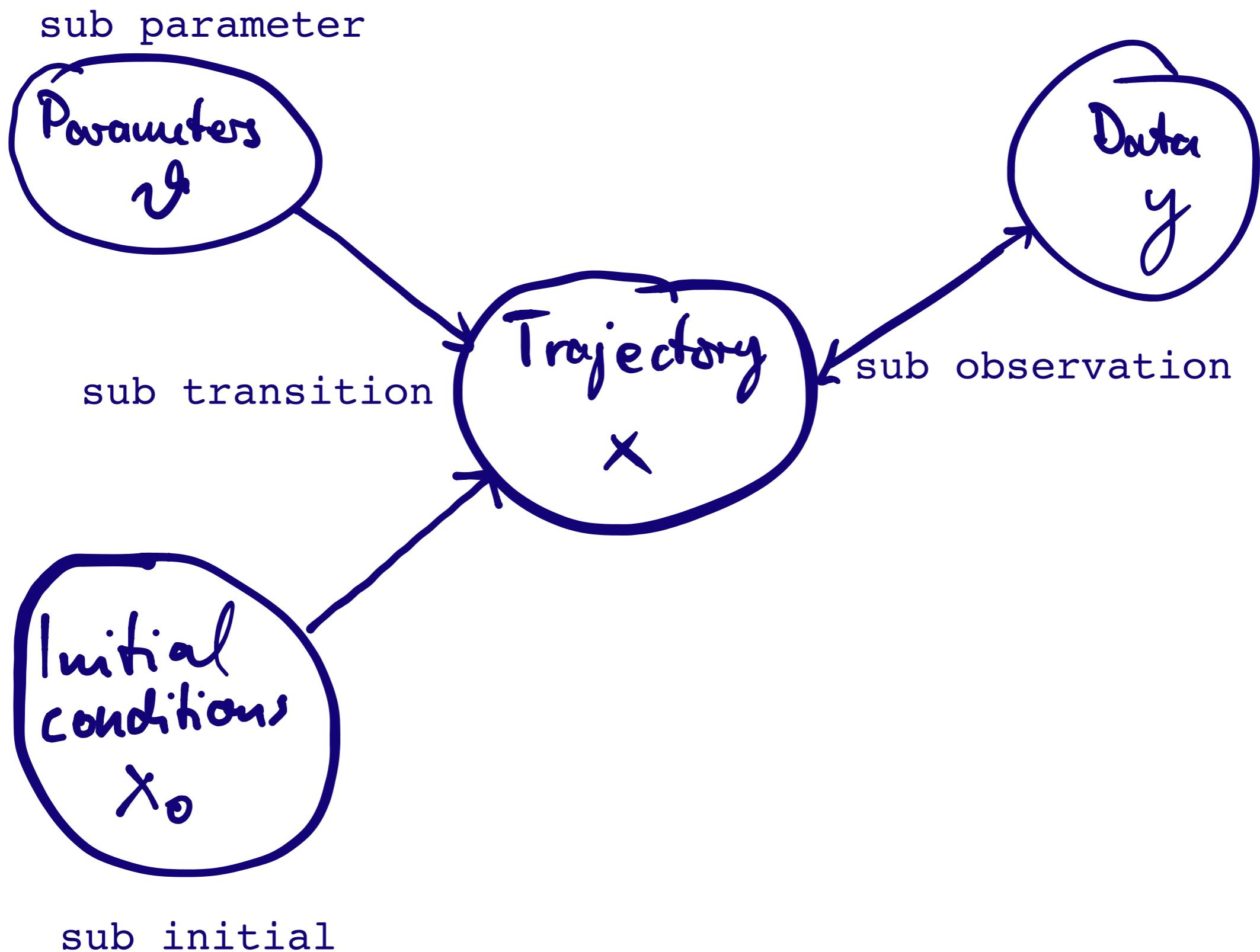
Version: 0.9.1
Imports: [ncdf4](#), [data.table](#), [reshape2](#)
Suggests: [coda](#), [R.rsp](#), [testthat](#), [stringi](#)
Published: 2018-05-14
Author: Pierre E. Jacob [aut], Anthony Lee [ctb], Lawrence M. Murray [ctb], Sebastian Funk [aut, cre]
Maintainer: Sebastian Funk <sebastian.funk@lshtm.ac.uk>
BugReports: <https://github.com/libbi/RBi/issues>
License: [GPL-3](#)
URL: <https://github.com/libbi/RBi>

pMCMC on GPUs with rbi/LibBi









```

model SIR_deterministic {
    const N = 1000; // population size
    const d_infection = 14; // duration of infection: 2 weeks

    state S, I, R; // susceptible, infectious, recovered

    obs Prevalence; // observations

    param R0; // basic reproduction number

    sub parameter {
        R0 ~ uniform(1, 3)
    }

    sub initial {
        S <- N - 1
        I <- 1
        R <- 0
    }

    sub transition { // daily time step
        inline beta = R0 / d_infection
        inline gamma = 1 / d_infection
        ode {
            dS/dt = - beta * S * I / N
            dI/dt = beta * S * I / N - gamma * I
            dR/dt = gamma * I
        }
    }

    sub observation {
        Prevalence ~ poisson(I)
    }
}

```

