CS-E5880 Modeling biological networks
Biological network structure selection: Bayesian model inference and comparison

Harri Lähdesmäki

Acknowledgement: Jukka Intosalmi, Juho Timonen

Department of Computer Science
Aalto University

February 1, 2022

Outline

- Differentiation of $CD4^+$ cells
- A recap of Bayesian inference
- Metropolis Hastings algorithm
- Population Markov chain Monte Carlo
- Bayesian model comparison
- Thermodynamic integration
- Reading (see references at the end):
  - Based on Chapter 9 from (Wilkinson, 2011)
  - Selected articles

Motivation

- Previous lecture: choose optimal parameter values (point estimates) for a deterministic biological network, given a fixed network model structure
- Today:
  - Bayesian inference to characterize uncertainty in parameters of a biological network model
  - Compare different biological network model structures, given experimental data

Differentiation of $CD4^+$ cells

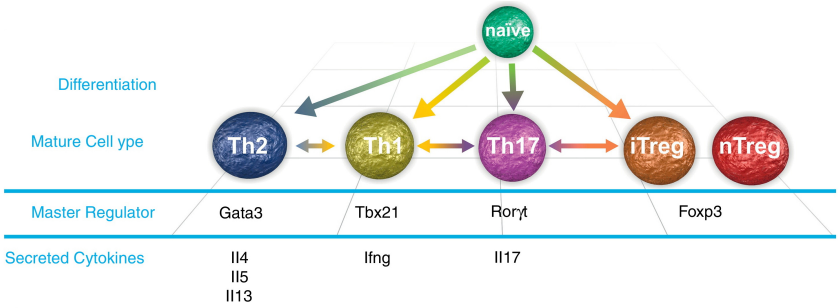


Figure: Mature $CD4^+$ subtypes are derived from naive $CD4^+$ cells (Hebenstreit et al., 2012).

T helper 17 (Th17) cell differentiation

- Key transcription factors: Rorγt and Stat3
- Cytokines: IL6 and TGFβ

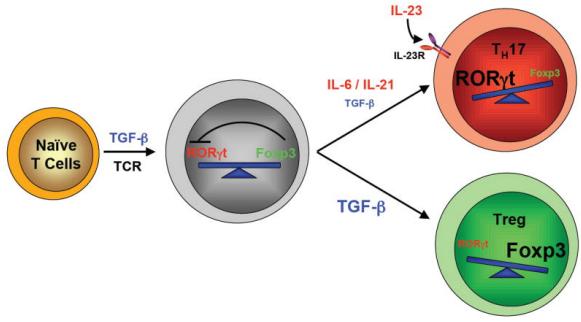
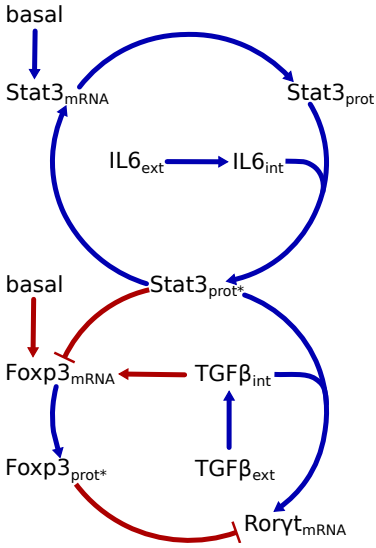


Figure: Th17 and Treg differentiation (Zhou and Littman, 2009).

Schematic model



- Use prior biological knowledge to construct an initial ODE model
  - Key genes involved in the process
  - Model structure
  - Parametric form of differential equations
- Three genes and two inducing cytokine signals
- Blue and red "connectors" are fixed; red connectors are hypothetical and will be tested against data later
- Mass-action kinetics

Ordinary differential equation system

$$
\begin{aligned}
\frac{d[\mathrm{IL6_{ext}}]}{dt} &= -\theta_1[\mathrm{IL6_{ext}}] \\
\frac{d[\mathrm{IL6_{int}}]}{dt} &= \theta_1[\mathrm{IL6_{ext}}] \\
\frac{d[\mathrm{STAT3_{mRNA}}]}{dt} &= \theta_2 + \theta_3[\mathrm{STAT3^*_{prot}}] - \theta_4[\mathrm{STAT3_{mRNA}}] \\
\frac{d[\mathrm{STAT3_{prot}}]}{dt} &= \theta_5[\mathrm{STAT3_{mRNA}}] - \theta_6[\mathrm{IL6_{int}}][\mathrm{STAT3_{prot}}] - \theta_7[\mathrm{STAT3_{prot}}] \\
\frac{d[\mathrm{STAT3^*_{prot}}]}{dt} &= \theta_6[\mathrm{IL6_{int}}][\mathrm{STAT3_{prot}}] - \theta_8[\mathrm{STAT3^*_{prot}}] \\
\frac{d[\mathrm{TGF}\beta_{\mathrm{ext}}]}{dt} &= -\theta_9[\mathrm{TGF}\beta_{\mathrm{ext}}] \\
\frac{d[\mathrm{TGF}\beta_{\mathrm{int}}]}{dt} &= \theta_9[\mathrm{TGF}\beta_{\mathrm{ext}}] \\
\frac{d[\mathrm{ROR}\gamma\mathrm{t_{mRNA}}]}{dt} &= \theta_{10}[\mathrm{TGF}\beta_{\mathrm{int}}][\mathrm{STAT3^*_{prot}}] - \theta_{11}[\mathrm{FOXP3^*_{prot}}][\mathrm{ROR}\gamma\mathrm{t_{mRNA}}] - \theta_{12}[\mathrm{ROR}\gamma\mathrm{t_{mRNA}}] \\
\frac{d[\mathrm{FOXP3_{mRNA}}]}{dt} &= \theta_{13} + \theta_{14}[\mathrm{TGF}\beta_{\mathrm{int}}] - \theta_{15}[\mathrm{STAT3^*_{prot}}][\mathrm{FOXP3_{mRNA}}] - \theta_{16}[\mathrm{FOXP3_{mRNA}}] \\
\frac{d[\mathrm{FOXP3^*_{prot}}]}{dt} &= \theta_{17}[\mathrm{FOXP3_{mRNA}}] - \theta_{18}[\mathrm{FOXP3^*_{prot}}]
\end{aligned}
$$

Experimental data: RNA-sequencing

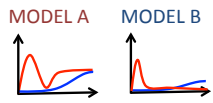
- Cells isolated from lymph nodes of C57BL/6 mice were cultured under Th17 cell polarization condition.
- A portion of the solution harvested at ten different time points (0, 0.5, 1, 2, 4, 6, 12, 24, 48, and 72 hours, three replicates at each time point)
- Gene expression measurements using RNA-sequencing

How to link count data with dynamic models?

Discrete read count data

?

Continuous-state dynamic models



Negative binomial (NB) model

- In the previous lecture, we introduced how ODE parameters can be learned by considering the parameter estimation as a statistical estimation problem
- But we focused primarily on the Gaussian likelihood model

Negative binomial (NB) model

- In the previous lecture, we introduced how ODE parameters can be learned by considering the parameter estimation as a statistical estimation problem
- But we focused primarily on the Gaussian likelihood model
- Negative binomial (NB) likelihood is commonly found as a good fit for RNA-seq data
- For a 1-dimensional case, if a dynamic model predicts the relative abundance of mRNA to be $x(t)$ at time t, then

$$y(t) \sim \mathrm{NB}(Lx(t), \phi),$$

where

  - $y(t)$ is the observed, discrete-valued read count (of mRNA abundance)
  - L is the library size (total number of sequencing reads from the experiment), and
  - ϕ can be taken as gene specific dispersion level

Statistical Inference

- d-dimensional data:

$$D = \{y^{(r)}(t_j) \in \mathbb{Z}_+^d, j = 1, \ldots, J, r = 1, \ldots, R\},$$

  where
  - d is the number of measured variables in our ODE model
  - J is the number of measurement time points, and
  - R denotes the number of replicates
- In our running example, $d = 3$, $J = 10$ and $R = 3$

Statistical Inference

- d-dimensional data:

$$D = \{y^{(r)}(t_j) \in \mathbb{Z}_+^d, j = 1, \ldots, J, r = 1, \ldots, R\},$$

  where
  - d is the number of measured variables in our ODE model
  - J is the number of measurement time points, and
  - R denotes the number of replicates
- In our running example, $d = 3$, $J = 10$ and $R = 3$
- NB Likelihood:

$$p(D|\theta, M) = \prod_{r,j,d} \text{NB}(L_{r,j} \cdot x_d(t_j), \phi)$$

  where M denotes the ODE model structure and $x_d(t)$ is the dth measured dimension of $x(t)$

Statistical Inference

- d-dimensional data:

$$D = \{y^{(r)}(t_j) \in \mathbb{Z}_+^d, j = 1, \ldots, J, r = 1, \ldots, R\},$$

  where
  - d is the number of measured variables in our ODE model
  - J is the number of measurement time points, and
  - R denotes the number of replicates
- In our running example, $d = 3$, $J = 10$ and $R = 3$
- NB Likelihood:

$$p(D|\theta, M) = \prod_{r,j,d} \text{NB}(L_{r,j} \cdot x_d(t_j), \phi)$$

  where M denotes the ODE model structure and $x_d(t)$ is the dth measured dimension of $x(t)$
- Instead of learning point estimates of the ODE model parameters, we would also like to characterize the uncertainty in the parameter value: $p(\theta|M, D)$

$\rightarrow$ Bayesian analysis

Bayesian inference

- It is often easy to understand the probability of some outcome X, conditional on various hypotheses H_i, $i = 1, \ldots, n$
- Assume H_i forms a partition: decomposition of a hypothesis space into a collection of mutually exclusive hypotheses that have positive probability
- We can typically compute likelihoods $P(X = x' | H_i)$

Bayesian inference

- It is often easy to understand the probability of some outcome X, conditional on various hypotheses H_i, $i = 1, \ldots, n$
- Assume H_i forms a partition: decomposition of a hypothesis space into a collection of mutually exclusive hypotheses that have positive probability
- We can typically compute likelihoods $P(X = x'|H_i)$
- When we observe $X = x$, we become interested in the probability of the hypotheses conditional on the outcome

$$P(H_i|X = x)$$

- We need to know prior probabilities of hypotheses, $P(H_i)$, $i = 1, \ldots, n$
- Bayes theorem provides us a coherent way of updating our prior beliefs about the hypothesis $P(H_i)$ to $P(H_i|X = x)$

Bayesian inference (2)

- Bayes theorem

$$\begin{aligned} P(H_i|X=x) &= \frac{P(X=x|H_i)P(H_i)}{P(X=x)} \\ &= \frac{P(X=x|H_i)P(H_i)}{\sum_{j=1}^{n} P(X=x|H_j)P(H_j)} \end{aligned}$$

- Probabilities $P(X=x|H_i)$ are likelihoods
  - A function of H_i for given fixed outcome $X=x$

Bayesian inference (2)

- Bayes theorem

$$\begin{aligned} P(H_i|X=x) &= \frac{P(X=x|H_i)P(H_i)}{P(X=x)} \\ &= \frac{P(X=x|H_i)P(H_i)}{\sum_{j=1}^{n} P(X=x|H_j)P(H_j)} \end{aligned}$$

- Probabilities $P(X=x|H_i)$ are likelihoods
  - A function of H_i for given fixed outcome $X=x$
- Extension to continuous case is straightforward
- Assume hypotheses are represented by a continuous parameter θ with a density $\pi(\theta)$, Bayes theorem becomes

$$\pi(\theta|X=x) = \frac{P(X=x|\theta)\pi(\theta)}{\int_{\theta\in\Theta} P(X=x|\theta')\pi(\theta')d\theta'}$$

Bayesian inference (3)

- Note that the denominator does not depend on θ
  - Constant of proportionality
- Bayes theorem in a simpler form

$$\pi(\theta|X = x) \propto P(X = x|\theta)\pi(\theta)$$

- The posterior is proportional to the prior times the likelihood
- See e.g. an example on page 198–199 in (Wilkinson, 2011)

Bayesian computation

- The previous discussion covers essentially everything (the very basics) that is needed for Bayesian inference
- The posterior is a conditional distribution for the parameters given the data
- Unfortunately, the posterior is often difficult to work with analytically

Bayesian computation

- The previous discussion covers essentially everything (the very basics) that is needed for Bayesian inference
- The posterior is a conditional distribution for the parameters given the data
- Unfortunately, the posterior is often difficult to work with analytically
- In the case of a "difficult" posterior, we can use
  - Numerical integration methods
  - Stochastic sampling methods

Markov chain Monte Carlo (MCMC)

- MCMC methods: a class of algorithms for sampling from a probability distribution
- Idea:
  - Construct a Markov chain that has the desired distribution as its equilibrium distribution
  - Simulate the Markov chain to obtain samples from the desired distribution

Markov chain Monte Carlo (MCMC)

- MCMC methods: a class of algorithms for sampling from a probability distribution
- Idea:
  - Construct a Markov chain that has the desired distribution as its equilibrium distribution
  - Simulate the Markov chain to obtain samples from the desired distribution
- MCMC algorithms
  - Gibbs sampling
  - Metropolis-Hasting
  - Slice sampling
  - Langevian dynamics
  - Hamiltonian Monte Carlo
  - Population MCMC

Metropolis-Hastings (MH) algorithm

- Assume $\pi(\theta)$ is the density of interest (e.g. parameter posterior $\pi(\theta) \triangleq \pi(\theta \mid D)$)
- Assume we have a transition kernel (also called proposal distribution) $q(\theta, \theta^*)$ which is easy to simulate but may or may not have $\pi(\theta)$ as its stationary distribution

Metropolis-Hastings (MH) algorithm

- Assume $\pi(\theta)$ is the density of interest (e.g. parameter posterior $\pi(\theta) \triangleq \pi(\theta \mid D)$)
- Assume we have a transition kernel (also called proposal distribution) $q(\theta, \theta^*)$ which is easy to simulate but may or may not have $\pi(\theta)$ as its stationary distribution
- The basic Idea of the MH algorithm is that we propose a move from the current state θ to a new state θ^* with a probability $q(\theta, \theta^*)$
- After proposing a move to θ^*, we need to decide whether to accept this proposal or not
- Accepting the new state is chosen probabilistically such that in the long-run the fraction of time spent in each state is proportional to $\pi(\theta)$
- If the new proposed state is accepted, then the new state is θ^*, otherwise the new state is the same as the current state θ (i.e., the Markov chain does not move to a new state)

Metropolis-Hastings algorithm (2)

- The Metropolis-Hastings algorithm
  1. Initialize iteration counter $i = 0$ and state of the chain $\theta^{(0)}$
  2. Generate a proposed value θ^* from the kernel $q(\theta^{(i)}, \theta^*)$
  3. Evaluate the acceptance probability of the proposed move

$$\alpha(\theta^{(i)}, \theta^*) = \min\left\{1, \frac{\pi(\theta^*)q(\theta^*, \theta^{(i)})}{\pi(\theta^{(i)})q(\theta^{(i)}, \theta^*)}\right\}$$

  4. Set $\theta^{(i+1)} = \theta^*$ with probability $\alpha(\theta, \theta^*)$; otherwise set $\theta^{(i+1)} = \theta^{(i)}$ (i.e., with probability $1 - \alpha(\theta, \theta^*)$)
  5. Set $i := i + 1$ and return to step 2

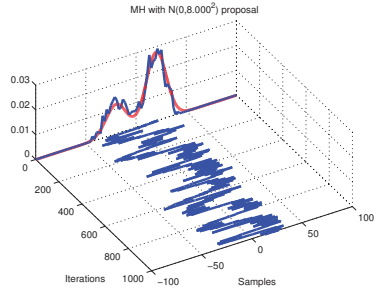
Metropolis-Hastings algorithm (2)

- The Metropolis-Hastings algorithm
  1. Initialize iteration counter $i = 0$ and state of the chain $\theta^{(0)}$
  2. Generate a proposed value θ^* from the kernel $q(\theta^{(i)}, \theta^*)$
  3. Evaluate the acceptance probability of the proposed move

$$\alpha(\theta^{(i)}, \theta^*) = \min\left\{1, \frac{\pi(\theta^*)q(\theta^*, \theta^{(i)})}{\pi(\theta^{(i)})q(\theta^{(i)}, \theta^*)}\right\}$$

  4. Set $\theta^{(i+1)} = \theta^*$ with probability $\alpha(\theta, \theta^*)$; otherwise set $\theta^{(i+1)} = \theta^{(i)}$ (i.e., with probability $1 - \alpha(\theta, \theta^*)$)
  5. Set $i := i + 1$ and return to step 2
- Note that the target density is used only via the ratio $\frac{\pi(\theta^*)}{\pi(\theta^{(i)})}$
- MH algorithm can be used even if we know only the un-normalized target density $\tilde{\pi}(\theta) = \frac{1}{Z}\pi(\theta)$ because $\frac{\pi(\theta^*)/Z}{\pi(\theta^{(i)})/Z} = \frac{\pi(\theta^*)}{\pi(\theta^{(i)})}$
- The above Markov chain is reversible and has stationary distribution $\pi(\cdot)$ regardless of the choice of $q(\cdot, \cdot)$ (assuming some technical conditions are satisfied) — proof page 213

Metropolis-Hastings example from (Murphy, 2012)



(c)

Figure 24.7 An example of the Metropolis Hastings algorithm for sampling from a mixture of two 1D Gaussians ($\boldsymbol{\mu} = (-20, 20)$, $\boldsymbol{\pi} = (0.3, 0.7)$, $\boldsymbol{\sigma} = (100, 100)$), using a Gaussian proposal with variances of $v \in \{1, 500, 8\}$. (a) When $v = 1$, the chain gets trapped near the starting state and fails to sample from the mode at $\mu = -20$. (b) When $v = 500$, the chain is very "sticky", so its effective sample size is low (as reflected by the rough histogram approximation at the end). (c) Using a variance of $v = 8$ is just right and leads to a good approximation of the true distribution (shown in red). Figure generated by `mcmcGmmDemo`. Based on code by Christophe Andrieu and Nando de Freitas.

SEIR example with control measures

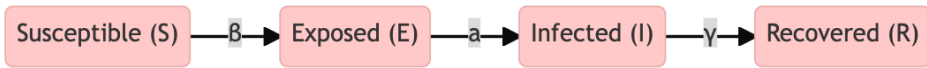
- Prepared by Juho Timonen
- Data from (Grinsztajn et a., 2021)
- In this example the proposal distribution is based on HMC sampler

The SEIR system

Population of N_{pop} people is divided in to Susceptible (S), Exposed (E), Infected (I), and Recovered (R) individuals. State of the system at time t is $\mathbf{y}(t) = [S(t), E(t), I(t), R(t)]^\top$. Disease transmission is modeled using a 4-dimensional ODE system

$$\begin{aligned}
\frac{\mathrm{d}S(t)}{\mathrm{d}t} &= -\beta S(t)\frac{I(t)}{N_{pop}}, \\
\frac{\mathrm{d}E(t)}{\mathrm{d}t} &= \beta S(t)\frac{I(t)}{N_{pop}} - aE(t) \\
\frac{\mathrm{d}I(t)}{\mathrm{d}t} &= aE(t) - \gamma I(t) \\
\frac{\mathrm{d}R(t)}{\mathrm{d}t} &= \gamma I(t),
\end{aligned}$$

where transmission rate $\beta > 0$, incubation rate $a > 0$, and recovery rate $\gamma > 0$ are (typically unknown) model parameters.

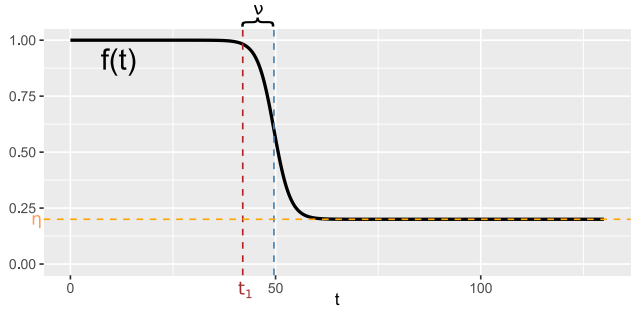


Control measures from (Grinsztajn et a., 2021)

Effect of control measures (lockdown, masks, less mobility) can be modeled by replacing β with $\beta^*(t) = \beta f(t)$. The forcing function $f(t)$ is

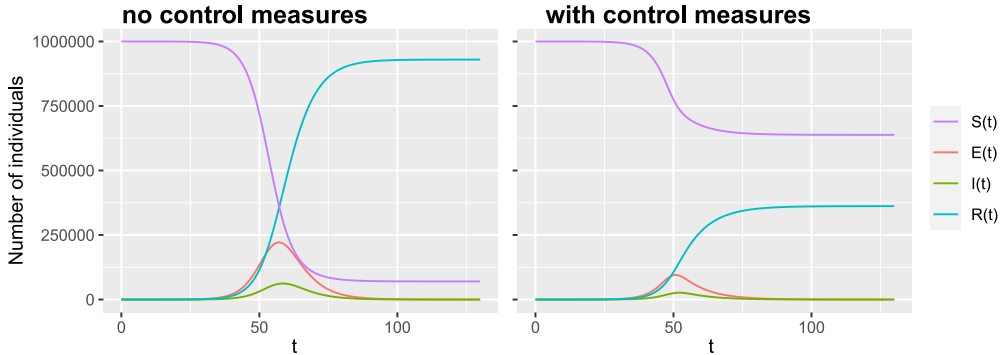
$$f(t) = \eta + \frac{1-\eta}{1+\exp\left(\xi(t-t_1-\nu)\right)} \tag{1}$$

and it has parameters: $1-\eta \in (0,1)$ is the effectiveness of the control measures, $\xi > 0$ is the slope of the decrease, t_1 is the known date of introduction of control measures, and $\nu > 0$ is a delay parameter.



ODE solution

Here is an example of the ODE solution $\mathbf{y}(t)$ when $N_{pop} = 10^6$ and initial state is $S(0) = N_{pop} - E(0) - I(0)$, $E(0) = I(0) = 1$ and $R(0) = 0$.



Used parameters values were $\beta = 2$, $a = 0.2$ and $\gamma = 0.7$ for both figures. In the right figure, the forcing function parameters were $\eta = 0.2$, $\nu = 7.6$, $t_1 = 42$ and $\xi = 0.5$ (same as previous slide).

Bayesian parameter inference

- In reality we don't know the parameter values $\boldsymbol{\theta} = \{\beta, a, \gamma, \eta, \nu, \xi\}$.
- They can be estimated from data $\mathcal{D}$ by applying Bayesian inference for a probabilistic model with posterior $p(\boldsymbol{\theta} \mid \mathcal{D}) \propto p(\mathcal{D} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta})$
- We can set reasonable priors $p(\boldsymbol{\theta})$ for the parameters, for example doctors might know what are likely/possible values for the disease incubation time $\frac{1}{a}$.
- In infectious disease modeling, we usually only have data $\mathcal{D}$ about how many new infections were reported each day, and we don't know how for example how many people are infected at given time t.
- $\rightarrow$ Problem: how to define likelihood $p(\mathcal{D} \mid \boldsymbol{\theta})$?

Modeling the reported number of new disease infections

- Assume we have measured $\mathcal{D}_t$, the reported number of new infections, on days $t = 1, \ldots, D$.
- One can define

$$p(\mathcal{D} \mid \boldsymbol{\theta}) = \prod_{t=1}^{D-1} \text{Negative-binomial}(\mathcal{D}_t | \rho \Delta(t), \phi) \tag{2}$$

  where $\Delta(t) = I(t+1) - I(t)$ is the modeled change in number of infections (incidence), $\rho \in (0, 1)$ is an additional reporting rate parameter, and ϕ is a noise magnitude parameter of the observation model.
- We can include ρ and ϕ in $\boldsymbol{\theta}$ and perform Bayesian inference jointly also for them.

Model fit using Covid-19 data

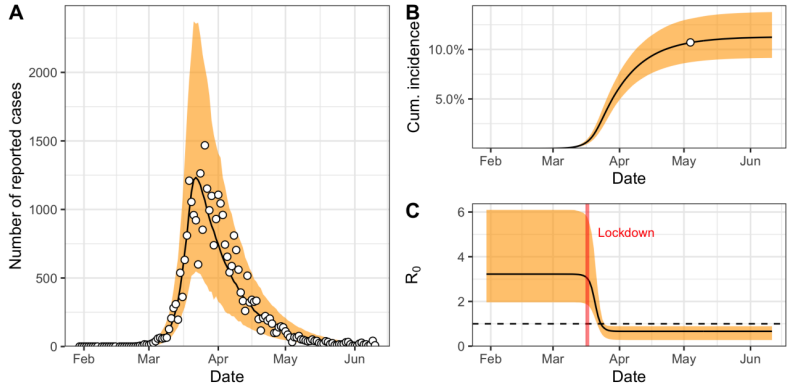
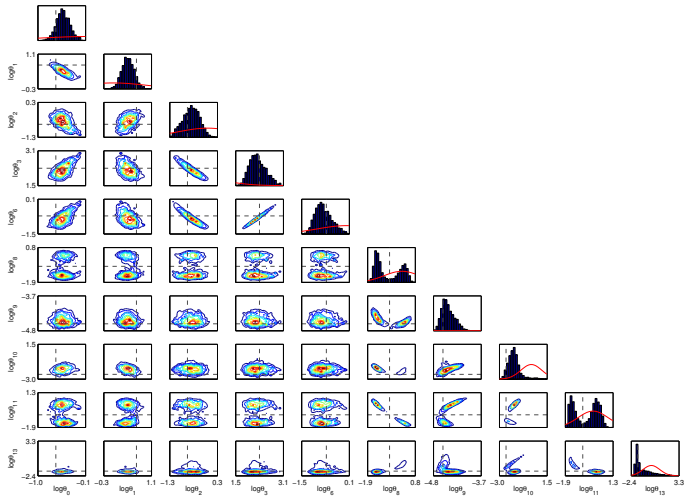


Figure: Posterior predictive distribution of a SEIR model with control measures and underreporting fitted to Covid-19 infections data from Switzerland 2020. Figure from (Grinsztajn et al., 2021). Note that in addition to reported cases (panel A) also serological test data (panel B) with binomial likelihood was used to fit this model. After model fitting, it was for example estimated that the effectiveness of control measures $1 - \eta$ is 73% with 95% credible interval of 53%-92%.

MCMC sampling example from (Chan et al., 2016)

- The MH is a general algorithm but can have difficulties in sampling highly multimodal posterior densities
- Parameter posteriors can have multimodal posteriors especially if the ODE model is misspecified: example from (Intosalmi et al., 2015)



Population Markov chain Monte Carlo

- Population Markov Chain Monte Carlo algorithm can be seen as an extension of the Metropolis-Hastings algorithm that has been developed to enable sampling from complex, multimodal distributions
- If the original target distribution $p(\theta|D)$ is complex / multimodal, it can be made smoother by raising the likelihood to a power of $\beta \in [0, 1]$

$$p_\beta(\theta|D) \propto p(D|\theta)^\beta p(\theta), \quad \beta \in [0, 1]$$

Population Markov chain Monte Carlo

- Population Markov Chain Monte Carlo algorithm can be seen as an extension of the Metropolis-Hastings algorithm that has been developed to enable sampling from complex, multimodal distributions
- If the original target distribution $p(\theta|D)$ is complex / multimodal, it can be made smoother by raising the likelihood to a power of $\beta \in [0, 1]$

$$p_\beta(\theta|D) \propto p(D|\theta)^\beta p(\theta), \quad \beta \in [0, 1]$$

- When $\beta = 0$, then $p_\beta(\theta|D) = p(\theta)$, i.e., the prior
- When $\beta = 1$, then $p_\beta(\theta|D) \propto p(D|\theta)p(\theta)$, i.e., the posterior
- When β changes from 0 to 1, $p_\beta(\theta|D)$ changes from (smooth) prior to more complex posterior

Population Markov chain Monte Carlo (2)

- Select a collection of "temperatures" $0 = \beta_1 < \ldots < \beta_{N_\beta} = 1$ (in general β_1 can be larger than 0)
- Define a separate parameter vector θ_{β_i} for each β_i
- Define a distribution that is the product of $p_\beta(\theta_{\beta_i}|D)$ across all temperatures

$$
\begin{aligned}
p(\theta_{\beta_1}, \ldots, \theta_{\beta_{N_\beta}}|D) &= \prod_{i=1}^{N_\beta} p_\beta(\theta_{\beta_i}|D) \\
&= \underbrace{p(\theta_{\beta_1})}_{\text{prior}} \prod_{i=2}^{N_\beta - 1} p_\beta(\theta_{\beta_i}|D) \underbrace{p(\theta_{\beta_{N_\beta}}|D)}_{\text{posterior}}
\end{aligned}
$$

Population Markov chain Monte Carlo (2)

- Select a collection of "temperatures" $0 = \beta_1 < \ldots < \beta_{N_\beta} = 1$ (in general β_1 can be larger than 0)
- Define a separate parameter vector θ_{β_i} for each β_i
- Define a distribution that is the product of $p_\beta(\theta_{\beta_i}|D)$ across all temperatures

$$
\begin{aligned}
p(\theta_{\beta_1}, \ldots, \theta_{\beta_{N_\beta}}|D) &= \prod_{i=1}^{N_\beta} p_\beta(\theta_{\beta_i}|D) \\
&= \underbrace{p(\theta_{\beta_1})}_{\text{prior}} \prod_{i=2}^{N_\beta - 1} p_\beta(\theta_{\beta_i}|D) \underbrace{p(\theta_{\beta_{N_\beta}}|D)}_{\text{posterior}}
\end{aligned}
$$

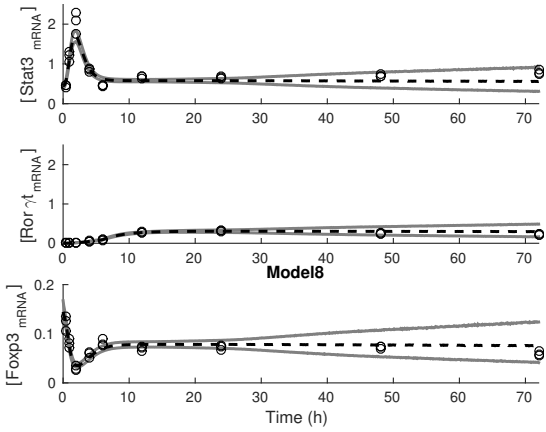
- Population MCMC:
  - Apply a modified MH algorithm to this product distribution that allows the parallel chains to interact

Population Markov chain Monte Carlo (3)

- Algorithm: Run N_β many MCMC chains in parallel, each with the corresponding proposal distribution $q(\theta_{\beta_i}, \theta^*_{\beta_i})$ and target distribution $p_\beta(\theta_{\beta_i}|D)$, but allow the parallel chains to interact and share information
  1. With a probability p_{MH} the algorithm takes the standard MH step for one of the components θ_{β_i} using the corresponding distribution $p_\beta(\theta_{\beta_i}|D)$
  2. With a probability $1 - p_{\mathrm{MH}}$ the algorithm proposes to swap the states for randomly chosen neighboring temperatures θ_{β_i} and $\theta_{\beta_{i+1}}$; accept the swap with the MH probability using $p(\theta_{\beta_1}, \ldots, \theta_{\beta_{N_\beta}}|D)$ where θ_{β_i} and $\theta_{\beta_{i+1}}$ are swapped
- Run the algorithm until convergence: the samples for $\theta_{\beta_{N_\beta}}$, where $\beta_{N_\beta} = 1$ are samples from the desired posterior

Population Markov chain Monte Carlo (4)

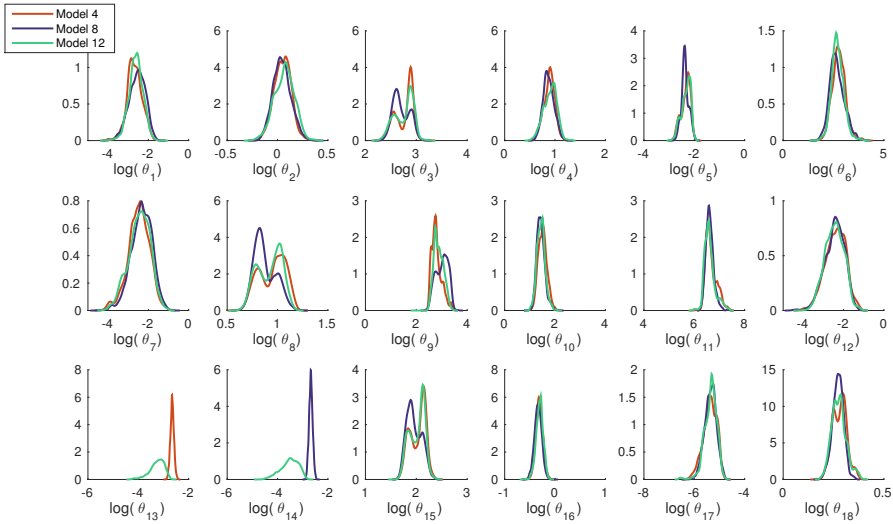
- Data and model fit to the RNA-seq data (Intosalmi et al., 2015)



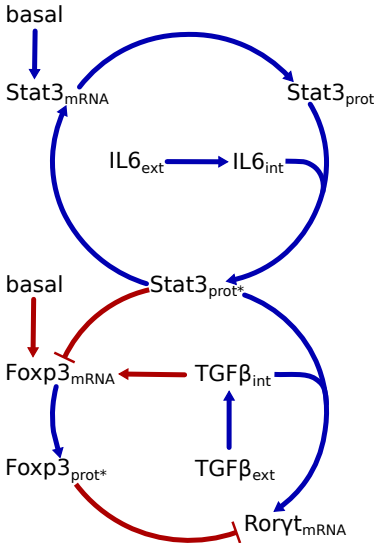
(h) Model 8

Population Markov chain Monte Carlo (5)

- Parameter posteriors for the ODE model (Intosalmi et al., 2015)

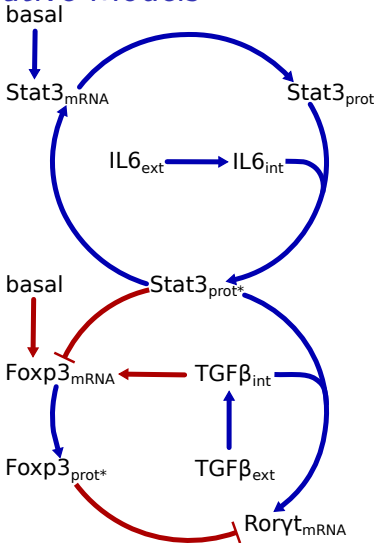


Schematic Model



- Often we are uncertain about the model structure
- Blue connectors are fixed (assumed to be known)
- Red connectors are hypothetical and will be tested against data

12 Alternative Models



	Basal expression for FOXP3	FOXP3 activation by TGFβ	RORγt inhibition by FOXP3	FOXP3 inhibition by STAT3
Model 1	×	–	–	–
Model 2	×	–	–	×
Model 3	×	–	×	–
Model 4	×	–	×	×
Model 5	–	×	–	–
Model 6	–	×	–	×
Model 7	–	×	×	–
Model 8	–	×	×	×
Model 9	×	×	–	–
Model 10	×	×	–	×
Model 11	×	×	×	–
Model 12	×	×	×	×

Ordinary Differential Equation System

$$
\begin{aligned}
\frac{d[\mathrm{IL6_{ext}}]}{dt} &= -\theta_1[\mathrm{IL6_{ext}}] \\
\frac{d[\mathrm{IL6_{int}}]}{dt} &= \theta_1[\mathrm{IL6_{ext}}] \\
\frac{d[\mathrm{STAT3_{mRNA}}]}{dt} &= \theta_2 + \theta_3[\mathrm{STAT3^*_{prot}}] - \theta_4[\mathrm{STAT3_{mRNA}}] \\
\frac{d[\mathrm{STAT3_{prot}}]}{dt} &= \theta_5[\mathrm{STAT3_{mRNA}}] - \theta_6[\mathrm{IL6_{int}}][\mathrm{STAT3_{prot}}] - \theta_7[\mathrm{STAT3_{prot}}] \\
\frac{d[\mathrm{STAT3^*_{prot}}]}{dt} &= \theta_6[\mathrm{IL6_{int}}][\mathrm{STAT3_{prot}}] - \theta_8[\mathrm{STAT3^*_{prot}}] \\
\frac{d[\mathrm{TGF}\beta_{\mathrm{ext}}]}{dt} &= -\theta_9[\mathrm{TGF}\beta_{\mathrm{ext}}] \\
\frac{d[\mathrm{TGF}\beta_{\mathrm{int}}]}{dt} &= \theta_9[\mathrm{TGF}\beta_{\mathrm{ext}}] \\
\frac{d[\mathrm{ROR}\gamma\mathrm{t_{mRNA}}]}{dt} &= \theta_{10}[\mathrm{TGF}\beta_{\mathrm{int}}][\mathrm{STAT3^*_{prot}}] - \theta_{11}[\mathrm{FOXP3^*_{prot}}][\mathrm{ROR}\gamma\mathrm{t_{mRNA}}] - \theta_{12}[\mathrm{ROR}\gamma\mathrm{t_{mRNA}}] \\
\frac{d[\mathrm{FOXP3_{mRNA}}]}{dt} &= \theta_{13} + \theta_{14}[\mathrm{TGF}\beta_{\mathrm{int}}] - \theta_{15}[\mathrm{STAT3^*_{prot}}][\mathrm{FOXP3_{mRNA}}] - \theta_{16}[\mathrm{FOXP3_{mRNA}}] \\
\frac{d[\mathrm{FOXP3^*_{prot}}]}{dt} &= \theta_{17}[\mathrm{FOXP3_{mRNA}}] - \theta_{18}[\mathrm{FOXP3^*_{prot}}]
\end{aligned}
$$

Alternative models can be obtained by setting the corresponding parameters to zero.

Posterior distribution over alternative models

- We would like to evaluate different model structures quantitatively
- The posterior distribution over the models M is again obtained by Bayes rule

$$p(M|D) = \frac{p(D|M)\pi(M)}{\sum_{M'} p(D|M')\pi(M')},$$

  where π is the prior distribution over the models.
- To determine the probability of a model, we need to compute the marginal likelihood

$$p(D|M) = \int p(D|\theta, M)p(\theta|M)d\theta$$

Thermodynamic integration

- It can be shown that the log marginal likelihood for a given model M can be estimated using the so-called thermodynamic integration

$$\ln(p(D|M)) = \int_0^1 \left[\int \ln(p(D|\theta, M)) p_\beta(\theta|D, M) d\theta \right] d\beta$$

Thermodynamic integration

- It can be shown that the log marginal likelihood for a given model M can be estimated using the so-called thermodynamic integration

$$\ln(p(D|M)) = \int_0^1 \left[\int \ln(p(D|\theta, M)) p_\beta(\theta|D, M) d\theta \right] d\beta$$

- Numerical estimation: first obtain a Monte Carlo estimate of the inner integral as

$$\int \ln(p(D|\theta, M)) p_\beta(\theta|D, M) d\theta \approx I_{\beta_i} = \frac{1}{N_s} \sum_{j=1}^{N_s} \ln(p(D|\theta_{\beta_i}^{(j)}, M)), \quad \theta_{\beta_i}^{(j)} \sim p_{\beta_i}(\theta_{\beta_i}|D, M),$$

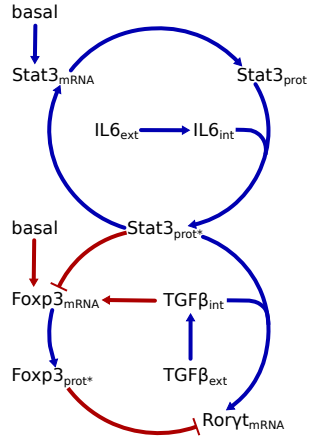
and then approximate the outer integral using numerical integration

$$\ln(p(D|M)) \approx \sum_{i=2}^{N_\beta} (\beta_i - \beta_{i-1}) \left(\frac{I_{\beta_i} + I_{\beta_{i-1}}}{2} \right),$$

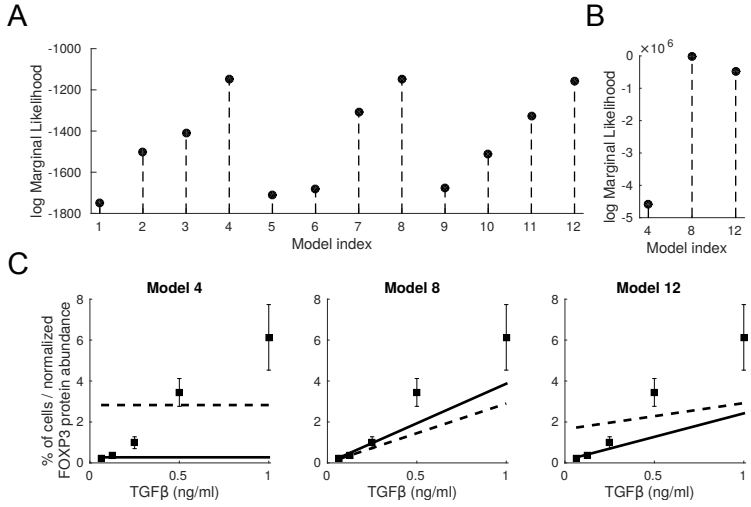
where and N_β and N_s are the number of temperatures and the number of samples from the population MCMC

Estimated logarithmic marginal likelihoods

	Basal expression for FOXP3	FOXP3 activation by TGFβ	**RORγt inhibition by FOXP3**	**FOXP3 inhibition by STAT3**	$\ln(\widehat{p(D \mid M_i)})$
Model 1	×	–	–	–	−1750
Model 2	×	–	–	×	−1502
Model 3	×	–	×	–	−1410
Model 4	×	–	×	×	−1146
Model 5	–	×	–	–	−1708
Model 6	–	×	–	×	−1680
Model 7	–	×	×	–	−1309
Model 8	–	×	×	×	−1149
Model 9	×	×	–	–	−1678
Model 10	×	×	–	×	−1513
Model 11	×	×	×	–	−1327
Model 12	×	×	×	×	−1156



Experimental validation approves Model 8



References

- Chan YH, Intosalmi J, Rautio S, Lähdesmäki H, A subpopulation model to analyze heterogeneous cell differentiation dynamics, Bioinformatics, Vol. 32, No. 21, pp. 3306-3313, 2016.
- Darren J. Wilkinson, *Stochastic Modelling for Systems Biology*, Chapman & Hall/CRC, 2011
- Friel N., Pettitt A.N., Marginal likelihood estimation via power posteriors. J. R. Stat. Soc. B 70(3), 589-607, 2008.
- Leo Grinsztajn et al., "Bayesian workflow for disease transmission modeling in Stan". Statistics in Medicine, 40.27 (2021), pp. 6209?6234. `https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.9164`.
- Hebenstreit et al., Current Opinion in Cell Biology, 2012
- Intosalmi J, Ahlfors H, Rautio S, Mannerström H, Chen ZJ, Lahesmaa R, Stockinger B, Lähdesmäki H, Analyzing Th17 cell differentiation dynamics using a novel integrative modeling framework for time-course RNA sequencing data, BMC Systems Biology, Vol. 9, No. 81, 2015.
- Murphy K (2012) Machine learning: a probabilistic perspective, MIT Press.
- Zhou and Littman, Current Opinion in Immunology, 2009