Estimating Differential Equations from Real Data

Jiguo Cao
Simon Fraser University

Outline

Outline

What are Ordinary Differential Equations (ODEs)?

A general form for ODEs:

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{g}(\mathbf{x}|\beta)$$

- **Variable: $\mathbf{x}(t)$;**
- **ODE parameter: β;**
- **ODEs relate the rate of change ($\frac{d\mathbf{x}}{dt}$) of a process to its current state (x)**

Why Using Ordinary Differential Equations (ODEs)?

- ODEs model the rate of change ($\frac{d\mathbf{x}}{dt}$) of a process
- Many models are given directly in ODE forms in Engineering, Physics, Biology, $\cdots$
- Example: Newton's Second Law: $F = m * \frac{d^2}{dt^2}\mathbf{x}(t)$
- Nonlinear ODEs can provide simple models for complex behavior

ODE Solution

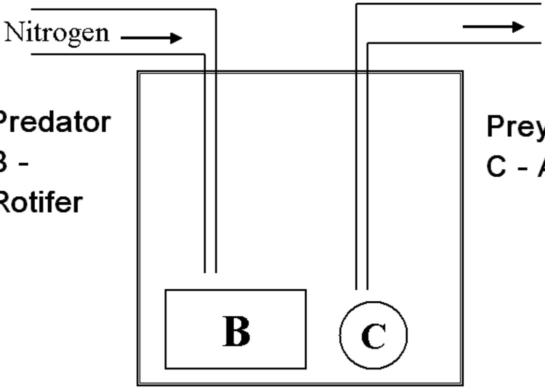
- Most ODEs do NOT have analytic solution.
- Numeric Analysis
  - Given ODE parameter β and initial values $\mathbf{x}(t_0)$
  - Finding ODE solutions and properties numerically
- ODE solutions are sensitive to parameters and initial values

Inverse Problem: Estimating Parameters in ODEs

- **ODEs:** $\frac{d}{dt}\mathbf{x}(t) = \mathbf{g}(\mathbf{x}|\beta)$
- **Observations:** $\mathbf{y}(t_i) = \mathbf{x}(t_i) + \epsilon_i$
- **Object: If** $g(\cdot)$ **is known, Estimating** β

Outline

A Predator-prey system



Predator
B -
Rotifer

Prey:
C - Algae

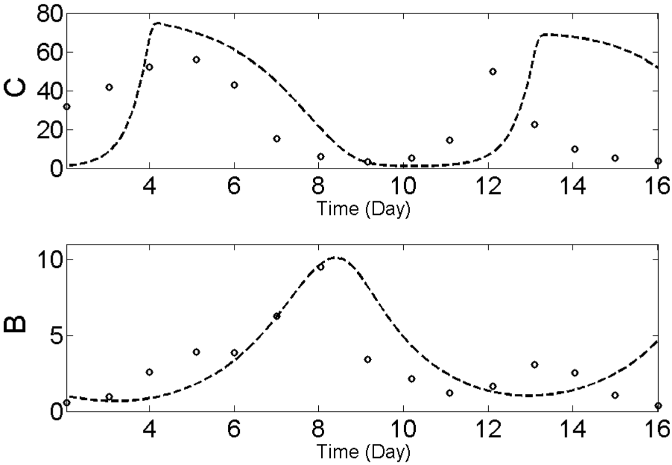
A Predator-prey dynamic Model

- Fussmann et al. (2000). Science 290.

$$
\begin{aligned}
\frac{dN}{dt} &= \delta(N^* - N) - F_C(N)C \\
\frac{dC}{dt} &= F_C(N)C - F_B(C)B/\epsilon - \delta C \\
\frac{dR}{dt} &= F_B(C)R - (\delta + m + \alpha)R \\
\frac{dB}{dt} &= F_B(C)R - (\delta + m)B \, .
\end{aligned}
$$

- N, C, R, B represent the concentration of Nitrogen, Algae, Female Rotifer, and Total Rotifer

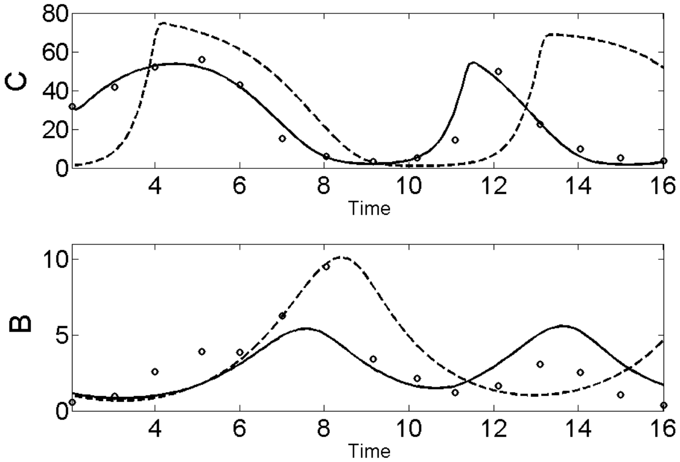
Experimental Data



Estimating Results

Estimates	ϵ	α	m	b_C	**MSE**
Fussmann	0.25	0.40	0.055	3.3	1.96
Profiling	0.11	0.01	0.152	3.9	0.34
SEs	0.020	0.14	0.073	0.47	

Fitting ODEs to data



A Predator-prey dynamic Model

$$\begin{aligned}
\frac{dN}{dt} &= \delta(N^* - N) - F_C(N)C \\
\frac{dC}{dt} &= F_C(N)C - F_B(C)B/\epsilon - \delta C \\
\frac{dR}{dt} &= F_B(C)R - (\delta + m + \alpha)R \\
\frac{dB}{dt} &= F_B(C)R - (\delta + m)B\,.
\end{aligned}$$

Two functional responses

- $F_C(N) = \frac{b_C N}{K_C + N}$
- $F_B(C) = \frac{b_B N}{K_B + N}$

Estimating Functional Responses

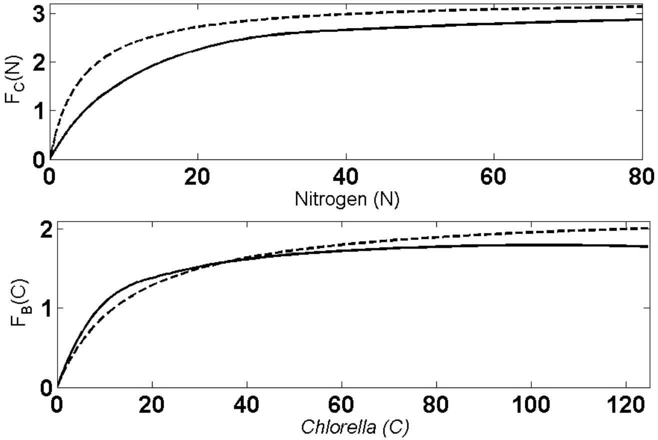
Nonparametric functional responses

$$F_C(N) = \sum c_i^1 \psi_i^1(N)$$

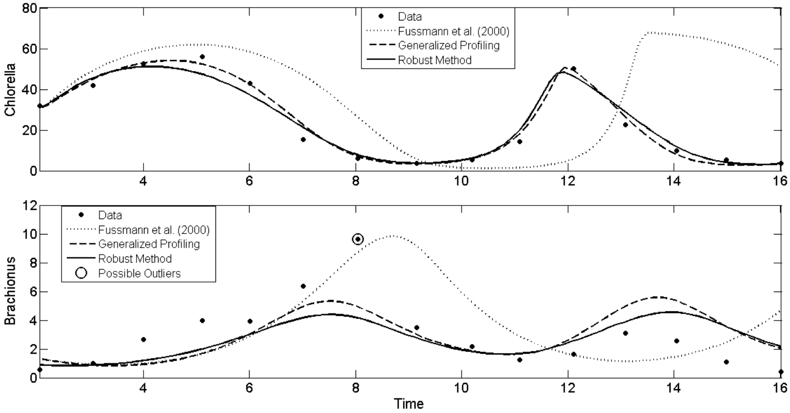
$$F_B(C) = \sum c_i^2 \psi_i^2(C)$$

- $\psi_i^1(N)$ and $\psi_i^2(C)$ are basis functions
- c_i^1 and c_i^2 are basis coefficients

Estimating Functional Responses



Robust Estimation

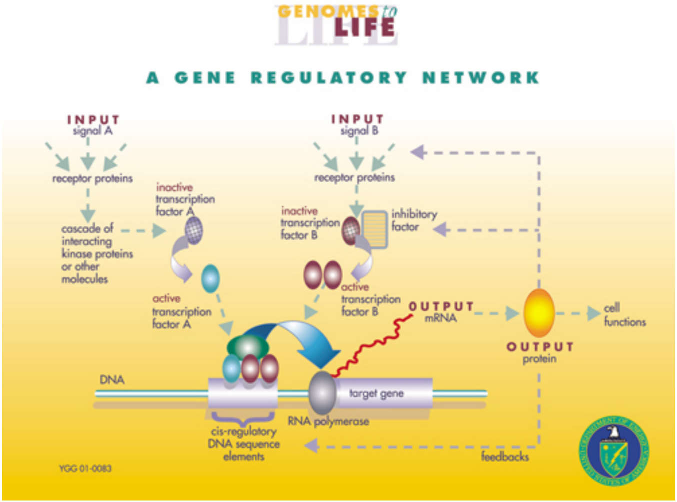


Challenges for the above application

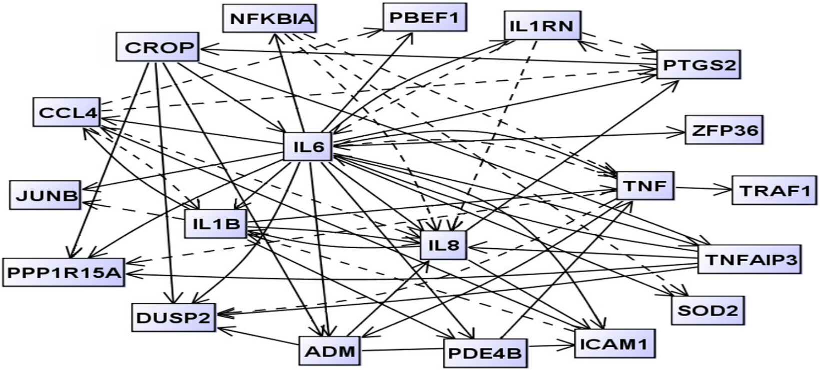
- Two out of four variables cannot be measured.
- Estimating functional parameters.
- Data may have outliers.

Outline

Gene Regulatory Network

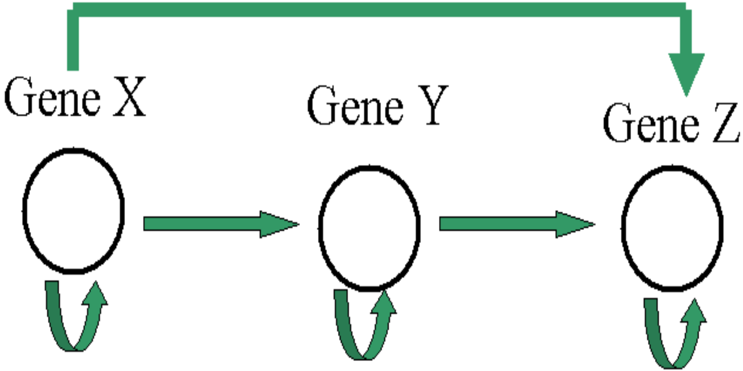


Gene Regulatory Network



Feed-Forward Loop

- Network motifs: certain regulation patterns occur much more often than by chance (Alon 2007)
- **Feed-Forward Loop(FFL): gene regulation networks with three genes.**



Dynamic model for feed-forward loop

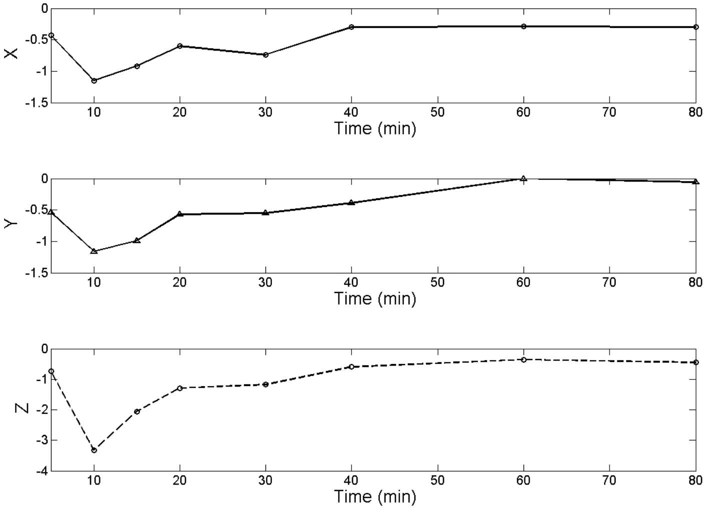
$$
\begin{aligned}
\frac{d}{dt}y(t) &= -\alpha_y * y(t) + \beta_y * f(x) \\
\frac{d}{dt}z(t) &= -\alpha_z * z(t) + \beta_z * f(x) * f(y)
\end{aligned}
$$

- $f(x)$ **is the regulation function**
  - **Activator:** $f(x, K) = \frac{(x/K)^H}{1+(x/K)^H}$
  - **Repressor:** $f(x, K) = \frac{1}{1+(x/K)^H}$
- **7 parameters to estimate:** $\beta = (\alpha_y, \beta_y, \alpha_z, \beta_z, K_{xy}, K_{xz}, K_{yz})$

Objective

- Question 1: If we know the type of regulation between three genes, can we estimate ODE parameters from gene expression data?
- Question 2: Can we infer the type of regulation between three genes from gene expression data?

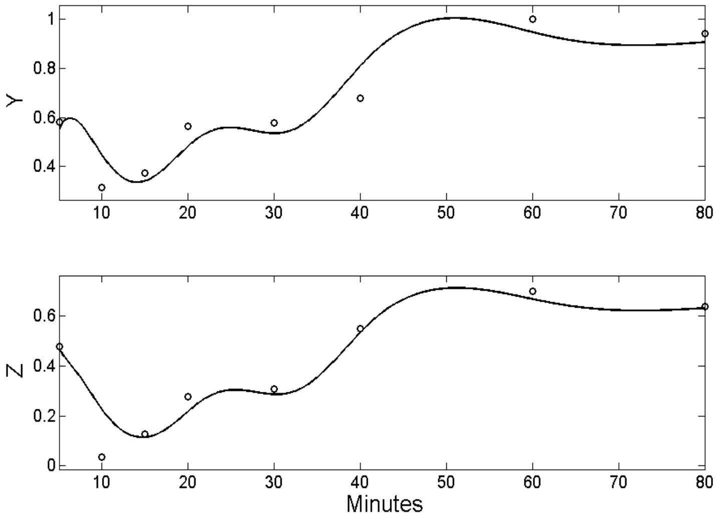
Gene expression data



Parameter estimates with standard errors

Parameters	α_y	α_z	K_{xy}	K_{xz}	K_{yz}
Estimates	0.44	0.69	0.90	0.60	0.56
Standard Errors	0.22	0.18	0.33	0.06	0.15

ODE solutions with estimated parameters values



TEST the goodness of fit of dynamic models

- Question 2: Infer the type of regulations between genes
- Parametric Bootstrap

Gene X	Gene Y	Gene Z	SSE	p-values
GCN4	LEU3	ILV5	0.090	0.25
PDR1	PDR3	PDR5	1.17	0.33
GCN4	LEU3	ILV1	0.092	0.34
YLL044W	YER096W	YDR279W	0.84	0.046

Challenges for the above application

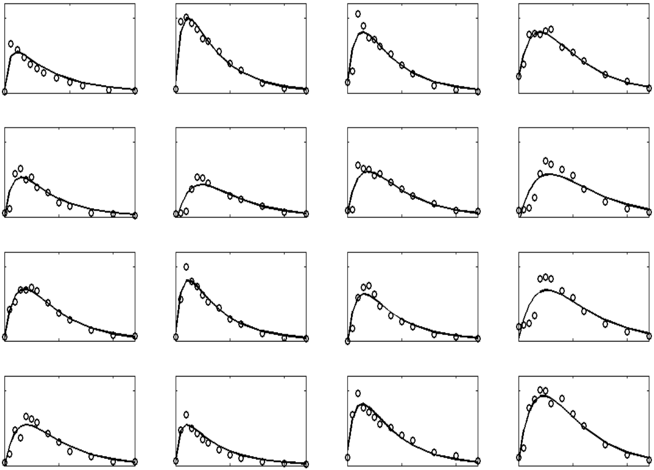
- Data are sparse and not accurate.
- Required to do the model selection.

Outline

Pharmacokinetic Study Of HIV Combination Therapy

- Wasmuth et al (2004) design two different combinations of IDV and RTV: 400/100 mg IDV/RTV combination, and 600/100 mg IDV/RTV combination, which are called Treatment I and II, respectively.
- The serum concentration of IDV and RTV are measured at 0, 0.5, 1.0, 2.0, 2.5, 3.0, 4.0, 5.0, 6.0, 8.0, 10.0 and 12.0 hours for 16 healthy volunteers after they take the dosage twice daily for two weeks.

Drug concentration in the blood



Pharmacokinetics of Drug Concentration

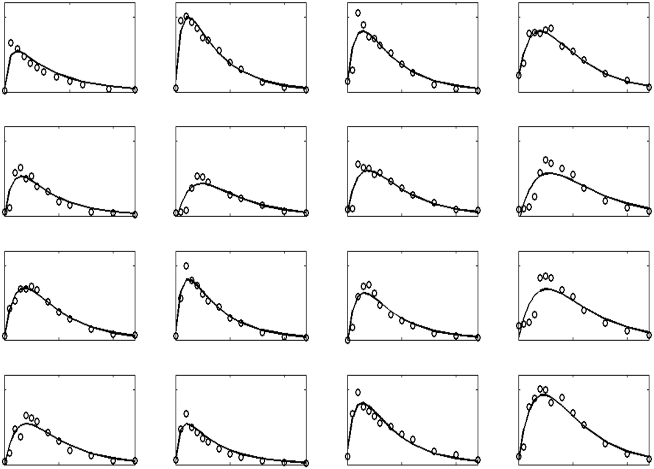
$$\frac{dC_i(t)}{dt} = -Ke_i C_i(t) + \frac{D_i Ke_i Ka_i}{Cl_i} \exp(-Ka_i t)$$

- $C_i(t)$: Drug concentration in the blood
- Cl_i: Rate of the totally body drug clearance
- Ka_i: Drug absorption rate
- Ke_i: Drug elimination rate
- $i = 1, \cdots, N$, is the index of patients.

Estimation for Mixed Effects

Drug		*Ke*		*Ka*		*Cl*		σ_{Ka}	σ_{Cl}
		I	II/I	I	II/I	I	II/I		
IDV	PE	0.30	1.04	1.01	0.68	21.91	0.83	0.57	0.27
	SE	0.02	0.10	0.17	0.16	1.48	0.09	0.11	0.04
RTV	PE	0.20	0.87	0.69	0.55	42.72	1.11	0.68	0.41
	SE	0.02	0.13	0.16	0.23	4.50	0.18	0.14	0.07

Data and ODE fit



Challenges for the above application

- Repeated measurements for one dynamic process.
- Estimate fixed and random effects of ODE parameters.

Outline

An naive approach - Nonlinear Regression

- Solve the ODE

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{g}(\mathbf{x}|\beta)$$

- Denote the solution as

$$\mathbf{x}(t|\beta, \mathbf{x}_0) \quad \mathbf{x}_0 = \mathbf{x}(t_0)$$

- Nonlinear least squares

$$\min_{\beta, \mathbf{x}_0} \sum_{i=1}^{n} \|\mathbf{y}_i - \mathbf{x}(t_i|\beta, \mathbf{x}_0)\|^2$$

Problems

- Optimization is not easy to converge
- Repeatedly solving ODE is computational expensive
- Need to estimate initial values $\mathbf{x}_0 = \mathbf{x}(t_0)$

A two-step procedure

- Ramsay and Silverman 2005; Chen and Wu 2008
- Our model

$$
\begin{aligned}
\mathbf{y}(t_i) &= \mathbf{x}(t_i) + \epsilon_i \\
\frac{d}{dt}\mathbf{x}(t) &= \mathbf{g}(\mathbf{x}|\beta)
\end{aligned}
$$

- Step 1: Estimating derivative $\frac{d}{dt}\mathbf{x}(t)$ using nonparametric smoothing
- Step 2: Estimating β using least squares

Pros and cons of the two-step procedure

- Pro: No need to solve ODE
- Con: The derivative may not be well estimated, especially for sparse data
- Missing data

Monte Carlo Methods

- Markov chain Monte Carlo Method:
  Huang, Liu and Wu (2006), Huang and Lu (2008)
- Smooth Functional Tempering Method:
  Campbell and Steele (Submitted)
  - Addressing Multiple Mode Problem
- Riemann Manifold Langevin and Hamiltonian Monte Carlo:
  Girolami and Calderhead (2010)
  - Fast Converge

Problems

- Sampling sequence is not easy to converge
- Repeatedly solving ODE is computational expensive
- Need to estimate initial values $\mathbf{x}_0 = \mathbf{x}(t_0)$
- Hard to program and tune for naive users.

Parameter Cascading Method

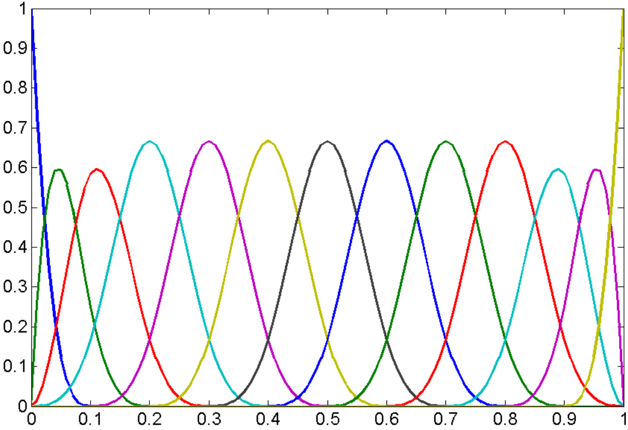
Ramsay, Hooker, Campbell and Cao (2007)

Estimate a smooth function to approximate ODE solutions by a linear combination of basis functions:

$$\mathbf{f}(t) = \sum_{k=1}^{K} c_k \phi_k(t) = \phi(t)'\mathbf{c}$$

- $\phi(t) = (\phi_1(t), \cdots, \phi_K(t))$ **is a vector of basis functions, for example, B-spline (Fixed and Known).**
- $\mathbf{c} = (c_1, \cdots, c_K)$ **is the basis coefficient.**

Cubic B-spline Basis



Two type of parameters

Spline coefficients: c

- Nuisance (local) parameters.
- Parameters not of primary interest.
- The dimension is large and increases with the sample size.

ODE parameters: β

- Structural (global) parameters.
- Parameters of primary interest.
- The dimension is small and fixed with the size of data.

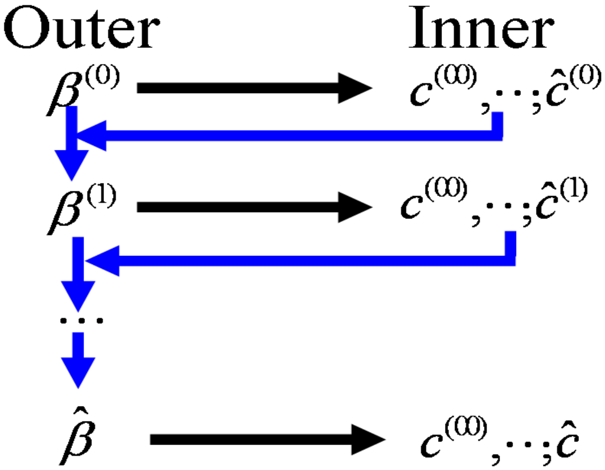
Two nested levels of optimization

- **Inner level:** $J(\mathbf{c}|\beta)$
  $\Rightarrow\hat{\mathbf{c}}$ **is a function of** β**:** $\hat{\mathbf{c}}(\beta)$**.**

- **Outer level:** $H(\hat{\mathbf{c}}(\beta), \beta)$

$\mathbf{c}$: basis coefficients; β: ODE parameter;

Diagram for Parameter Cascading



Inner level of optimization criterion

Smooth function: $\mathbf{f}(t) = \phi(t)'\mathbf{c}$

Fitting to data

- **Observations:** $\mathbf{y}(t_i)$
- **Fitting to data**: $C_1 = \sum_{i=1}^{n} [\mathbf{y}(t_i) - \mathbf{f}(t_i)]^2$

Fidelity to ODE $\frac{d}{dt}\mathbf{x}(t) = \mathbf{g}(\mathbf{x}|\beta)$

- **Difference between two sides of ODE**: $L\mathbf{f}(t) = \frac{d}{dt}\mathbf{f}(t) - \mathbf{g}(\mathbf{f}(t)|\beta)$
- **Fidelity to ODE**: $C_2 = \int [L\mathbf{f}(t)]^2 dt$

Criterion: $J(\mathbf{c}|\beta) = C_1 + \lambda C_2$

Outer level: estimating ODE parameter β

- Criterion: $H(\beta) = \sum_{i=1}^{n} [\mathbf{y}(t_i) - \phi(t_i)'\hat{\mathbf{c}}(\beta)]^2$

- **Inner level:** $J(\mathbf{c}|\beta)$
  $\Rightarrow \hat{\mathbf{c}}$ is a function of β: $\hat{\mathbf{c}}(\beta)$.

- **Outer level:** $H(\hat{\mathbf{c}}(\beta), \beta)$

Unique aspects of Parameter Cascading Method

- **Different optimization criterion in each level**
- **Functional relationships among parameters**

$\mathbf{c}$: basis coefficients; β: ODE parameter;

Low computation load

- **Newton-Raphson algorithm is applied.**
- **Gradients and Hessian matrices worked out analytically**

- **Inner level:** $J(\mathbf{c}|\beta)$
- **Median level:** $H(\beta)$
- **Gradient:** $\frac{dH}{d\beta} = \frac{\partial H}{\partial \beta} + \frac{\partial H}{\partial \hat{\mathbf{c}}} \frac{\partial \hat{\mathbf{c}}}{\partial \beta}$

$\mathbf{c}$: basis coefficients; β: ODE parameter;

Low computation load

- Newton-Raphson algorithm is applied.
- Gradients and Hessian matrices worked out analytically

- Inner level: $J(\mathbf{c}|\beta)$
- Median level: $H(\beta)$
- Gradient: $\frac{dH}{d\beta} = \frac{\partial H}{\partial \beta} + \frac{\partial H}{\partial \hat{\mathbf{c}}} \frac{\partial \hat{\mathbf{c}}}{\partial \beta}$

Implicit Function Theorem

- Inner level: $J(\mathbf{c}|\beta, \lambda, \mathbf{y}) \quad \Rightarrow \hat{\mathbf{c}}(\beta, \lambda)$
- $\frac{\partial \hat{\mathbf{c}}}{\partial \beta} = -\left[\frac{\partial^2 J}{\partial \mathbf{c}^2}\right]^{-1} \left[\frac{\partial^2 J}{\partial \mathbf{c} \partial \beta}\right]$

Smoothing Parameter Selection

- $\mathbf{x}(\hat{\beta})$ is the numeric solution of ODE with parameter $\hat{\beta}$
- Minimizing Prediction Error
  $F(\lambda|\mathbf{y}) = \sum_{i=1}^{n}[\mathbf{y}(t_i) - \mathbf{x}(\hat{\beta}(\lambda))]^2$
- The ODE parameter estimation is NOT sensitive to the precise value of the smoothing parameter within certain range.

Byproduct: estimating initial values

- Fitted curve: $\hat{\mathbf{x}}(t) = \phi(t)'\hat{\mathbf{c}}$
- Estimating initial values: $\hat{\mathbf{x}}(t_0) = \phi(t_0)'\hat{\mathbf{c}}$