CS-E5885 Modeling biological networks
Parameter estimation for biological networks

Harri Lähdesmäki

Acknowledgement: Jukka Intosalmi, Henrik Mannerström

Department of Computer Science
Aalto University

January 26, 2024

Outline

- ODE model calibration as an optimization problem
- Local gradient based optimization
- Sensitivity equations
- Practical considerations
- ODE model calibration as a statistical estimation problem
- Identifiability

- A collection of articles covering the topic listed at the end for additional reading

Parameter estimation

Parameter estimation is an important part of biological network modeling.
Here we focus on ODE models but other models have parameters too.
Why parameter estimation?

- Testing if a hypothetical model can produce observed dynamics
- Estimation of the unknown reaction rates and initial values
- Model construction in a data-driven manner
- Experimental data is typically noisy, so many parameter values can fit the data reasonably well

Parameter estimation as an optimization problem (1)

- Let us consider a one dimensional initial value problem

$$\frac{dx}{dt} = f(x, \boldsymbol{\theta}), \quad x(0) = x_0,$$

  where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)$ is a parameter vector.
- The system has the solution $x(t, \boldsymbol{\theta})$.
- The solution can be obtained numerically e.g. by the Euler or Runge-Kutta methods
- The observed values of $x(t, \boldsymbol{\theta})$ are denoted by

$$y(t_i), \quad i = 1, \ldots, n$$

  where t_i are the measurement times.
- Observed values can be corrupted by noise

Parameter estimation as an optimization problem (2)

- In optimization setting, our goal is to minimize a distance between the solution of the ODE model and data
- For example, we may wish to minimize the sum-of-squares objective function

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{n} (y(t_i) - x(t_i, \boldsymbol{\theta}))^2$$

- For ODE models, closed-form solutions are only rarely available and numerical optimization techniques need to be used.
- Typically a non-convex optimization problem

Optimization techniques

Global and local optimization techniques

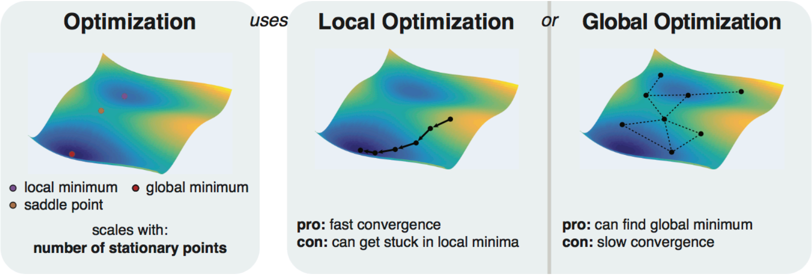
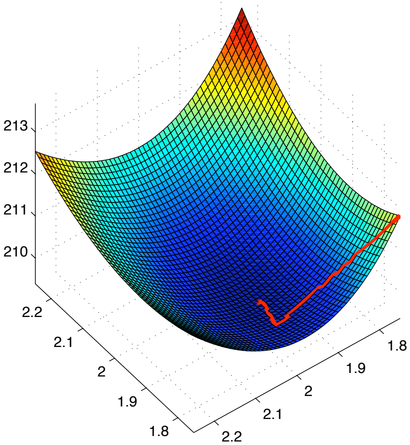


Figure: Illustration of optimization problem (from Fröchlich et al., 2017)

- In this lecture, we will concentrate on deterministic local optimization
- Extentions to global optimization through multistart approach

Gradient based optimization

- Search for the nearest (local) minimum of the objective function following the direction of the negative gradient



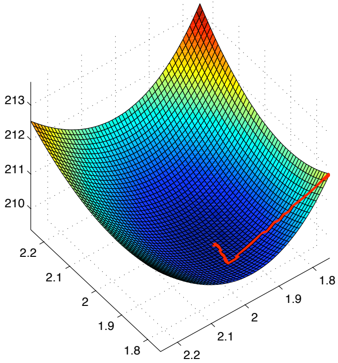
Gradient descent method

1. Set $k = 0$ and define initial parameter values $\boldsymbol{\theta}_k$ as well as step size (learning rate) r
2. Evaluate the objective function $L(\boldsymbol{\theta}_k)$
3. Compute the gradient of the objective function $\nabla L(\boldsymbol{\theta}) = \left(\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_1}, \ldots, \frac{\partial L(\boldsymbol{\theta})}{\partial \theta_d}\right)$ at $\boldsymbol{\theta}_k$, $\nabla L(\boldsymbol{\theta}_k)$

[4.] Optional line search: solve for the $\hat{\mathrm{r}}$ minimizing $L(\boldsymbol{\theta}_k - \mathrm{r} \cdot \nabla L(\boldsymbol{\theta}_k))$, set $r := \hat{\mathrm{r}}$

5. Set $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - r \cdot \nabla L(\boldsymbol{\theta}_k)^T$
6. Compare the current objective function value with the previous value;
   - if $L(\boldsymbol{\theta}_{k+1}) > L(\boldsymbol{\theta}_k)$ or
   - if $||\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k|| < \epsilon$

   then the solution is $\boldsymbol{\theta}_k$
7. Otherwise, set $k := k + 1$ and go to step 2



Differentiation of the objective function

- To obtain the gradient, we need to differentiate the objective function

$$\begin{aligned} \frac{\partial L(\boldsymbol{\theta})}{\partial \theta_j} &= \frac{\partial}{\partial \theta_j} \sum_{i=1}^{n} (y(t_i) - x(t_i, \boldsymbol{\theta}))^2 \\ &= \sum_{i=1}^{n} -2(y(t_i) - x(t_i, \boldsymbol{\theta})) \frac{\partial x(t_i, \boldsymbol{\theta})}{\partial \theta_j} \end{aligned}$$

Differentiation of the objective function

- To obtain the gradient, we need to differentiate the objective function

$$
\begin{aligned}
\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_j} &= \frac{\partial}{\partial \theta_j} \sum_{i=1}^{n} (y(t_i) - x(t_i, \boldsymbol{\theta}))^2 \\
&= \sum_{i=1}^{n} -2(y(t_i) - x(t_i, \boldsymbol{\theta})) \frac{\partial x(t_i, \boldsymbol{\theta})}{\partial \theta_j}
\end{aligned}
$$

- The gradient depends on $\frac{\partial x(t_i, \boldsymbol{\theta})}{\partial \theta_j}$, $i = 1, \ldots, n; j = 1, \ldots, d$
- Note that $x(t_i, \boldsymbol{\theta})$ is a function (infinite dimensional)
- Three approaches:
  1. Finite differences approximation
  2. Sensitivity equations (forward accumulation)
  3. Adjoint method (reverse accumulation)

Finite differences approximation for $\frac{\partial x(t_i,\boldsymbol{\theta})}{\partial \theta_j}$

- The approximation is computed by perturbing the parameters

$$\begin{aligned}\frac{\partial x(t_i,\boldsymbol{\theta})}{\partial \theta_j} &= \lim_{\varepsilon\to 0} \frac{x(t_i,\boldsymbol{\theta}) - x(t_i,\boldsymbol{\theta}+\varepsilon \mathrm{e}_j)}{\varepsilon} \\ &\approx \frac{x(t_i,\boldsymbol{\theta}) - x(t_i,\boldsymbol{\theta}+h \mathrm{e}_j)}{h}\end{aligned}$$

  where e_j is the jth unit vector and h is a sufficiently small constant.

- Easy to implement numerically, but. . .
  - Requires several numerical solutions for the ODE system
  - No generic way of choosing the constant h
  - In general finite differences approach may result in a poor approximation
  - Better results can be obtained using the sensitivity equations

Sensitivity equations

- We can take a so-called total derivative of the ODE system w.r.t. θ_j

$$\frac{d}{d\theta_j}\frac{dx}{dt} = \frac{d}{d\theta_j} f(x, \boldsymbol{\theta})$$

Sensitivity equations

- We can take a so-called total derivative of the ODE system w.r.t. θ_j

$$\frac{d}{d\theta_j}\frac{dx}{dt} = \frac{d}{d\theta_j} f(x, \boldsymbol{\theta})$$

- Recall that the derivative of a composite function $f(g(x))$ is $f'(g(x))g'(x)$
- Here x depends on $\boldsymbol{\theta}$ (and time), $x(\boldsymbol{\theta})$, so

$$f(x(\boldsymbol{\theta}), \boldsymbol{\theta}) = f(\ \underbrace{(x(\boldsymbol{\theta}), \boldsymbol{\theta})}_{\text{function } g(\boldsymbol{\theta})}\ )$$

Sensitivity equations

- The total derivative is thus

$$\frac{d}{d\theta_j}\frac{dx}{dt} \quad = \quad \frac{d}{d\theta_j}f(x(\boldsymbol{\theta}),\boldsymbol{\theta}) = f'(\underbrace{(x(\boldsymbol{\theta}),\boldsymbol{\theta})}_{\text{function } g(\boldsymbol{\theta})})\underbrace{\left(\frac{dx(\boldsymbol{\theta})}{d\theta_j}\;\;\frac{d\boldsymbol{\theta}}{d\theta_j}\right)^T}_{\text{derivative } g'(\boldsymbol{\theta})}$$

Sensitivity equations

- The total derivative is thus

$$
\begin{aligned}
\frac{d}{d\theta_j}\frac{dx}{dt} &= \frac{d}{d\theta_j} f(x(\boldsymbol{\theta}), \boldsymbol{\theta}) = f'(\underbrace{(x(\boldsymbol{\theta}), \boldsymbol{\theta})}_{\text{function } g(\boldsymbol{\theta})}) \underbrace{\left(\frac{dx(\boldsymbol{\theta})}{d\theta_j} \;\; \frac{d\boldsymbol{\theta}}{d\theta_j}\right)^T}_{\text{derivative } g'(\boldsymbol{\theta})} \\
&= \left(\frac{\partial f(x, \boldsymbol{\theta})}{\partial x} \;\; \frac{\partial f(x, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right) \left(\frac{dx}{d\theta_j} \;\; \frac{d\boldsymbol{\theta}}{d\theta_j}\right)^T \\
&= \frac{\partial f(x, \boldsymbol{\theta})}{\partial x}\frac{dx}{d\theta_j} + \frac{\partial f(x, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\frac{d\boldsymbol{\theta}}{d\theta_j} \\
&= \frac{\partial f(x, \boldsymbol{\theta})}{\partial x}\frac{dx}{d\theta_j} + \frac{\partial f(x, \boldsymbol{\theta})}{\partial \theta_j}
\end{aligned}
$$

(note that we change the notation from $x(\boldsymbol{\theta})$ back to x after the first line)

Sensitivity equations

- The total derivative

$$\frac{d}{d\theta_j}\frac{dx}{dt} = \frac{d}{dt}\frac{dx}{d\theta_j} = \frac{\partial f(x,\boldsymbol{\theta})}{\partial x}\frac{dx}{d\theta_j} + \frac{\partial f(x,\boldsymbol{\theta})}{\partial \theta_j}$$

- Let us define the so-called sensitivity w.r.t. the jth parameter as $s_j(t) = \frac{\partial x(t,\boldsymbol{\theta})}{\partial \theta_j}$
- The sensitivity equation can then be written as

$$\frac{ds_j}{dt} = \frac{\partial f(x,\boldsymbol{\theta})}{\partial x}s_j + \frac{\partial f(x,\boldsymbol{\theta})}{\partial \theta_j}$$

- This is another differential equation for $s_j(t)$

Solving the sensitivities

- The sensitivities and the original ODE model form a coupled system and they are solved simultaneously using numerical solvers.
- The accuracy of the computed sensitivities depends only on the numerical error of the ODE solver!

Comparing the finite differences and sensitivity equation approaches

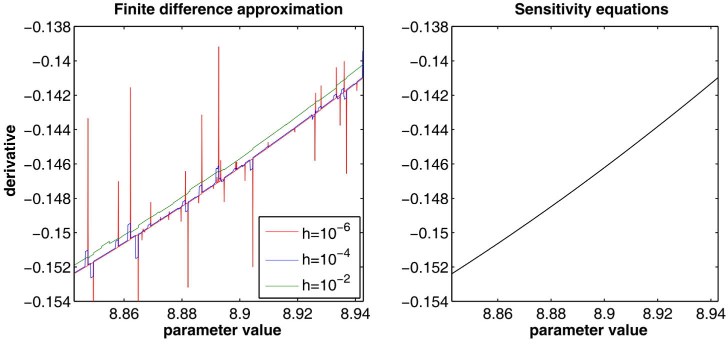


Figure: From Raue et al., 2013

Example of sensitivity equation construction (1)

- Let us consider the initial value problem

$$\frac{dx}{dt} = \underbrace{\alpha + \theta_1 \frac{x}{x+K} - \theta_2 x}_{f(x,\boldsymbol{\theta})}, \quad x(0) = x_0$$

  where θ_1 and θ_2 are unknown rate parameters.

- We have

$$\begin{aligned} \frac{\partial f(x,\boldsymbol{\theta})}{\partial x} &= \theta_1 \frac{K}{(x+K)^2} - \theta_2 \\ \frac{\partial f(x,\boldsymbol{\theta})}{\partial \theta_1} &= \frac{x}{x+K} \\ \frac{\partial f(x,\boldsymbol{\theta})}{\partial \theta_2} &= -x. \end{aligned}$$

Example of sensitivity equation construction (2)

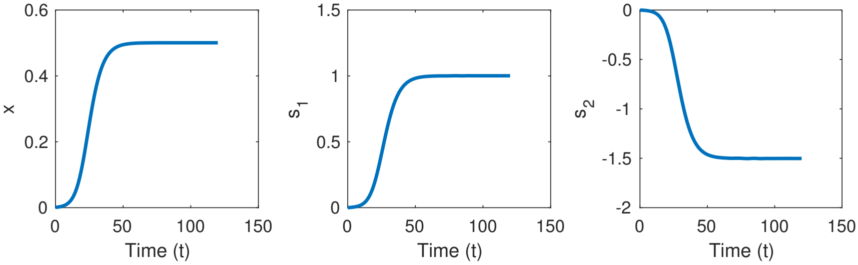
- The sensitivity equations together with the actual ODE model form the following ODE system

$$
\begin{aligned}
\frac{dx}{dt} &= \alpha + \theta_1 \frac{x}{x+K} - \theta_2 x \\
\frac{ds_1}{dt} &= \left(\theta_1 \frac{K}{(x+K)^2} - \theta_2\right) s_1 + \frac{x}{x+K} \\
\frac{ds_2}{dt} &= \left(\theta_1 \frac{K}{(x+K)^2} - \theta_2\right) s_2 - x
\end{aligned}
$$

with $x(0) = x_0 = 0$ and $s_1(0) = s_2(0) = 0$.

Example of sensitivity equation construction (3)

- And the system can be solved.



- The final gradient computation via the derivative of the objective function

$$\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_j} = \sum_{i=1}^{n} -2(y(t_i) - x(t_i, \boldsymbol{\theta}))\frac{\partial x(t_i, \boldsymbol{\theta})}{\partial \theta_j}$$

Sensitivity equation in the general form

- Let us consider an m-dimensional initial value problem ($\mathrm{x} \in \mathbb{R}^m$, $\boldsymbol{\theta} \in \mathbb{R}^d$, $\mathrm{f}(\cdot,\cdot) : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}^m$)

$$\frac{d\mathrm{x}}{dt} = \mathrm{f}(\mathrm{x}, \boldsymbol{\theta}), \quad \mathrm{x}(0) = \mathrm{x}_0,$$

  where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)^T$ is a parameter vector.

- The sensitivity equations can be expressed in the general form

$$\underbrace{\frac{d}{dt}\frac{d\mathrm{x}}{d\boldsymbol{\theta}}}_{\in\mathbb{R}^{m\times d}} = \underbrace{\frac{\partial \mathrm{f}(\mathrm{x},\boldsymbol{\theta})}{\partial \mathrm{x}}}_{\in\mathbb{R}^{m\times m}} \underbrace{\frac{d\mathrm{x}}{d\boldsymbol{\theta}}}_{\in\mathbb{R}^{m\times d}} + \underbrace{\frac{\partial \mathrm{f}(\mathrm{x},\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}}_{\in\mathbb{R}^{m\times d}} \quad \text{or} \quad \frac{d\mathrm{S}}{dt} = \frac{\partial \mathrm{f}(\mathrm{x},\boldsymbol{\theta})}{\partial \mathrm{x}}\mathrm{S} + \frac{\partial \mathrm{f}(\mathrm{x},\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$$

  and solved together with the original ODE shown above

Adjoint method

- The original problem of solving $x(t, \theta)$ forward in time is transformed into an *adjoint problem* of solving $\lambda(t, \theta)$ backward in time.
- Integrate the original problem $x(t, \theta)$ forward from t_1 to t_n to obtain the trajectory $x(t_1, \theta), \ldots, x(t_n, \theta)$.
- Integrate the adjoint problem $\lambda(t, \theta)$ backward from t_n to t_1 to obtain the trajectory $\lambda(t_1, \theta), \ldots, \lambda(t_n, \theta)$.
- Extract the derivatives from the adjoint solution.

Practical considerations

- In practice, it can be beneficial to carry out the parameter estimation using log-transformed parameters
- Automated construction of the sensitivity equation and adjoint equations.
- Selection of the numerical solver (e.g. Sundials CVODES)
- Least squares optimization has been used in many applications

Global optimization through multistart approach

- A non-convex optimization problem
- Multiple local minima
- Parameter space explored using multiple optimization runs
- Parallelizable for high-dimensional problems

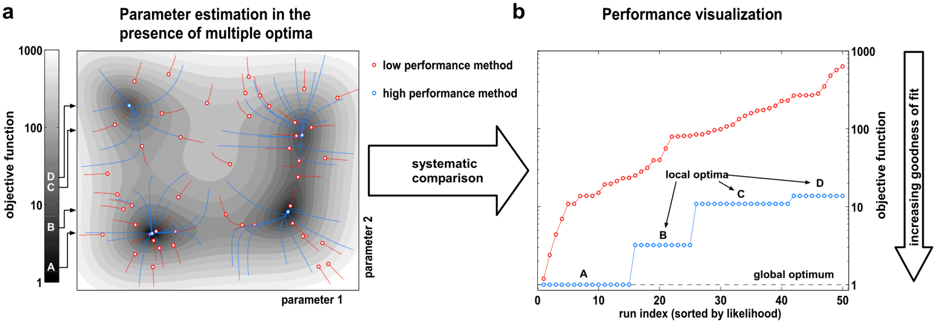


Figure: Illustration of multistart approach (from Raue et al, 2013)

Choosing the initial points for multistart (1)

- Random (Monte Carlo)
- Latin hypercube sampling (LHS)
  - Divide the range of each variable θ_j into equal number (equally probable) of bins
  - Sampling such that for each variable θ_j a particular bin is chosen only once
  - Thus, sampling does not depend on dimension (d) of θ

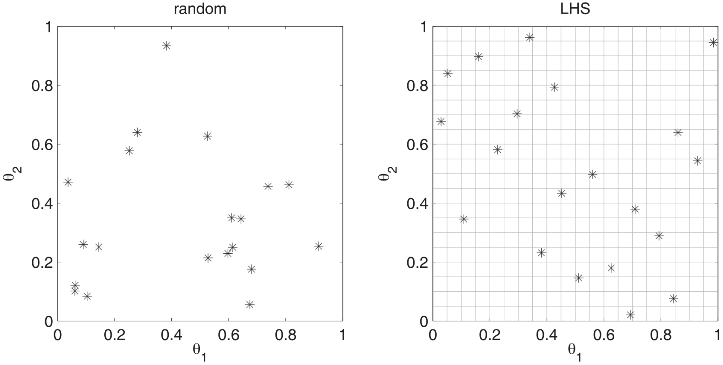


Figure: From Raue et al, 2013

Choosing the initial points for multistart (2)

- Latin hyper-cube sampling gives initial parameter values which are on average more far away from each other

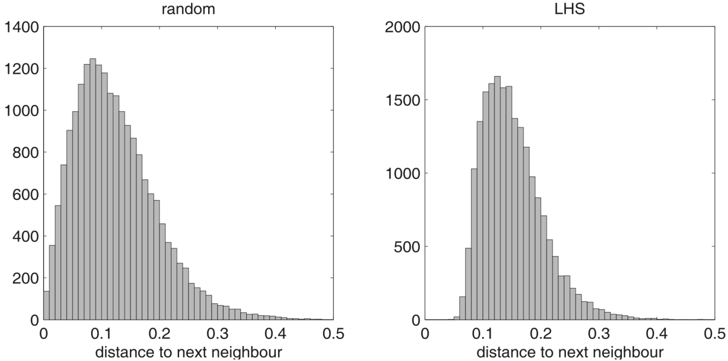


Figure: From Raue et al, 2013

Statistical formulation of the parameter estimation problem

- Formulating what is known about the measurement system in statistical terms
- If we assume normally distributed measurement errors, we can write

$$y(t_i) = x(t_i, \boldsymbol{\theta}) + \epsilon(t_i),$$

where $\epsilon(t_i) \sim \mathrm{N}(0, \sigma_i^2)$.

- This is equivalent to

$$y(t_i) \sim \mathrm{N}(x(t_i, \boldsymbol{\theta}), \sigma_i^2).$$

- By assuming independent measurements, we can express the likelihood of the data given the parameters in the form

$$l(\boldsymbol{\theta}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(y(t_i) - x(t_i, \boldsymbol{\theta}))^2}{2\sigma_i^2}\right).$$

Maximum likelihood estimation

- The maximum likelihood principle is to choose the value $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$ which maximizes $l(\boldsymbol{\theta})$.
- In other words, the maximum likelihood estimate is

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} l(\boldsymbol{\theta})$$

- It is equivalent to solve

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}),$$

  where $L(\boldsymbol{\theta}) = -2\log(l(\boldsymbol{\theta}))$ is the negative log-likelihood.

Negative log-likelihood (1)

$$
\begin{aligned}
L(\boldsymbol{\theta}) &= -2\log\left(\prod_{i=1}^{n}\frac{1}{\sqrt{2\pi\sigma_i^2}}\exp\left(-\frac{(y(t_i)-x(t_i,\boldsymbol{\theta}))^2}{2\sigma_i^2}\right)\right)\\
&= -2\left(\sum_{i=1}^{n}-\frac{1}{2}\log(2\pi\sigma_i^2)-\frac{1}{2}\left(\frac{(y(t_i)-x(t_i,\boldsymbol{\theta}))^2}{\sigma_i^2}\right)\right)\\
&= \sum_{i=1}^{n}\left(\log(2\pi)+2\log(\sigma_i)+\frac{(y(t_i)-x(t_i,\boldsymbol{\theta}))^2}{\sigma_i^2}\right)\\
&= \sum_{i=1}^{n}\left(2\log(\sigma_i)+\frac{(y(t_i)-x(t_i,\boldsymbol{\theta}))^2}{\sigma_i^2}\right)+C_1
\end{aligned}
$$

Negative log-likelihood (2)

- If $\sigma_i = \sigma, i = 1, \ldots, n$, we have

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{n} \left(2\log(\sigma) + \frac{(y(t_i) - x(t_i, \boldsymbol{\theta}))^2}{\sigma^2} \right) + C_1$$

$$= \frac{1}{\sigma^2} \sum_{i=1}^{n} (y(t_i) - x(t_i, \boldsymbol{\theta}))^2 + C_2$$

Negative log-likelihood (2)

- If $\sigma_i = \sigma, i = 1, \ldots, n$, we have

$$\begin{aligned} L(\boldsymbol{\theta}) &= \sum_{i=1}^{n} \left(2\log(\sigma) + \frac{(y(t_i) - x(t_i, \boldsymbol{\theta}))^2}{\sigma^2} \right) + C_1 \\ &= \frac{1}{\sigma^2} \sum_{i=1}^{n} (y(t_i) - x(t_i, \boldsymbol{\theta}))^2 + C_2 \end{aligned}$$

- In general, measurement uncertainty can be characterized by any (possibly non-Gaussian) distribution

$$y(t_i) \sim g(x(t_i, \boldsymbol{\theta}) | \theta_{\text{meas}})$$

Parameter identifiability

- In many cases, the parameters are not well determined
- Infinitely many parameter settings may have equal likelihood
- The effect of changing one parameter can be compensated by tuning other parameters

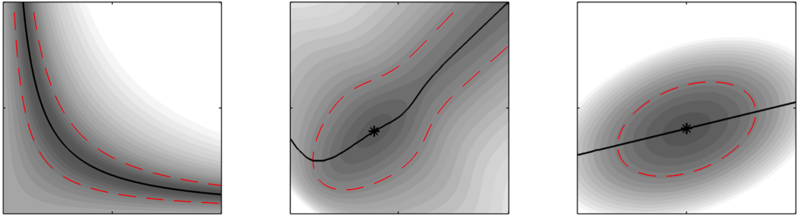


Figure: From Raue et al, 2010

Profile likelihood (1)

- Parameter identifiability can be studied e.g. using the profile likelihood method.
- The profile likelihood for the jth variable is defined by

$$\mathrm{PL}_j(p) = \max_{\boldsymbol{\theta} \in \{\boldsymbol{\theta} | \theta_j = p\}} L(\boldsymbol{\theta}),$$

  where L is now the log-likelihood, $L(\boldsymbol{\theta}) = \log l(\boldsymbol{\theta})$
- Evaluate the profile likelihood around the optimal parameter values $\boldsymbol{\theta}^*$
- Structural and practical non-identifiability
- Sometimes non-identifiability can be solved by reparameterizing the ODE system

Profile likelihood (2)

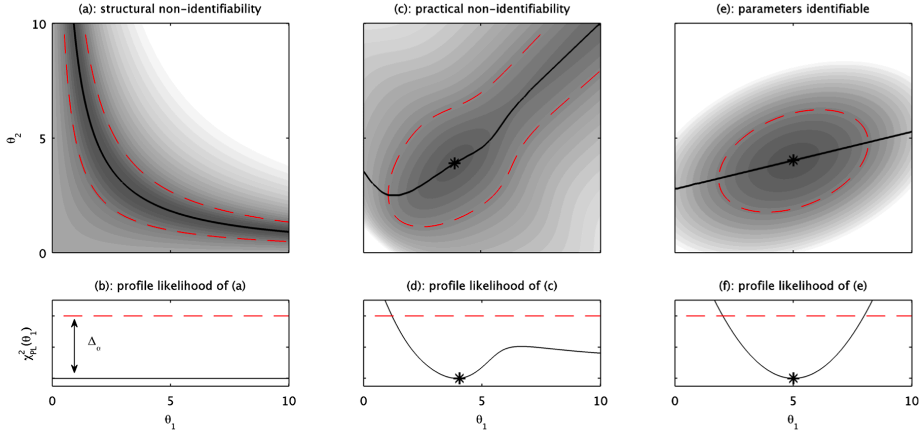


Figure: From Raue et al, 2010

T helper cell differentiation example

- Naive T cells sense the environment and react to different cytokine milieu and differentiate to functionally different T helper subsets
- Proper regulation and balance between various lymphocyte subsets is central the human immune system
- For example, so-called T helper 1 cells are involved in defense against intracellular bacteria
- Molecular mechanisms which control T helper cell differentiation are not at all fully understood
  - A more thorough understanding would allow e.g. better drug design to modulate immune response and help in autoimmune diseases

T helper cell differentiation example (2)

- Different T cell subsets

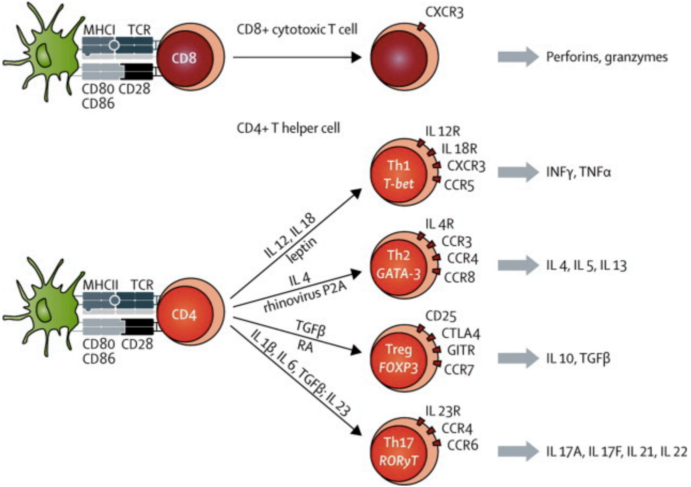


Figure: From (Brusselle et al, 2011)

T helper cell differentiation example (3)

- Key factors (i.e., genes) involved in and driving T helper 1 differentiation include
  - IFN-γ, T-bet, IL-12 and IL-12Rβ2
- The standard/assumed molecular model, until recently, has been the following

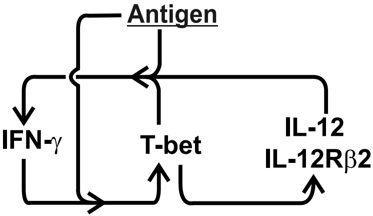


Figure: From (Schulz et al, 2009)

T helper cell differentiation example (4)

- The actual molecular mechanisms are assumed to follow the following ODE system

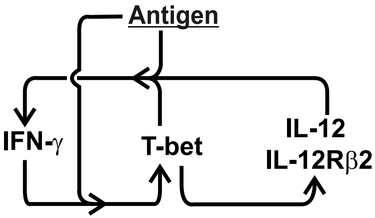


Figure: From (Schulz et al, 2009)

$$\frac{d\,Tbet_{mRNA}}{dt} = \alpha_1 + f_{Tbet}(IFN_{Prot}, Rec_{Prot}, Ag) - \gamma_{Tbet} \cdot Tbet_{mRNA}$$

$$\frac{d\,Tbet_{Prot}}{dt} = \beta \cdot Tbet_{mRNA} - \delta_{Tbet} \cdot Tbet_{Prot}$$

$$\frac{d\,Rec_{mRNA}}{dt} = f_{Rec}(Tbet_{Prot}, Ag) - \gamma_{Rec} \cdot Rec_{mRNA}$$

$$\frac{d\,Rec_{Prot}}{dt} = \beta \cdot Rec_{mRNA} - \delta_{Rec} \cdot Rec_{Prot}$$

$$\frac{d\,IFN_{mRNA}}{dt} = f_{IFN}(Tbet_{Prot}, Rec_{Prot}, Ag) - \gamma_{IFN} \cdot IFN_{mRNA}$$

$$\frac{d\,IFN_{Prot}}{dt} = \beta \cdot IFN_{mRNA} - \delta_{IFN} \cdot IFN_{Prot}$$

Figure: From (Schulz et al, 2009)

T helper cell differentiation example (5)

- To test the model, mRNA time-course measurements have been collected after inducing the Th1 differentiation
- ODE model is fit to the experimental data using simulated annealing

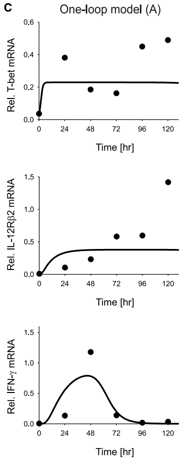
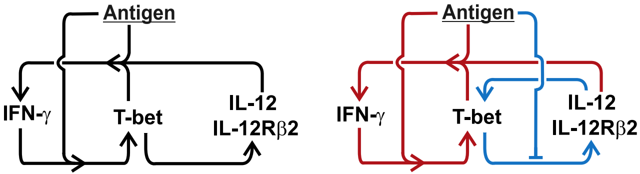


Figure: From (Schulz et al, 2009)

T helper cell differentiation example (6)

- Something wrong with the model? — another network model



$$f_{Tbet} = \alpha_2 \cdot \frac{Ag(t)}{K_1 + Ag(t)} \cdot \frac{IFN_{Prot}}{K_2 + IFN_{Prot}}$$

$$f_{Rec} = \alpha_4 \cdot \frac{Tbet_{Prot}}{K_8 + Tbet_{Prot}}$$

$$f_{Tbet} = \alpha_2 \cdot \frac{Ag(t)}{K_1 + Ag(t)} \cdot \frac{IFN_{Prot}}{K_2 + IFN_{Prot}} + \alpha_3 \cdot \frac{Rec_{Prot}}{K_3 + Rec_{Prot}}$$

$$f_{Rec} = \alpha_4 \cdot \frac{Tbet_{Prot}}{K_8 + Tbet_{Prot}} \cdot \frac{K_4}{K_4 + Ag(t)}$$

Figure: From (Schulz et al, 2009)

T helper cell differentiation example (7)

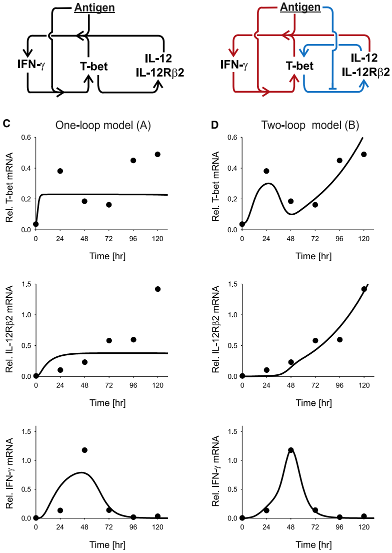


Figure: From (Schulz et al., 2009)

T helper cell differentiation example (8)

- The ODE model is predictive (at least qualitatively)

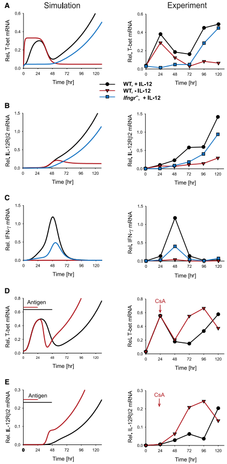


Figure: From (Schulz et al, 2009)

T helper cell differentiation example (9)

- More networks models — a fundamental question concerns choosing the correct biological network model

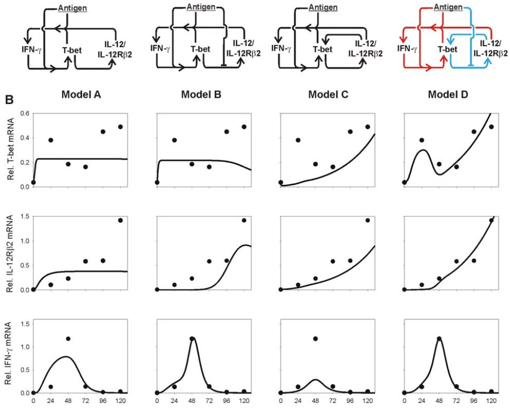


Figure: From (Schulz et al, 2009)

References

- Fröchlich et al. (2017) Scalable Inference of Ordinary Differential Equation Models of Biochemical Processes, *arXiv preprint arXiv:1711.08079*
- Raue et al. (2013) Lessons learned from quantitative dynamical modeling in systems biology, *PLoS One*, 8(9), e74335.
- Raue et al. (2010) Identifiability and observability analysis for experimental design in nonlinear dynamical models, *Chaos*, 20, 045105.
- Brusselle GG et al, (2011) New insights into the immunology of chronic obstructive pulmonary disease, *Lancet*, 378(9795):10–16.
- Schulz et al, (2009) Sequential Polarization and Imprinting of Type 1 T Helper Lymphocytes by Interferon-g and Interleukin-12, *Immunity*, 30, 673–683.