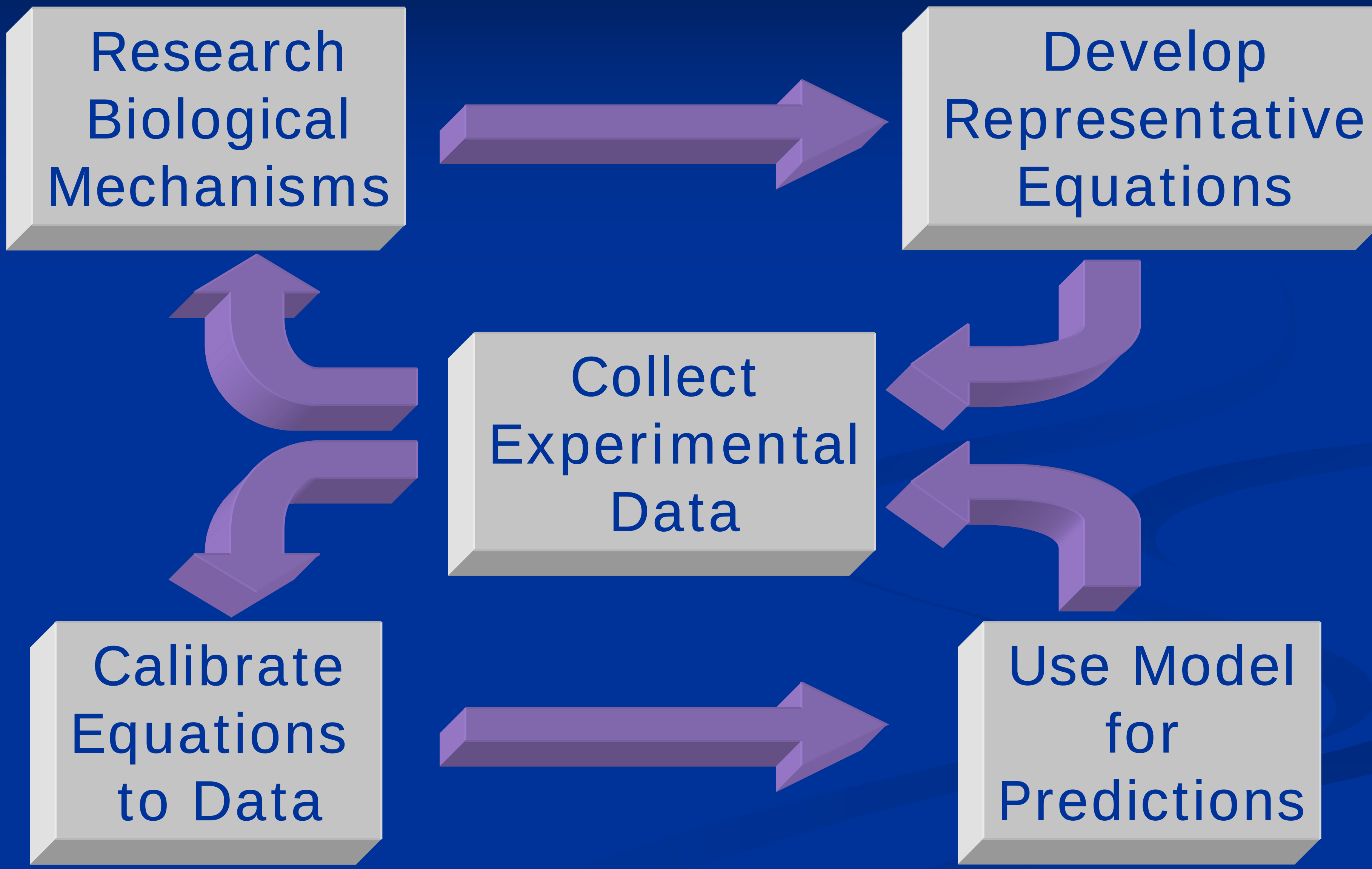
Methods for Inverse Problems In Biology

John Bartels, Immunetrics, Inc.

Problem statement

- Modelers create rules or equations to capture essential biological interactions
  - ODE example: $P' = gP * P - dP * P * K$
- Often employ free parameters for tuning model behavior
- Unknown for many reasons:
  - Unmeasurable, nonphysical, individual variation, etc.
- Inverse problem: given data, find the "right" parameters
- Goals: choose parameter values such that
  - "Fitting is **not** enough!"
  - We reproduce experimental data well
  - Values are justifiable
  - We understand the error in our estimates

Estimation is part of a <u>cycle</u>



Inputs For Estimation Process

- Identify parameters to estimate
  - Obtain bounds & initial estimates if possible!

- Data
  - Empirical data (clinic / lab / literature)
    - e.g. time series for all scenarios of interest
  - Qualitative heuristics (literature, expert intuition)
    - e.g. system constraints, parameter constraints
  - Separate into Training & Validation sets
    - Don’t cheat with validation sets!

Define "best" parameters!

- Intuition: choose parameter θ value to maximize "likelihood":
  - Probability of predicting data given these parameters

- Bayesian statistics gives us formalism

- Need two probabilities
  - "Probability of data" given some parameters
    - Natural for stochastic model
    - Deterministic (e.g. ODE) models – add noise
  - Probability of a parameter value
    - Based on *a priori* knowledge – what's biologically plausible?
    - Often no info – *assume* some distribution!

Maximum Likelihood Principle

- Likelihood of parameter values given by Bayes' Theorem:

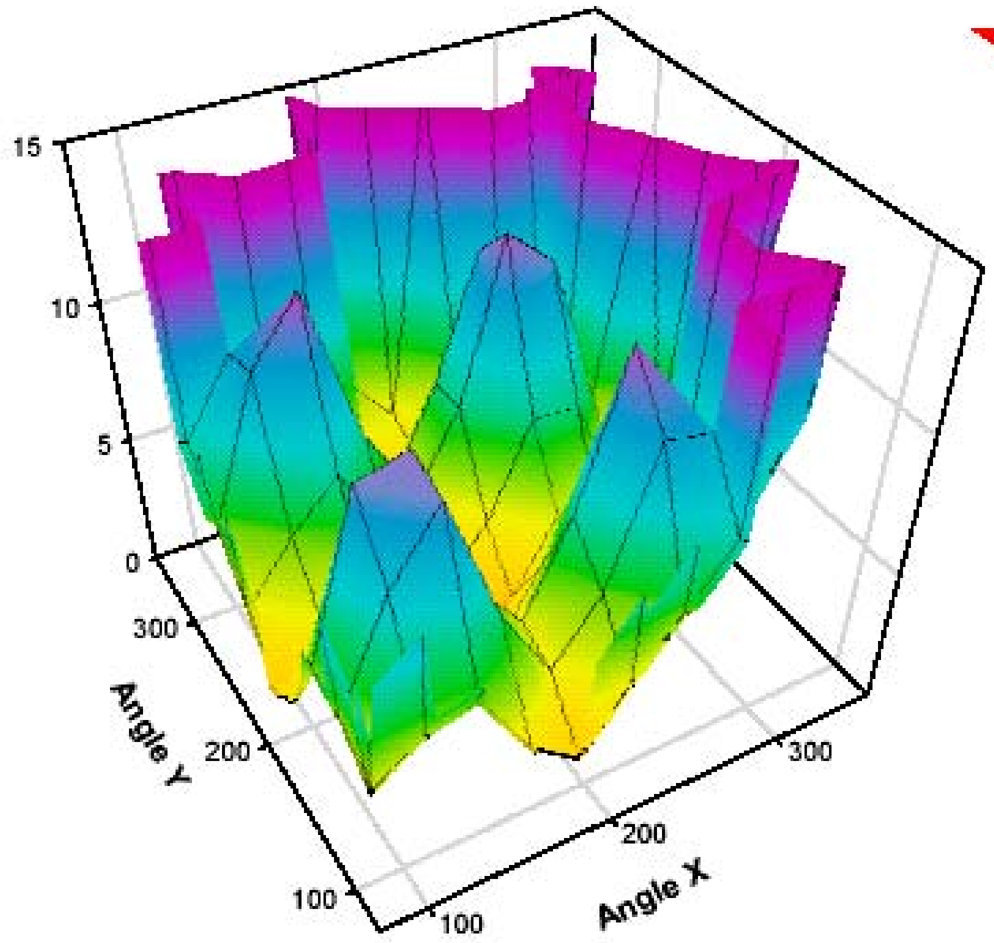
$$P(params \mid data) = \frac{P(data \mid params)P(params)}{P(data)}$$

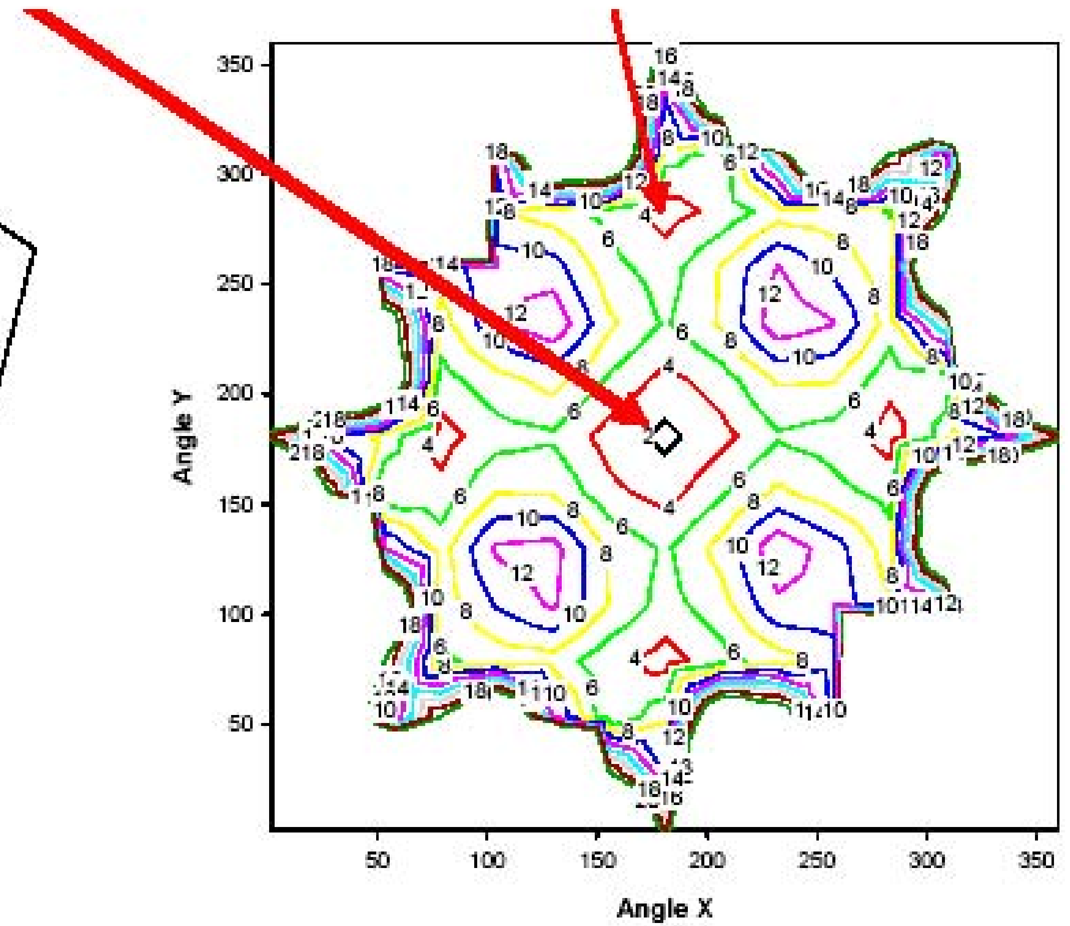
- Concrete example - estimate parameter θ from some data:

$$P(\theta == 1.8 \mid data) = \frac{P(map(1) = 100, map(2) = 95, \ldots \mid \theta = 1.8) P(\theta == 1.8)}{P(map(1) == 100, map(2) = 95, \ldots)}$$

- Gives basis for practical methods:
  - Least-squares fitting
  - Monte Carlo (e.g. Markov Chain, etc.)
  - Kalman Filters

Visualizing likelihoods





How hard can it be?

- Nonlinear interactions: easy for linear problems
- High Dimensionality (10's to 1000's)
- Rough Fitness Landscape
  - many comparable “high scoring” solutions
- Large spread in data
- Sparseness of data
  - Many unmeasured system variables!
  - Limited time-points
- Must generalize beyond training scenarios
- Beware of **overfitting**!

Approach I: Least Squares Fits

- If errors in data follow normal dist with mean=0, minimize:

$$Err(\theta) = \sum_s \sum_v \sum_t (obs(s,v,t) - pred(\theta,s,v,t))^2$$

- Advantages:
  - Optimal under *somewhat* realistic assumptions
  - Intuitive, symmetric, cheap, differentiable
  - Chi-square term for measurement error
  - Confidence interval formulas

- How to find the values of θ that minimize the error?

Numerical Optimization For Least Squares

- Exhaustive search (simple, slow/intractable)

- Linear programming (only for linear problems)

- Dynamic programming (only for decomposable problems)

- Gradient searches: potential candidate!
  - Recall: derivative ==0 at minimum of a function
  - Gradient of scoring function is direction of steepest change – follow it!
  - Problems: Step size choice! Gets stuck!

Common Search Algorithms

- Gradient methods (local search)
  - Levenberg-Marquardt, Gauss-Newton, etc.
  - Hillclimbing

- Stochastic Gradient-like methods (global)
  - Simulated Annealing
  - Evolutionary / Genetic Algorithms

- Simplex methods (e.g. Nelder-Mead)

Practical concerns

- Guide the optimizer:
  - Exploit domain knowledge: symmetry, conservation, physical limits, etc.
  - Weight Important features of curves, etc.

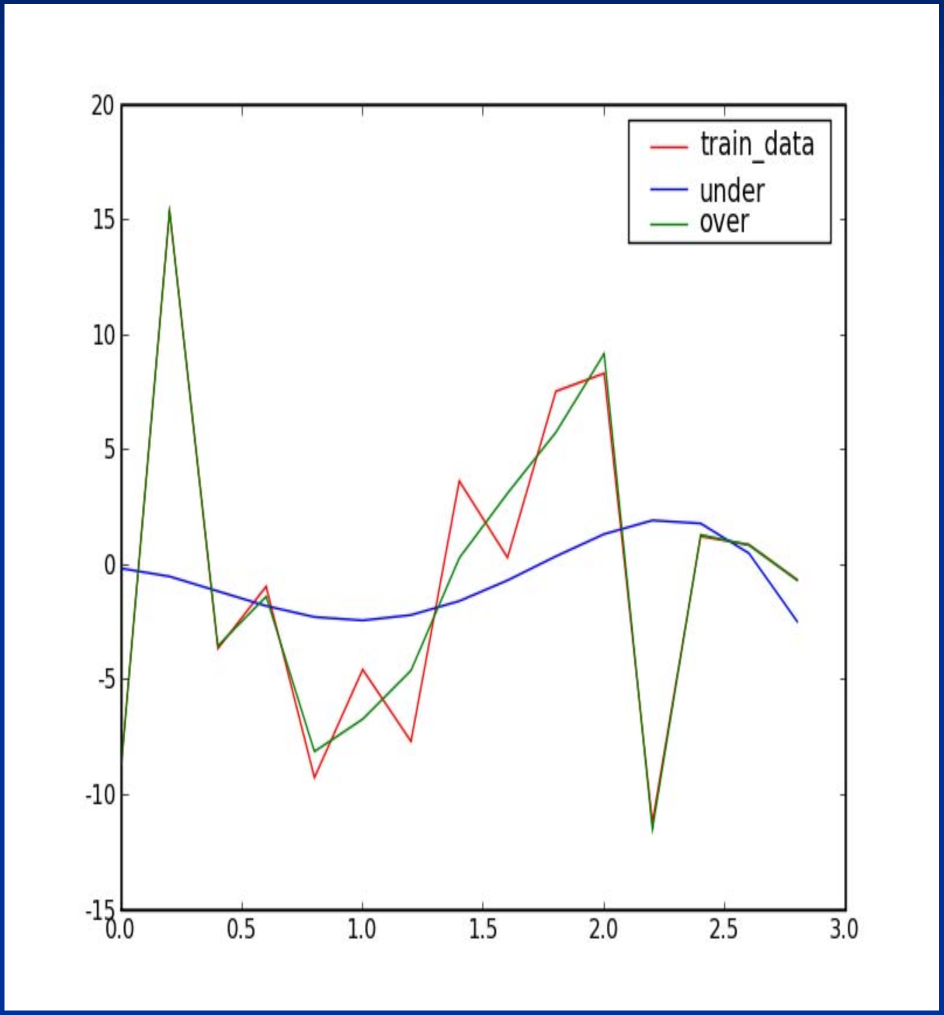
- Beware of:
  - Non-physical solutions
  - Data on vastly different scales
  - Effects of noisy data
  - Biased training sets

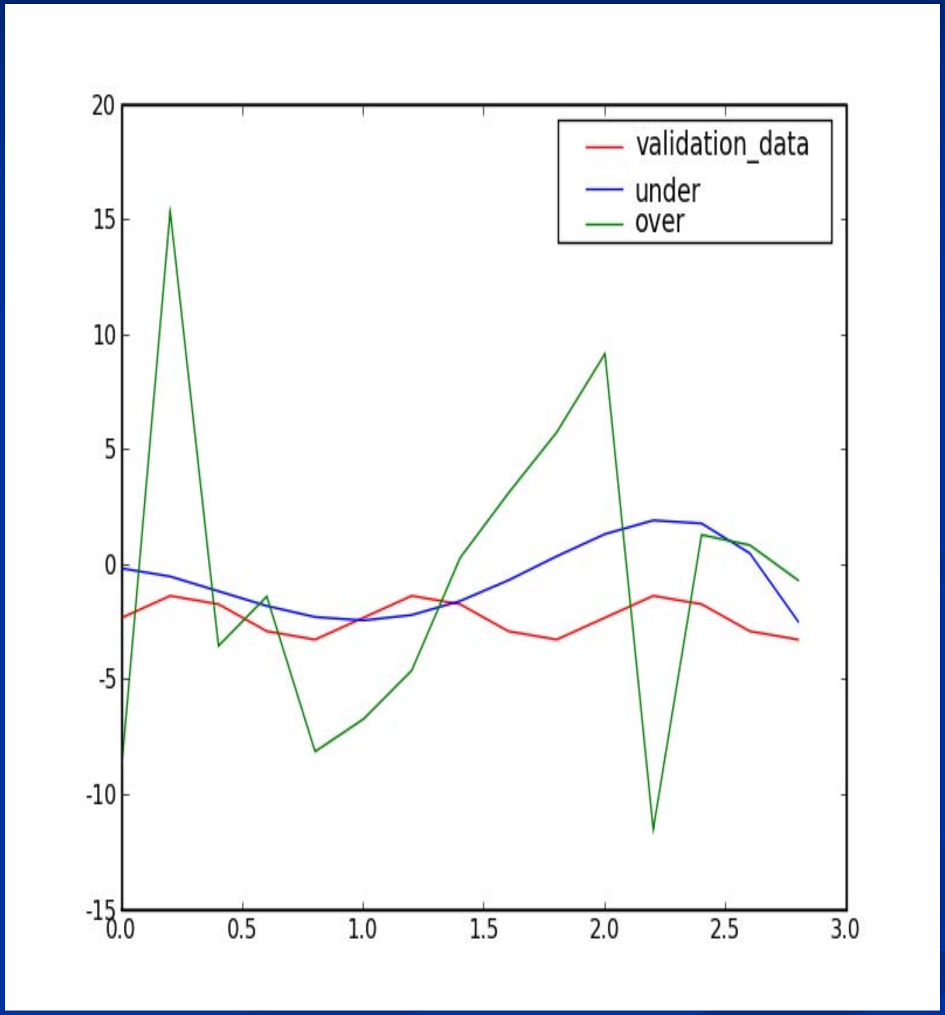
Dangers: Overfitting

- Problem:
  - Model trains "too well"
  - Poor generalization to prediction sets
- Causes: Too many parameters, too little data!
  - Model memorizes *noise* in input

- Worse fit is sometimes better?

Overfitting Example





Dangers: Identifiability

- Question: how reliably can we estimate the parameters?
  - Three classes: Unique / Non-unique / Non-identifiable
    - Estimates of non-identifiable parameters are meaningless!
  - Consider:

$$A'(x) = \theta_1 B - \theta_2 B = B(\theta_1 - \theta_2)$$

$$A'(x) = \theta_1 B - \theta_2 C$$

- Complicated by:
  - Scarce / noisy data
  - Biology's full of feedback loops & compensation

- Quantified by: PCA / SVD, correlation matrices

Approach II: Back to Bayes

- Best point estimate for 1 parameter is the average value:

$$\theta^* = \int_{\min}^{\max} \theta \, P(\theta \mid data) \partial\theta$$

- For n > 1, same idea: mean of multi-dim function:

$$\theta_i^* = \int \theta_i P(\theta_i \mid data) \partial \theta_i$$

where

$$P(\theta_i \mid data) = \int_{\min_1}^{\max_1} \int_{\min_2}^{\max_2} \ldots \int_{\min_n}^{\max_n} P(\theta \mid data) \partial\theta_1, \ldots, \partial\theta_{i-1}, \partial\theta_{i+1}, \ldots, \partial\theta_n$$

Two big problems to solve

- Bayes: we can get $P(\theta \mid data)$ if we know $P(data \mid \theta)$
  - Stochastic model: repeat runs
  - Deterministic model: add noise to data
    - Know your error sources!
  - Priors for parameter values!

- Multi-dimensional integrals are hugely expensive!
  - Solution: throw *a lot* of darts to approximate it
  - Intuitive method: rejection sampling
  - Better, fancier methods based on these ideas
    - Terms: Gibbs Sampling, Markov Chain Monte Carlo, VEGAS

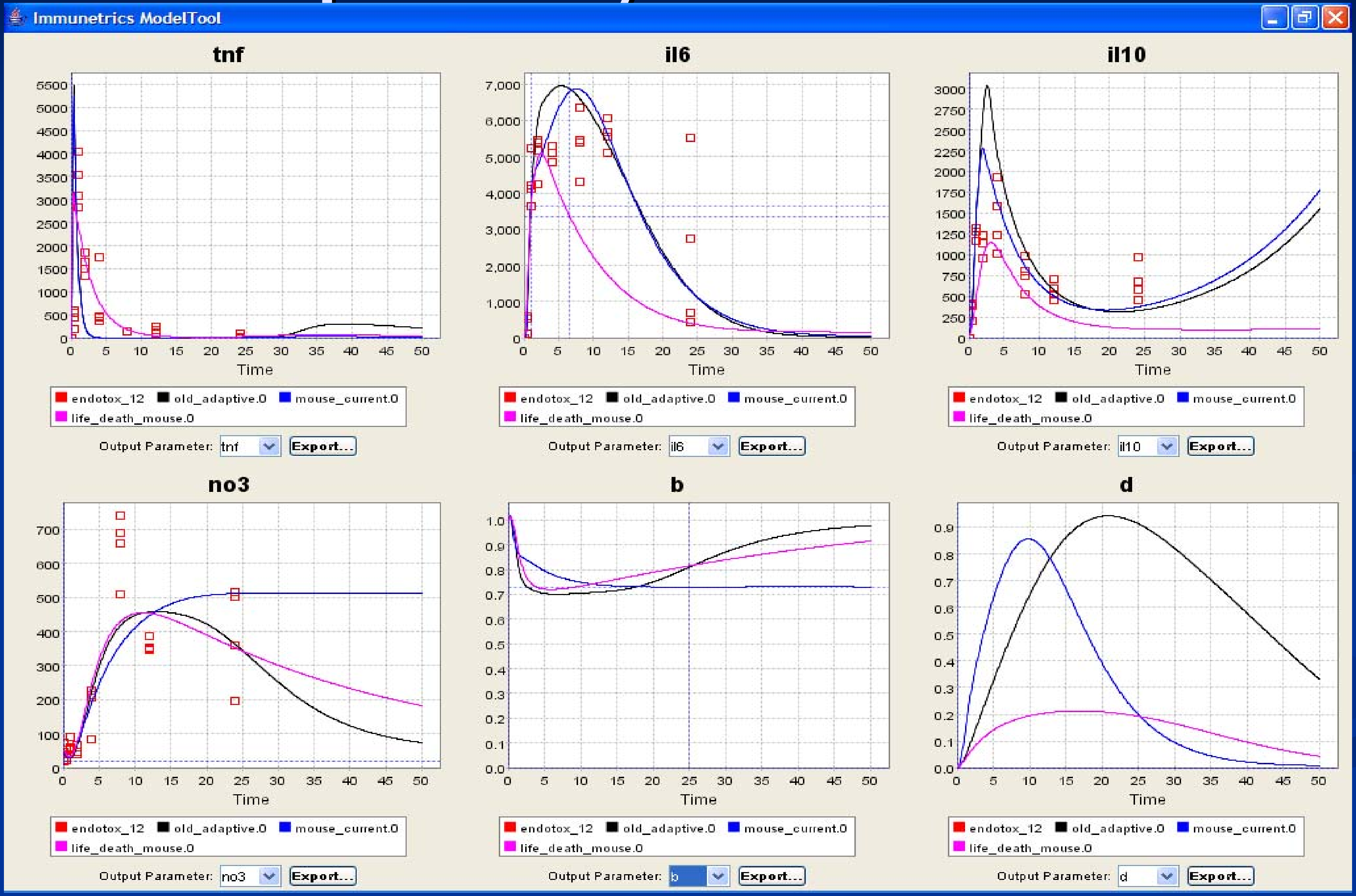
Least-Squares vs. Monte Carlo

- Least Squares
  - Classic, widely used, lots of libraries
  - Assumes normal-dist, independent errors!
  - (Rough) Confidence intervals available
  - Often faster, good enough
- Monte Carlo
  - Relatively new, fewer libraries
  - Immune to badly distributed error in data
  - Sensitive to assumed priors
  - Still computationally expensive
  - Probably the way of the future (eventually)

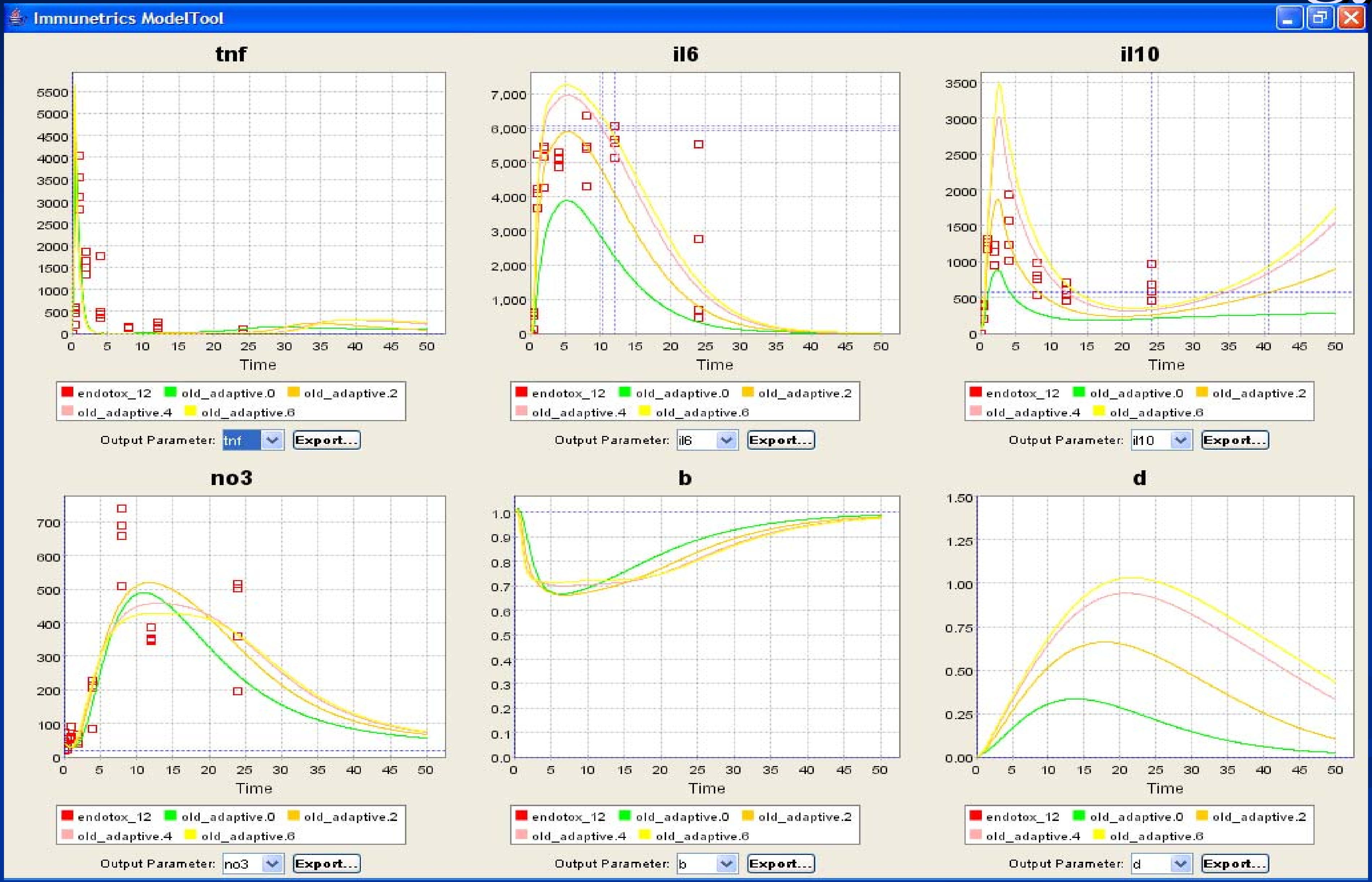
Real world examples

- Apply these algorithms to existing model
- Training Data: only 4 measured curves curves
- Can we fit the training data well (over all scenarios)?
- Try multiple fits: do they find similar results?
- What about the unmeasured variables?

4 Equation / 5 Scenario Fits



Without Heuristics: Violates Biology

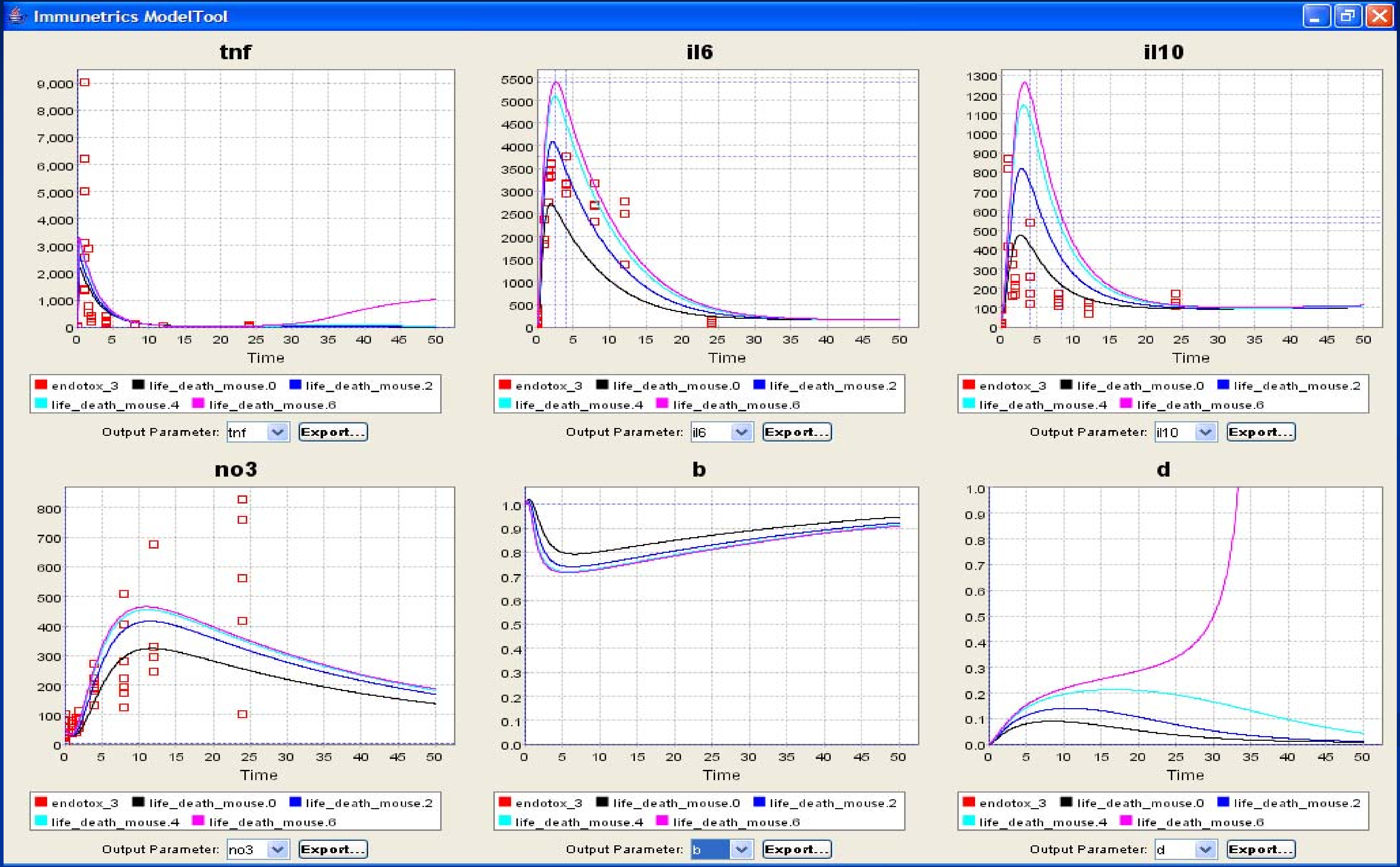


Results

- Multiple models that fit, but too under-constrained!
  - Models show different behavior on untrained data
  - Identifiability: similar fits but
    - Different mechanisms used
    - Very different parameter values

- Assessing models:
  - Biological intuition for simple cases?
  - Without intuition: when is the model right?
  - Good fits for the wrong reasons

Heuristic Fit: Enforces Qualitative Rules



Fitting Underconstrained Models

- Assess model identifiability
- Consider Model Clouds
- Model & experiment design must influence each other!
- Simplification:
  - Reduce to simplest justifiable model (PCA)
- Knowledge of the model:
  - limit parameters (and ranges) to search
  - Optimization algorithms with greater awareness of the model's structure
  - Biological heuristics as further constraints

Some References

- Books / Papers:
  - Overview: Schittkowski
  - MC: Tarantola, Geman & Geman,
  - Identifiability: Jaquez
  - Genetic Algorithms: M. Mitchell; D. Goldberg
  - Simulated Annealing: Kirkpatrick
  - General numerics, simple optimization: Numerical Recipes: Press, et al.

- Software:
  - Matlab (plus add-ons; LS, MC)
  - DAKOTA (LS)
  - BUGS (MC)
  - GNU Scientific Library (LS)
  - Lots of open-source (and commercial) code
  - Write your own?