Modeling & Simulation
(Computational Immunology)

Steven H. Kleinstein

YALE Pathology Informatics

Departments of Pathology and Immunobiology
Yale School of Medicine

steven.kleinstein@yale.edu

April 15, 2019

Inverse Model

- A mathematical model designed to fit experimental data so as to explicitly quantify physical or physiological parameters of interest

- Values of model elements are obtained using parameter estimation techniques aimed at providing a "best fit" to the data

- Generally involves an iterative process to minimize the average difference between the model and the data

- Evaluating the quality of an inverse model involves a combination of established mathematical techniques as well as intuition and creative insight

Understanding cell proliferation and death

BrdU (thymidine analog) incorporated into cell DNA during S-phase

Flow cytometry to quantify antigen-specific germinal center B cells…

BrdU incorporated
during S phase

science.csustan.edu/confocal/Images/SCE/index.SCE.htm
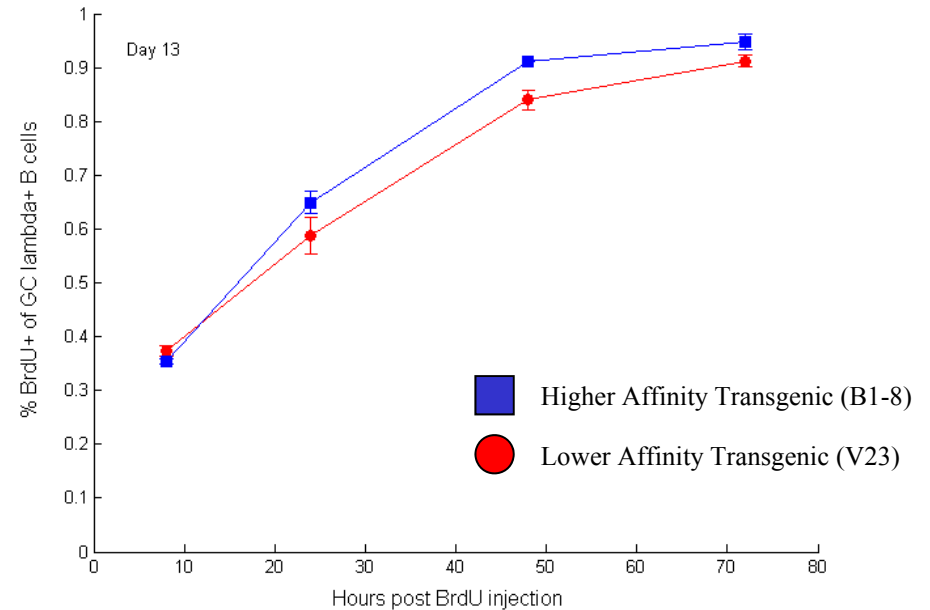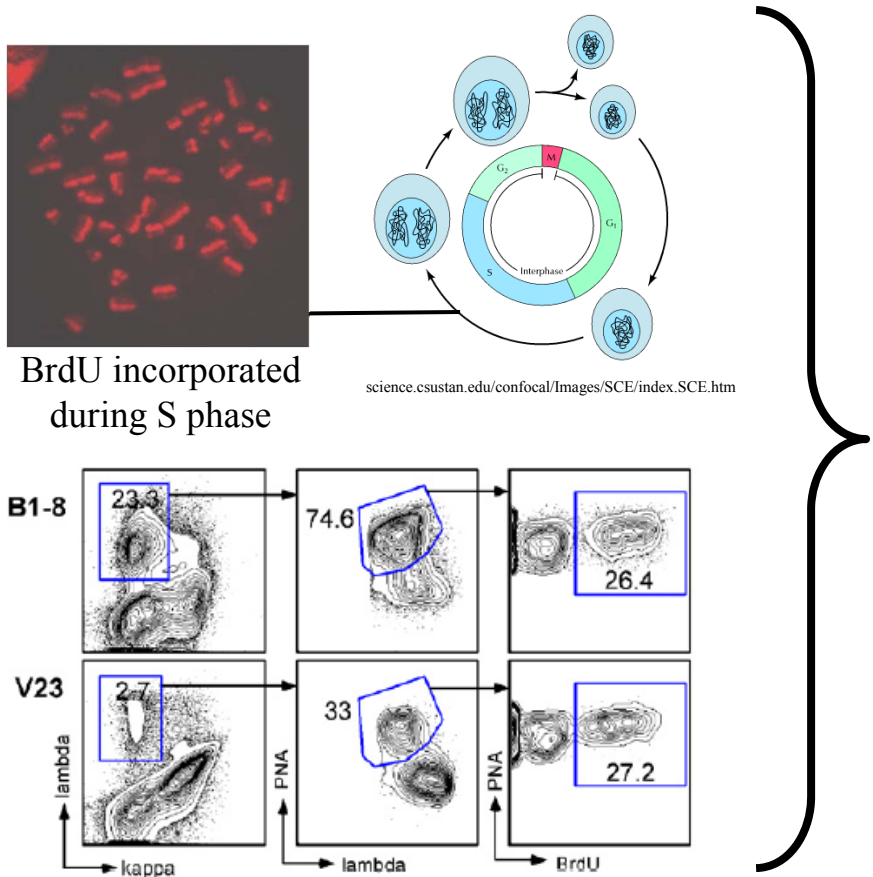
Labeling curves look similar – suggests same proliferation rate?

Understanding cell proliferation and death

At steady-state, rate at which the fraction of BrdU labeled cells increases is indicative of the sum of the per cell proliferation and death rates

Quantification of Cell Turnover Kinetics Using 5-Bromo-2′-deoxyuridine[1]

Sebastian Bonhoeffer,* Hiroshi Mohri,† David Ho,† and Alan S. Perelson[2]*‡

The Journal of Immunology, 2000, 164: 5049–5054.

Rapid Turnover of T Lymphocytes in SIV-Infected Rhesus Macaques

Hiroshi Mohri, Sebastian Bonhoeffer, Simon Monard, Alan S. Perelson, David D. Ho*

www.sciencemag.org • SCIENCE • VOL. 279 • 20 FEBRUARY 1998

International Immunology, Vol. 15, No. 3, pp. 301–312
doi:10.1093/intimm/dxg025, available online at www.intimm.oupjournals.org

Asynchronous differentiation models explain bone marrow labeling kinetics and predict reflux between the pre- and immature B cell pools

Ramit Mehr[1], Gitit Shahaf[1], Alex Sah[2] and Michael Cancro[2]

The Journal of Immunology

Taking Advantage: High-Affinity B Cells in the Germinal Center Have Lower Death Rates, but Similar Rates of Division, Compared to Low-Affinity Cells[1]

Shannon M. Anderson,* Ashraf Khalil,† Mohamed Uduman,‡§ Uri Hershberg,*†§
Yoram Louzoun,¶ Ann M. Haberman,† Steven H. Kleinstein,‡§ and Mark J. Shlomchik[2]*†

Oncogene (2005) 24, 7514–7523
© 2005 Nature Publishing Group All rights reserved 0950-9232/05 $30.00
www.nature.com/onc

Reduced cell turnover in lymphocytic monkeys infected by human T-lymphotropic virus type 1

Christophe Debacq[1,5], Jean-Michel Héraud[2,5], Becca Asquith[3], Charles Bangham[3], Fabrice Merien[2], Vincent Moules[4], Franck Mortreux[4], Eric Wattel[4], Arsène Burny[1], Richard Kettmann[1], Mirdad Kazanji[2] and Luc Willems*[1]

Models of BrdU incorporation integral part of many studies

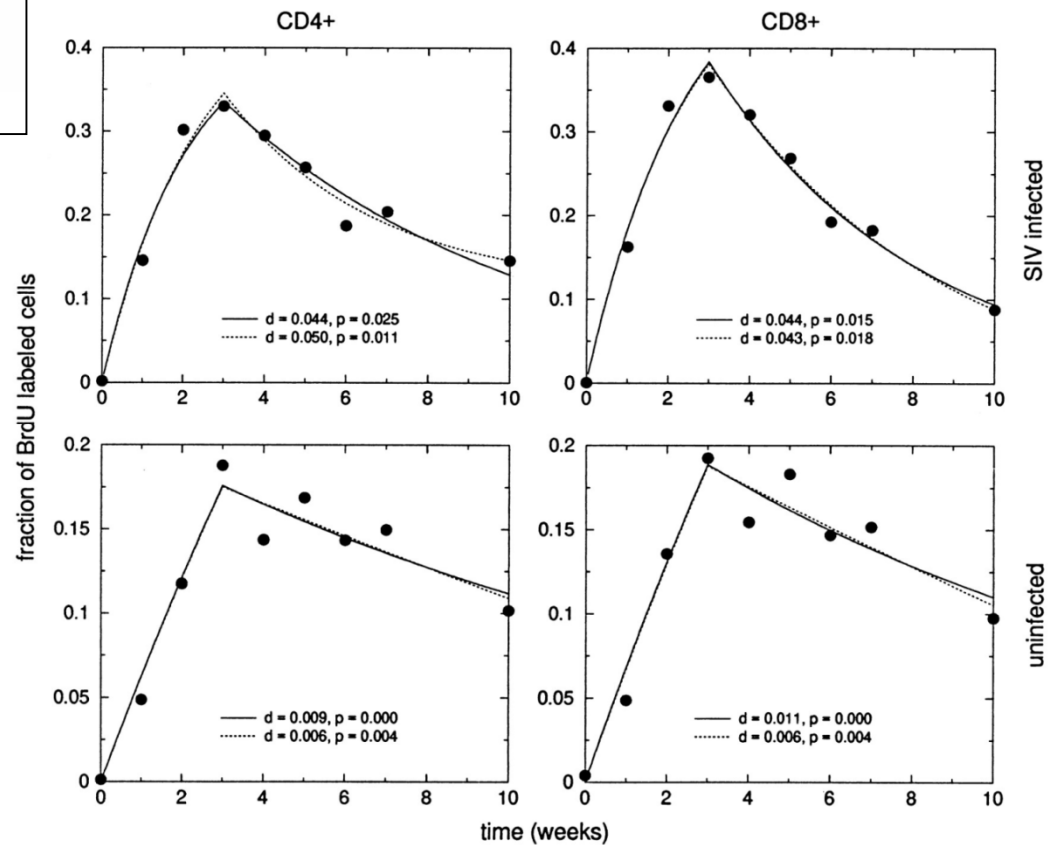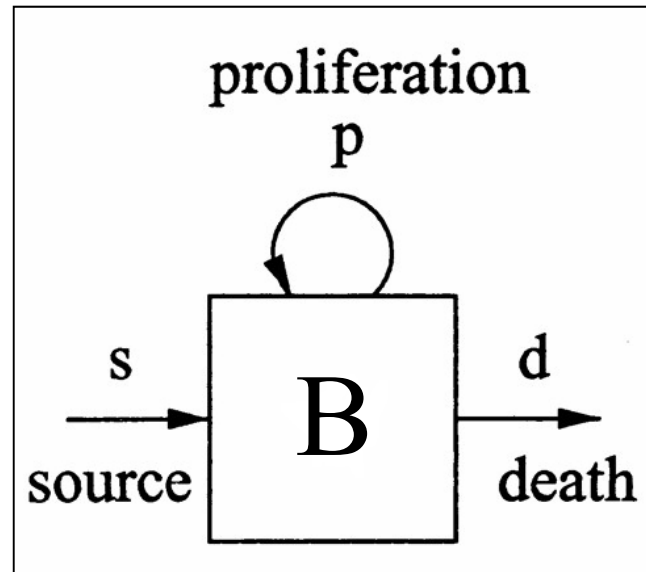# BrdU labeling of CD4+ and CD8+ T lymphocytes

SIV-infected and an uninfected macaque. Data are from Mohri et al., Science (1998)

Is there a difference in cell turnover?

Model of BrdU Labeling

Start with a basic model of cell population dynamics…

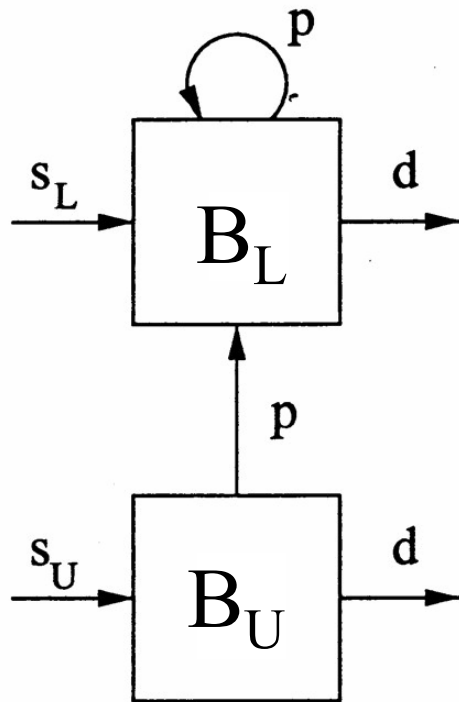Rate of change
in B cell
population

$$\frac{dB}{dt} = s + pB - dB$$

Often can often assume population in steady-state (i.e., constant)

Model of BrdU Labeling

Split the B cell population into Labeled (B_L) and Unlabeled (B_U) subsets

B) During BrdU administration
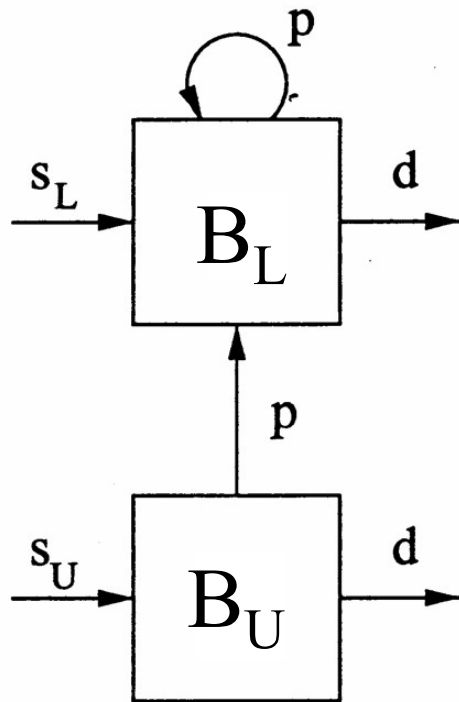
$$\frac{dB_U}{dt} = s_u - pB_U - dB_U$$

$$\frac{dB_L}{dt} = s_l + 2pB_U + pB_L - dB_L$$

Do data contain enough information to estimate parameters?
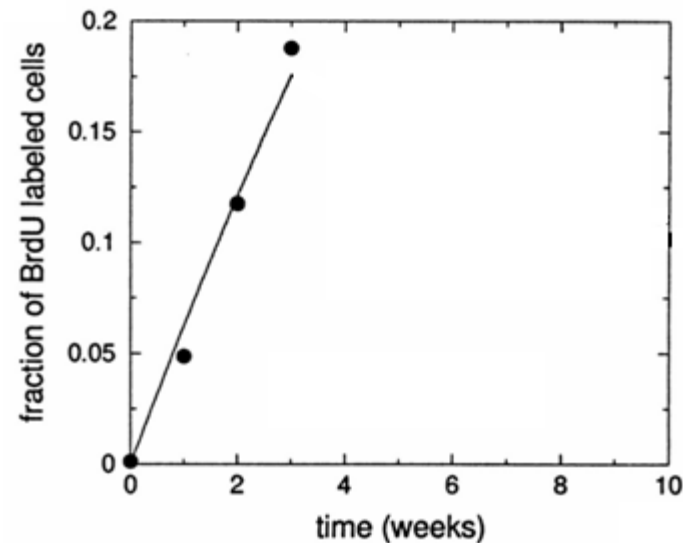
Model of BrdU Labeling

Label is administered continuously over some time-period

B) During BrdU administration

$$f_L(t) = A_1(1 - e^{-(d+p)t})$$

$$A_1 = 1 - \frac{s_U}{(s_U + s_L)} \times \frac{(d-p)}{(d+p)}$$

Labeling curve reflects both proliferation AND death

Model Identifiability

A model is identifiable if possible to learn true value of underlying parameter after obtaining enough observations

Identifiable parameters are those which effect the value of the data and can be estimated with some degree of certainty.

Non-identifiable parameters are those which effect the value of the data but which cannot be estimated accurately
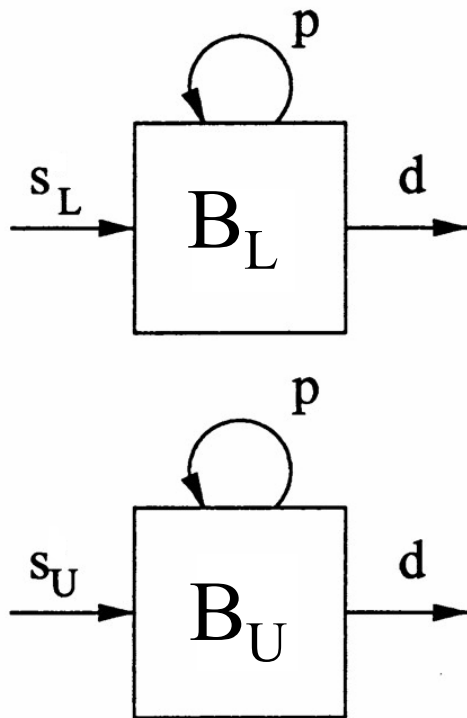
Non-observable parameters are those which don't have an effect on the data.

Cannot estimate both proliferation AND death
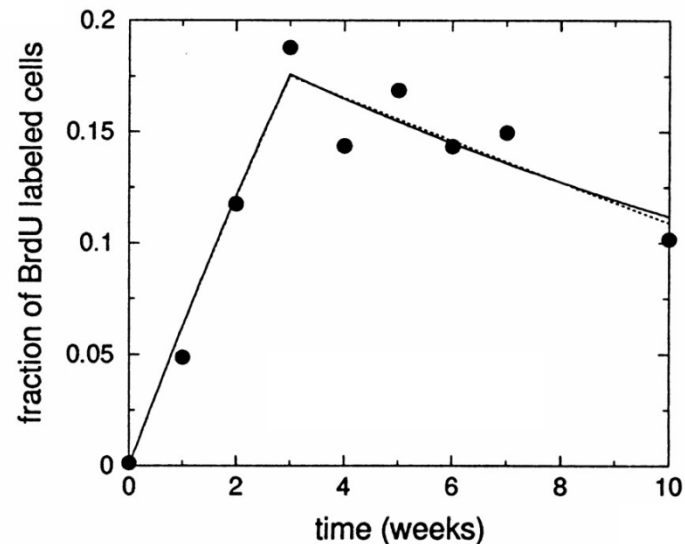
Model of BrdU DE-Labeling

Stop administering label after some time (t_e)

C) After BrdU administration
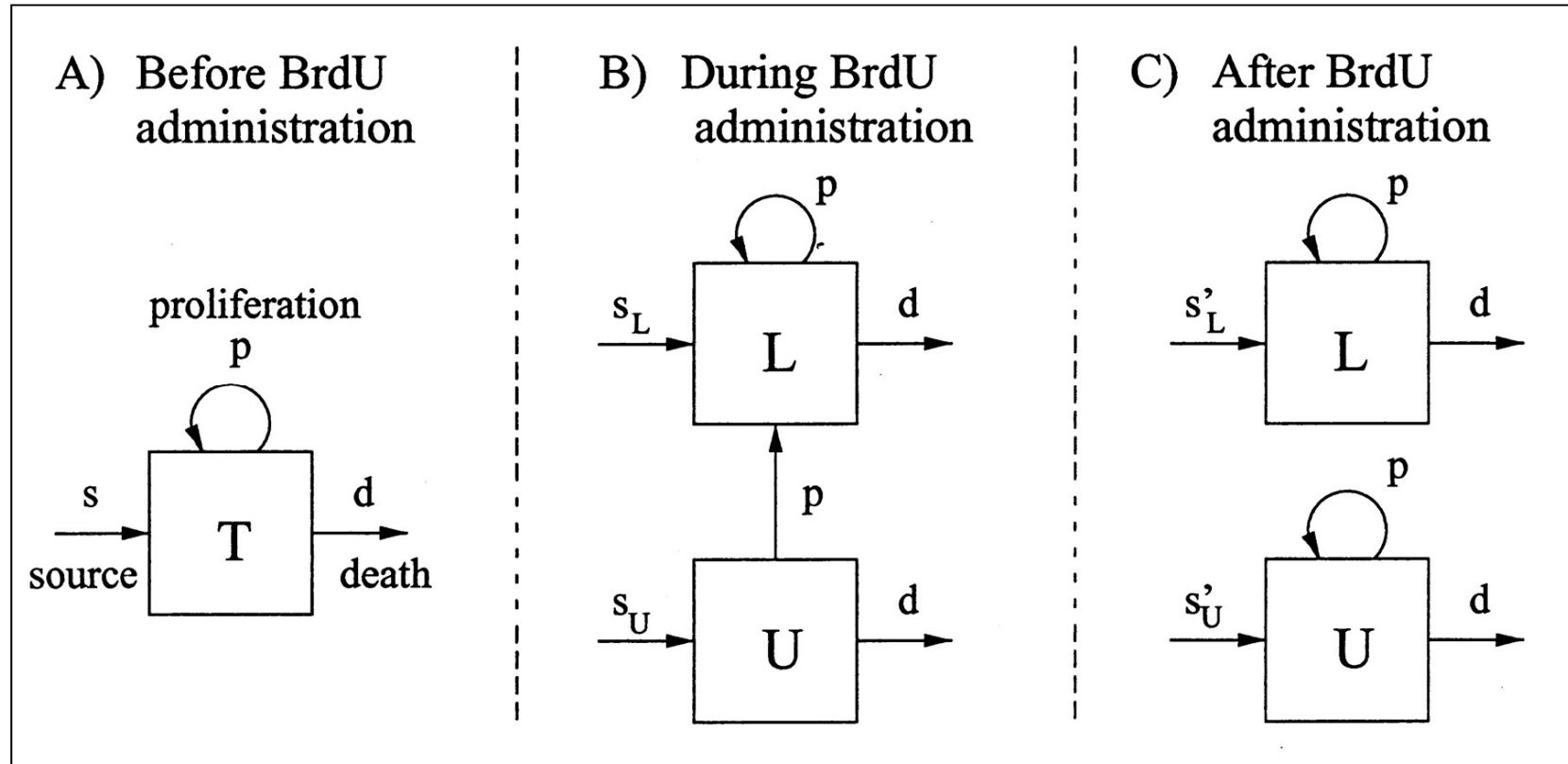
$$f_L(t) = A_2 + A_3 e^{-(d-p)(t-t_e)})$$

$$A_2 = \frac{s_L}{(s_U + s_L)}, A_3 = f_L(t_e) - A_2$$

Now, we can estimate BOTH proliferation AND death

Model of BrdU Labeling

Model changes with experiment

A) Before BrdU administration

proliferation
p

source s → T → d death

B) During BrdU administration

p

s_L → L → d

p

s_U → U → d

C) After BrdU administration

p

s'_L → L → d

p

s'_U → U → d

We can express these as sets of ordinary differential equations

Characteristics of a
Good Inverse Model

- Fit is good—model should be able to adequately describe a relatively noise-free data set   (of course a poor fit provides some insight also)

- Model parameters are unique
  - Theoretically identifiable for noise-free data
  - Well-determined model parameters in presence of measurement noise

- Values of parameter estimates are consistent with hypothesized physical or physiologic meanings and change appropriately in response to alterations in the actual system

Six Steps for Inverse-Modeling of Data

1. Select an appropriate mathematical model
   - Polynomial or other functional form
   - Based on underlying theoretical equations

2. Define a "figure of merit" function
   - Measures agreement between data & model for given parameters

3. Adjust model parameters to get a "best fit"
   - Typically involves minimizing the figure of merit function

4. Examine "goodness of fit" to data
   - Never perfect due to measurement noise

5. Determine whether a much better fit is possible
   - Tricky due to possible local minima vs. global minimum
   - F-test for comparing models of different complexity

6. Evaluate accuracy of best-fit parameter values
   - Provide confidence limits and determine uniqueness
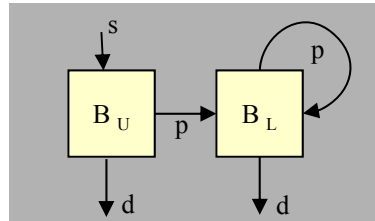   - Assess physical reasonability of estimated parameter values

(Costa, Kleinstein and Hershberg, Sci Signal. 2011)

Interaction of Computation & Experiment

Compare simulation and experiment using least-squares objective
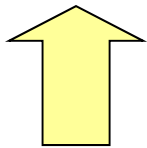
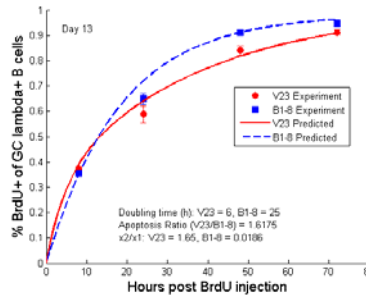**Experimental Observations**
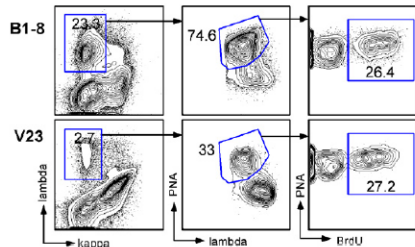
Computational Model

Least-squares objective function

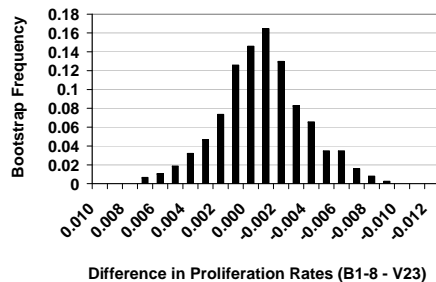$$E = \sum_i \frac{(y_i - \hat{y}_i)^2}{VAR(y_i)}$$

New Experiments

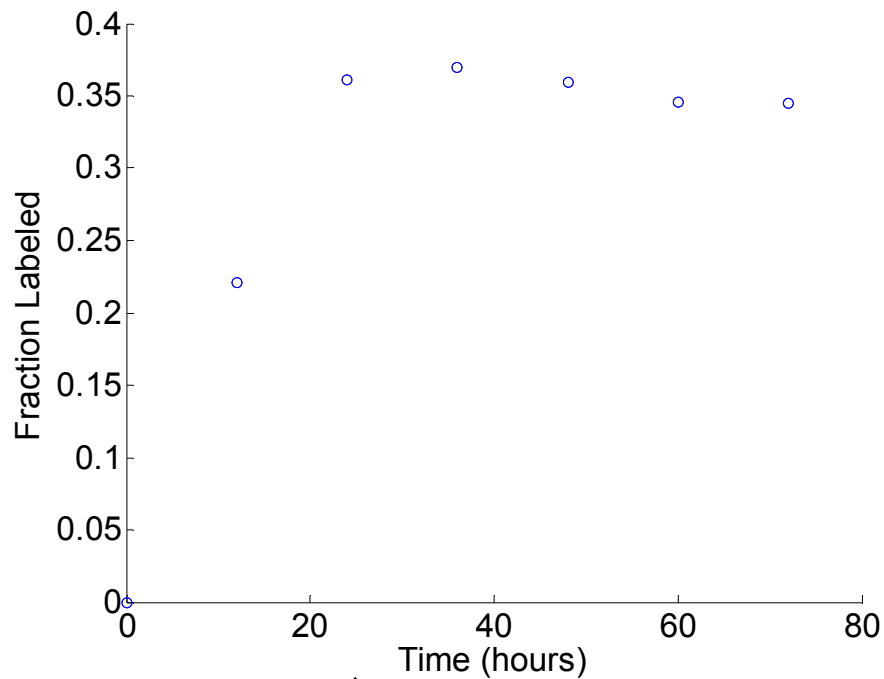**Fit Model to Data**
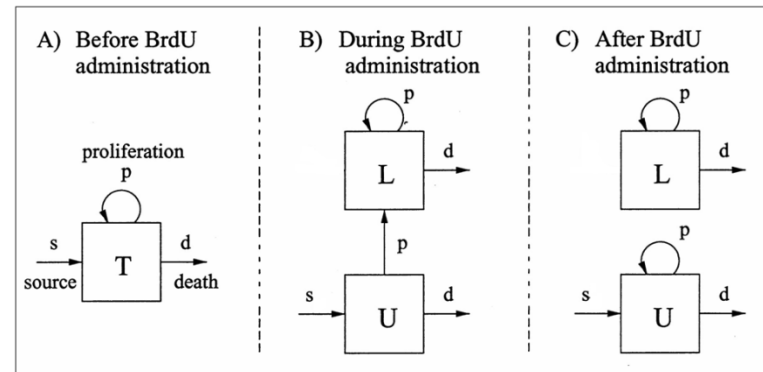
Model Predictions

Bootstrapping Confidence Intervals

Continuous cycle of modeling and experimentation

Simulated Experiment

Demonstrate full cycle of fitting model to data to estimate parameters

Parameters used to create synthetic data

s = 0.003 per hour

p = 0.01 per hour

d = p + s (to achieve steady state)

Random noise added to each data point

BrdU withdrawn

How can we estimate flow/proliferation/death rates?

Numerical solution to ODEs

$$y'(t) = f(t, y(t)), \qquad y(t_0) = y_0,$$

$$y'(t) \approx \frac{y(t+h) - y(t)}{h},$$

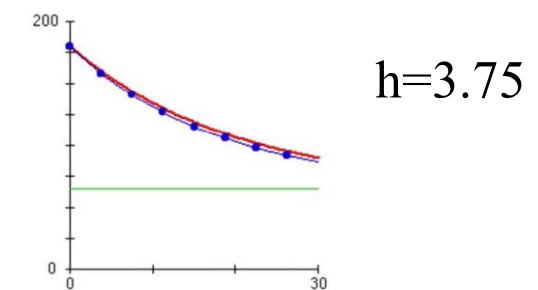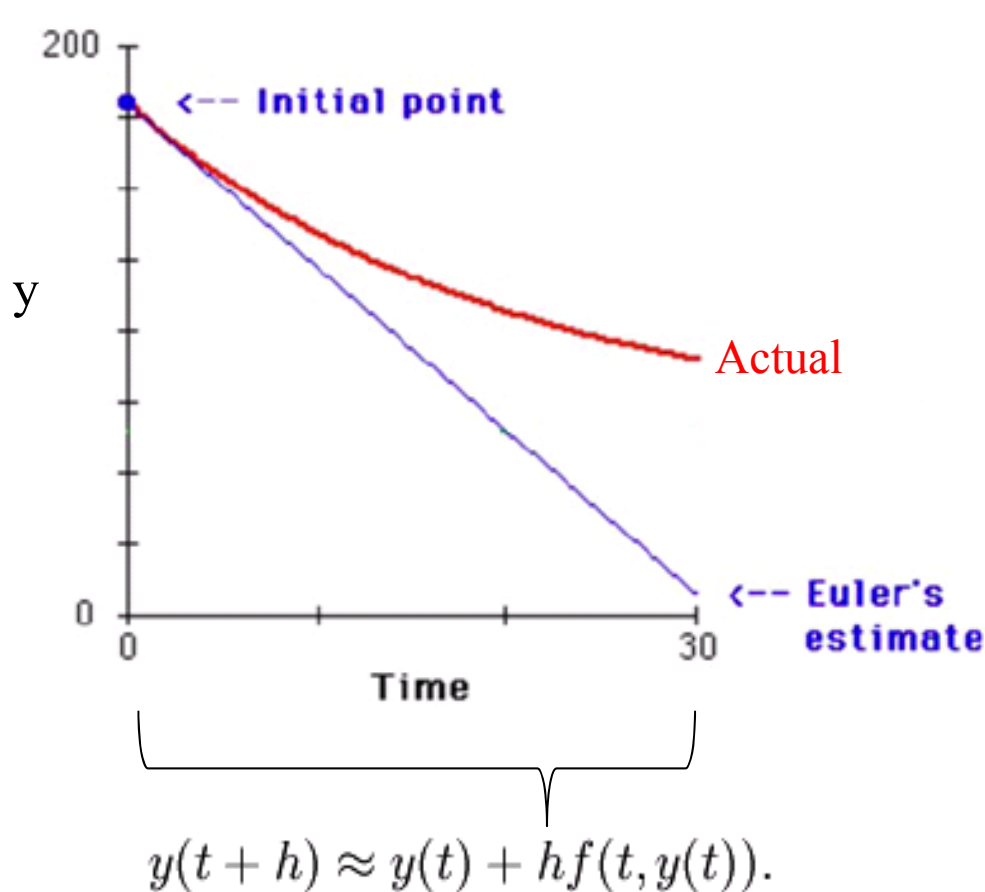$$y(t+h) \approx y(t) + hf(t, y(t)).$$

Leonhard Euler
(1707-1783)

From any point on curve, find approximation of nearby point on curve by moving a short distance along a line tangent to the curve

Numerical solution to ODEs: Euler Method
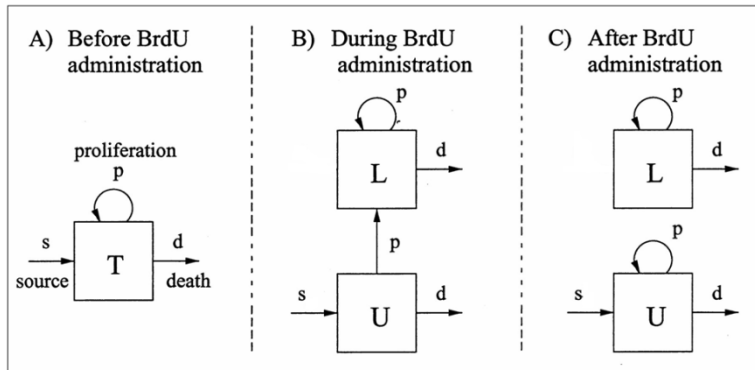
From any point on curve, find approximation of nearby point on curve by moving a short distance along a line tangent to the curve

$$y(t + h) \approx y(t) + hf(t, y(t)).$$

Much better ways to do this in practice. Eg, Runge-Kutta

Simulating the BrdU Labeling Model

A) Before BrdU administration
B) During BrdU administration
C) After BrdU administration

Yin = [1 0];   % Initial Conditions [unlabeled labeled]

pr = [s p d tau]; % Model Parameters

t = [0,12,24,36,48,60,72];   % Times to evaluate

[T,Y] = ode45(@fode,t,Yin,opts,pr);

fl = Y(:,2) ./ sum(Y,2); % Fraction labeled

```
function dy = fode(t, y, pr)

s = pr(1); p = pr(2); d = pr(3); tau = pr(4);

U = y(1); L = y(2);

dy = zeros(2,1);  % Vector of derivatives

if (t<tau)  % During BrdU Administration (B)

   dy(1) =  s - p.*U - d.*U;         % dbU/dt

   dy(2) =  2.*p.*U + p.*L - d.*L;   % dbL/dt

else       % After BrdU Administration (C)

   dy(1) =  s + p.*U - d.*U;         %dbU/dt

   dy(2) =       p.*L - d.*L;        %dbL/dt

end
```
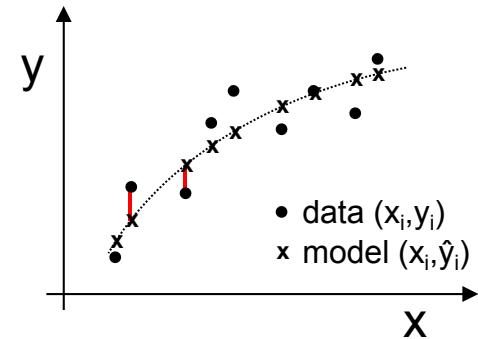
Simple models can be solved analytically -- faster

Least-Squares Error Minimization

- Goal is to fit *N* data points (x_i, y_i) i=1..N

- The model is a function with *M* adjustable parameters a_k, k=1..M used to generate *N* model points $(x_i, \hat{y}_i)$

$$\hat{y}_i = \hat{y}(x_i, a_1..a_M)$$

- The <u>residual</u> measures the difference between a data point and the corresponding model estimate

$$y_i - \hat{y}(x_i, a_1..a_M)$$

- Since residuals can be positive or negative, a sum of residuals is <u>not</u> a good measure of overall error in the fit

$$\sum_{i=1}^{N} [y_i - \hat{y}(x_i, a_1..a_M)]$$

- A better measure is the sum of squared residuals, *E*, which is only zero if each and every residual is zero

$$E = \sum_{i=1}^{N} [y_i - \hat{y}(x_i, a_1..a_M)]^2$$

Maximum Likelihood Estimation

- Not meaningful to ask "What is the probability that my set of model parameters is correct?"
  - Only one correct parameter set ➔ Mother Nature!
- Better to ask "Given my set of model parameters, what is the probability that this data set could be obtained?"
  - What is the <u>likelihood</u> of the parameters given the data?
- Inverse modeling is also known as "maximum likelihood estimation".

(Costa, Kleinstein and Hershberg, Sci Signal. 2011)

Fitting the Model to Experimental Data

Compare simulation and experiment using least-squares objective
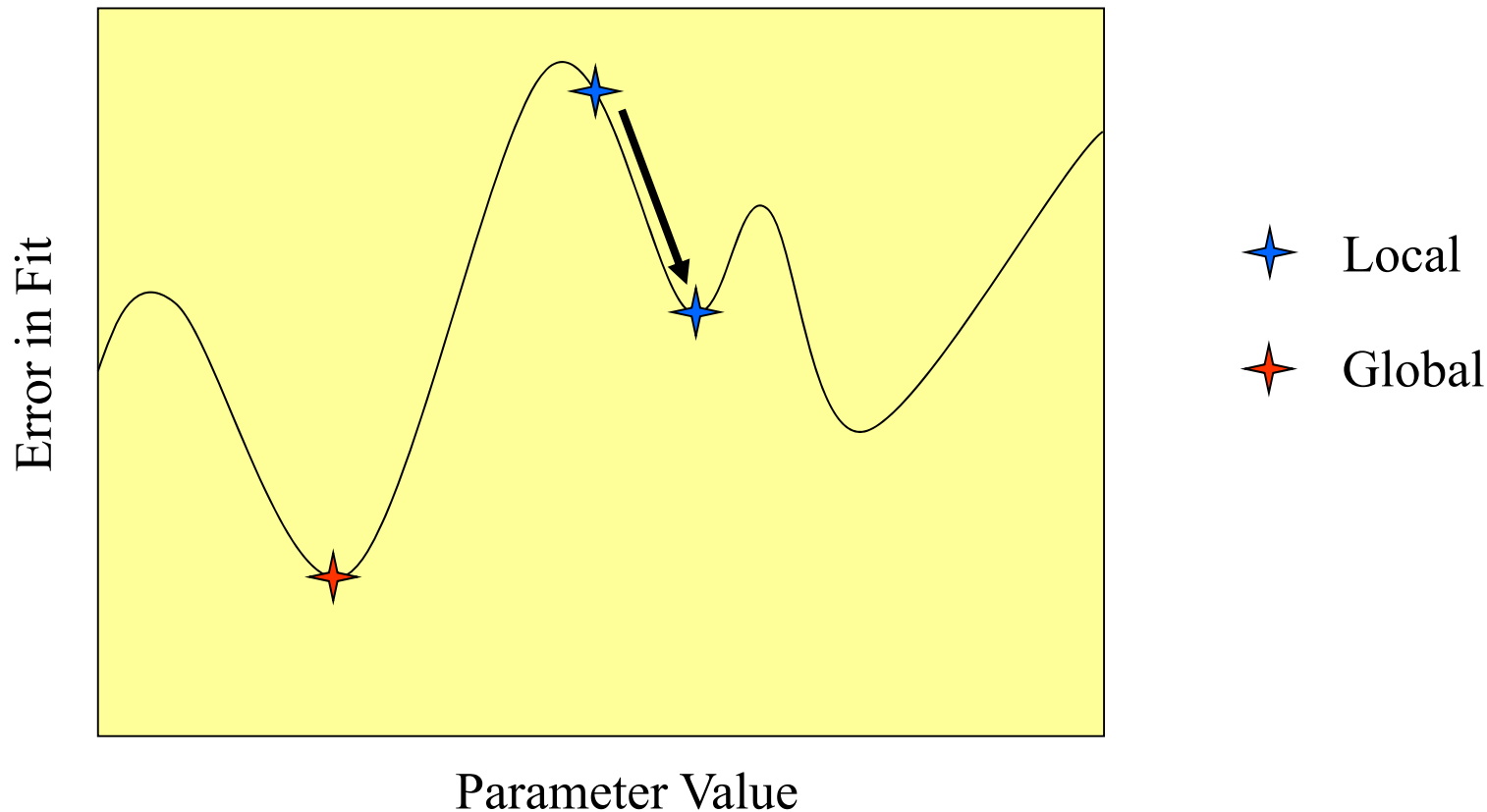

$\}e_i$
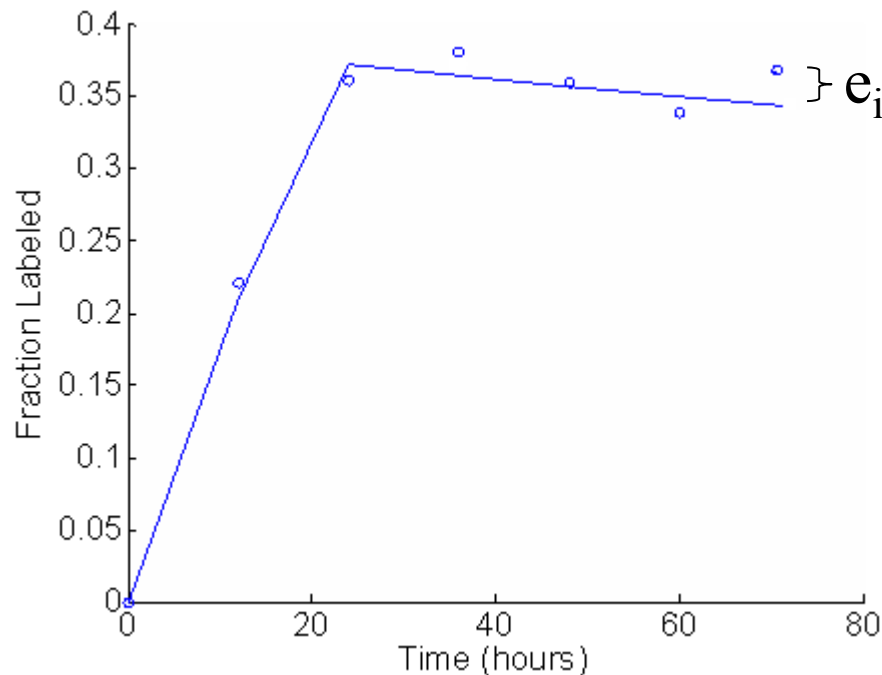
Least-squares objective function

$$E = \sum_i \frac{(y_i - \hat{y}_i)^2}{VAR(y_i)}$$

Find parameters to minimize objective

Many options for how to optimize the fit

Local and Global Optimization

The error function depends on M model parameters, and can be thought of as an M-dimensional "surface" of which we seek the minimum

Error in Fit

Parameter Value

Local

Global

Local optimization techniques find optimal fit around given starting point
Global optimization attempts to avoid local minima

Fitting Models to Data in MATLAB

Several optimization functions available in many programming languages

A) Before BrdU administration — B) During BrdU administration — C) After BrdU administration
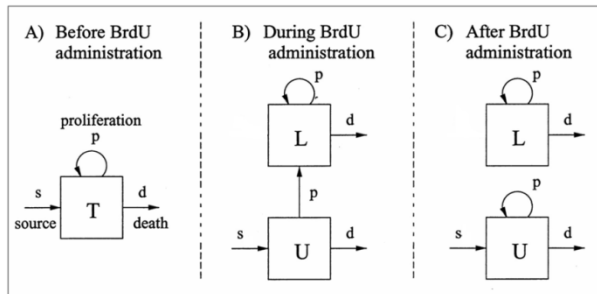
pri = [.01 .01]; %Initial guess for parameter values to be fitted [s p]

[pr,fval,exitflag] = lsqnonlin (@efun,pri,[],[],options,fl_observed,t,tau);

s = pr(1); p = pr(2); % Optimal parameter values

Optional parameters

function error = efun (pr,fl_observed,t,tau)

 **s = pr(1); p = pr(2); d = s+p;**   % Assume steady-state

  [fl_predicted] = labelBrdU(s,p,d,tau,t);   % Function that simulates model

  error = sum((fl_predicted-fl_observed).^2); % Least-squares objective

lsqnonlin, fminsearch, fmincon, fminbnd

Goodness of Fit and the Residuals Plot

- A high correlation can exist even for a model that systematically differs from the data (all 3 examples have $r^2 > 0.99$)

- One must also examine the distribution of residuals—a good model fit should yield residuals equally distributed along x and normally distributed around zero with no systematic trends, as in A rather than B or C
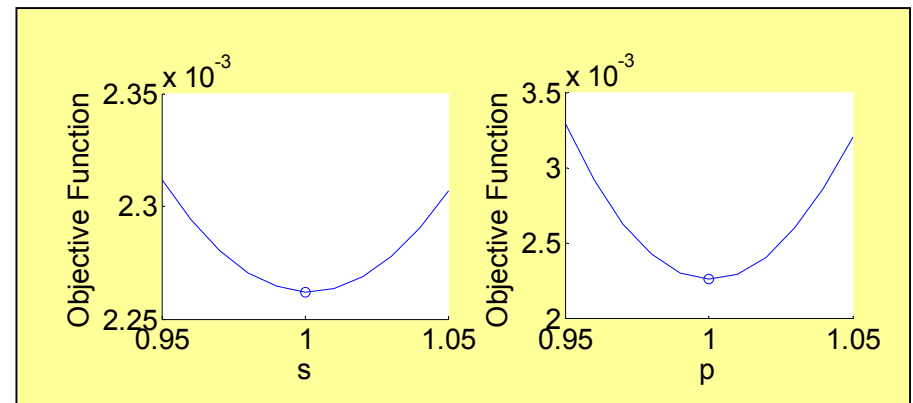

model fits   residuals
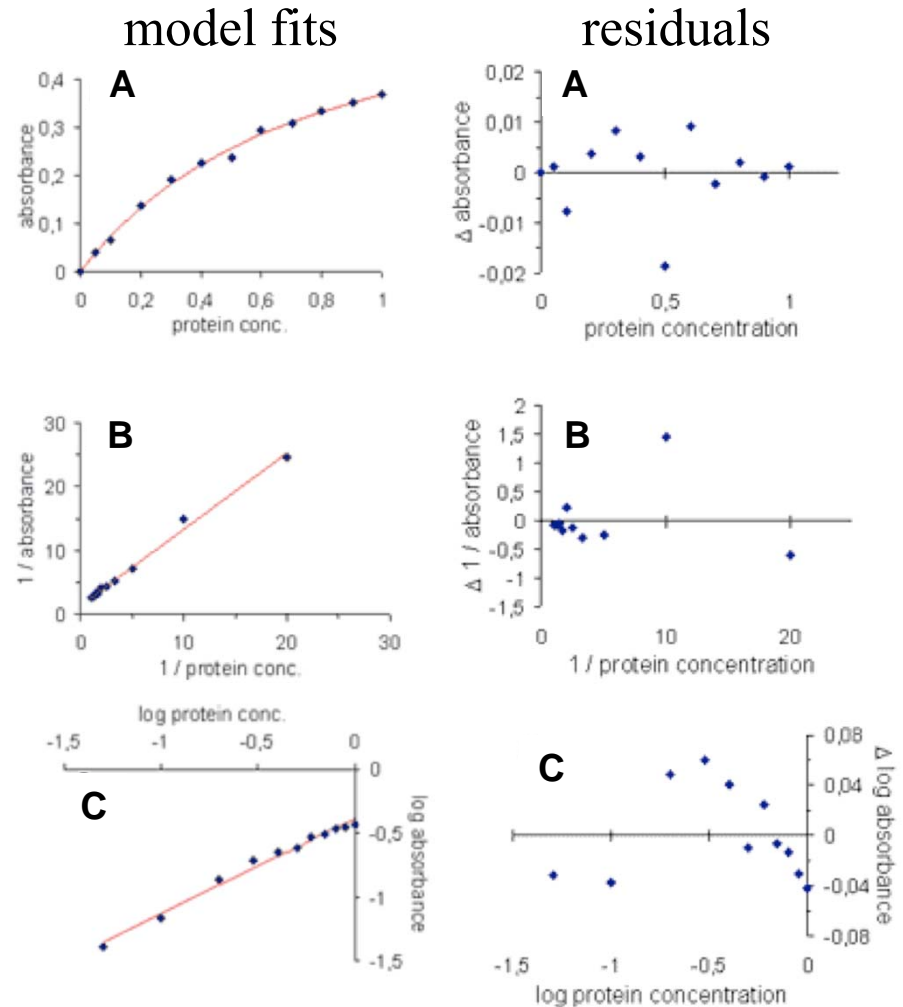
adapted from Lobemeier, 2000

Optimal Parameter Estimates

Least-squares fit using lsqnonlin in MATLAB

Parameter estimates

s = 0.002 per hour
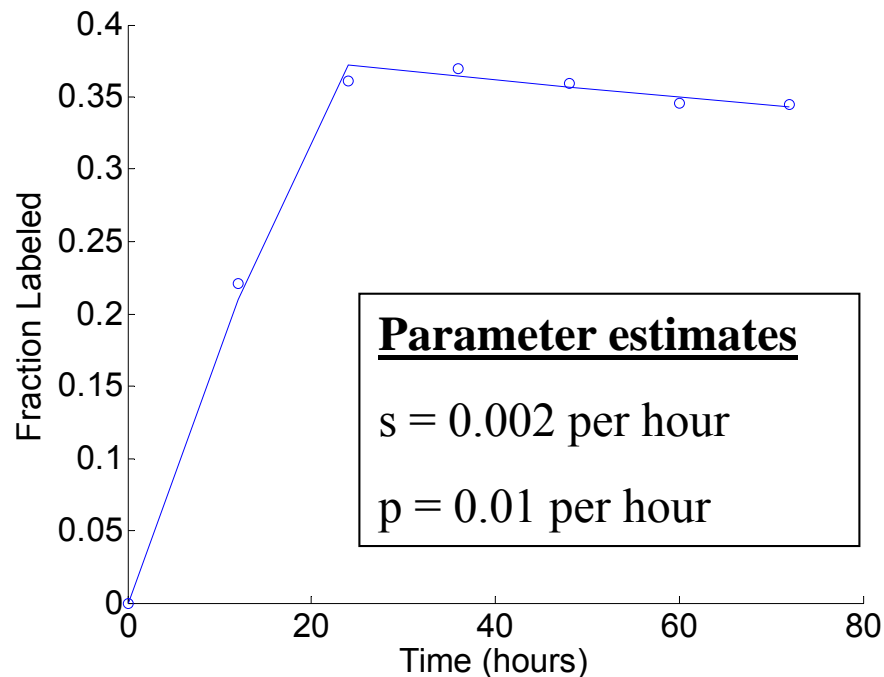
p = 0.01 per hour

Plot local curvature to check minimization…
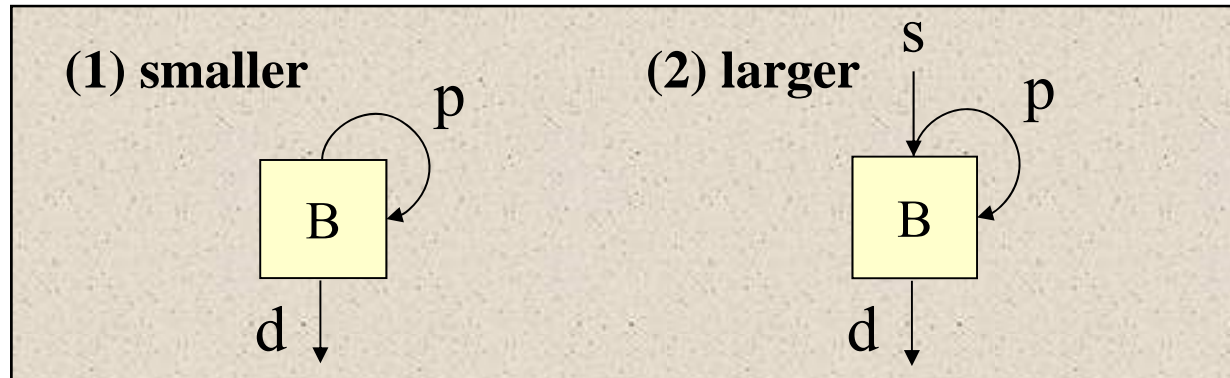
Recall, parameters used to create data:

s = 0.003 per hour

p = 0.01 per hour

d = p + s (to achieve steady state)

Is inflow necessary to fit the data? Can we use simpler model?

Is inflow (s) significant?

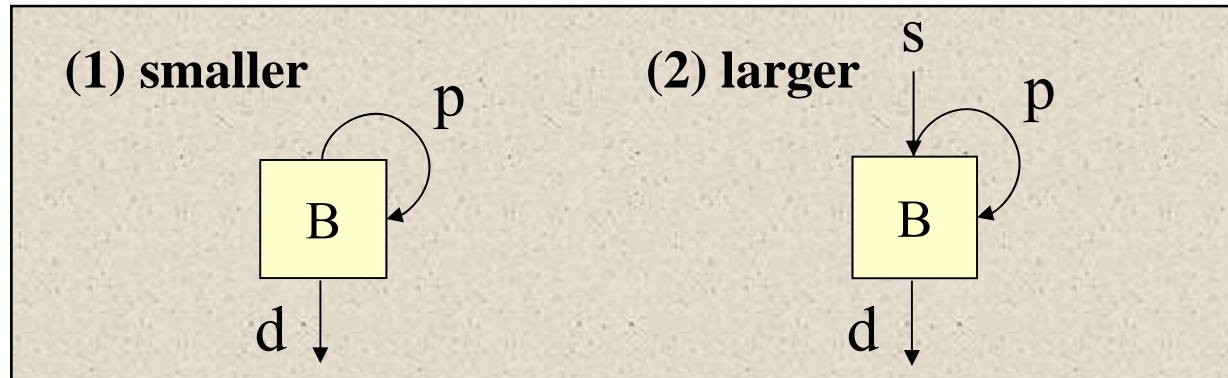Residual Sum of Squares $$RSS = \sum_i (y_i - \hat{y}_i)^2$$

$$F = \frac{\dfrac{RSS_{smaller} - RSS_{larger}}{df_{smaller} - df_{larger}}}{\dfrac{RSS_{larger}}{df_{larger}}}$$

} Reduction in RSS per extra parameter

} Measure of 'noise' in model

Degrees of Freedom $$df = \#\,observations - \#\,parameters$$

F distribution with ($df_{smaller}$-df_{larger}, df_{larger}) degrees of freedom

Is inflow (s) significant?

(1) smaller   p

B

d

(2) larger   s   p

B

d

$$F = \frac{\left. RSS_{smaller} - RSS_{larger} \middle/ df_{smaller} - df_{larger} \right.}{\left. RSS_{larger} \middle/ df_{larger} \right.}$$

} Reduction in RSS per extra parameter

} Measure of 'noise' in model

	Observations	Parameters	RSS	F test (1-fcdf in MATLAB)
(1) No flow (s=0)	6	1	9.38e-7	
(2) Including flow	6	2	0.95e-7	**53.1 (p<0.0004)**

Inflow (s) is important to explain observations

Comparing Two Model Fits

- The number of data points, N, must exceed the number of model parameters, M, yielding the degrees of freedom ($DOF = N-M$)

- Increasing M using a more complex model will generally improve the quality of fit and reduce RSS

- Increasing MSE with decreasing RSS can reveal an over-parameterized model

$$M \leq N-1$$

$$MSE = \frac{RSS}{N-M} = \frac{RSS}{DOF}$$

- An F-statistic can be computed to compare the results of two model fits
  - F ~ 1, the simpler model is adequate
  - F > 1, the more complex model is better, or random error led to a better fit with the complex model
  - P-value defines the probability of such a "false positive" result (lookup in F table)
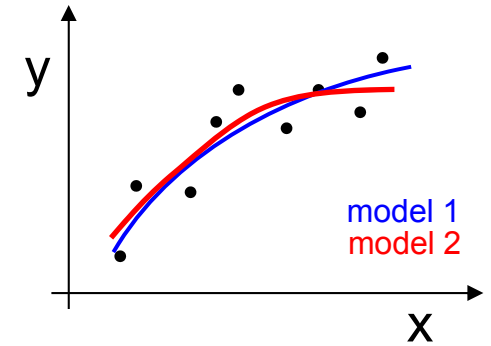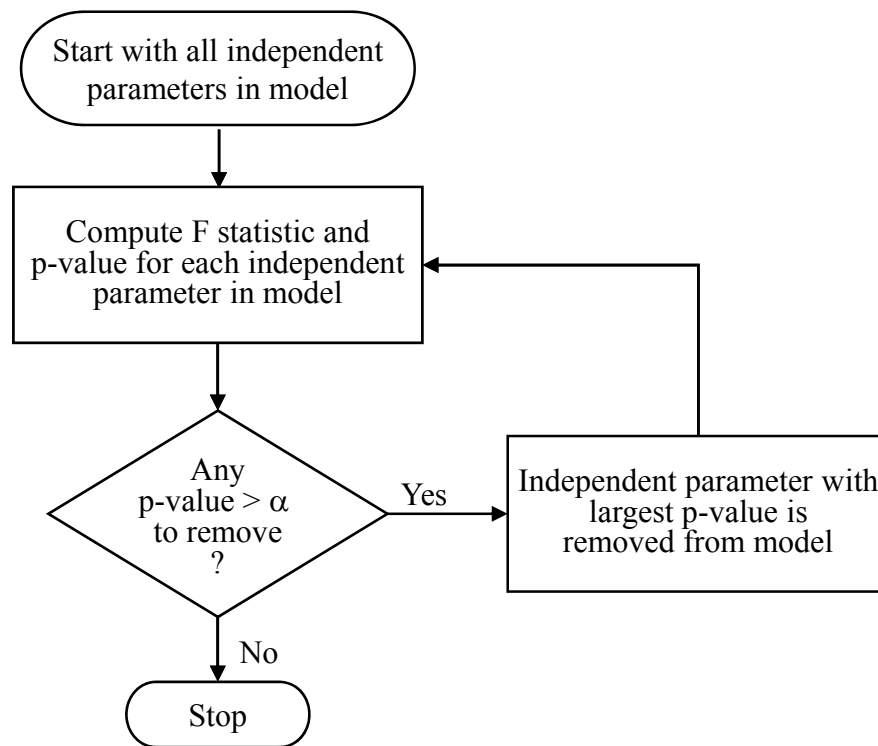
Building models with variable selection

F statistic determines if variable added or deleted from model

Backward Elimination

Other Variations:

Start with all independent parameters in model

Compute F statistic and p-value for each independent parameter in model

Any p-value > α to remove?

Yes → Independent parameter with largest p-value is removed from model

No

Stop

Forward selection: adds variables one at a time as long as significant F test.

Stepwise procedure: allows for removal of a parameter at each step

No guarantee that globally optimal model with be found
(need all subsets, but prohibitive for large parameter space)

How much confidence to put in estimate?

Construct confidence intervals for model parameters

Parameter estimates

s = 0.002 per hour

p = 0.01 per hour
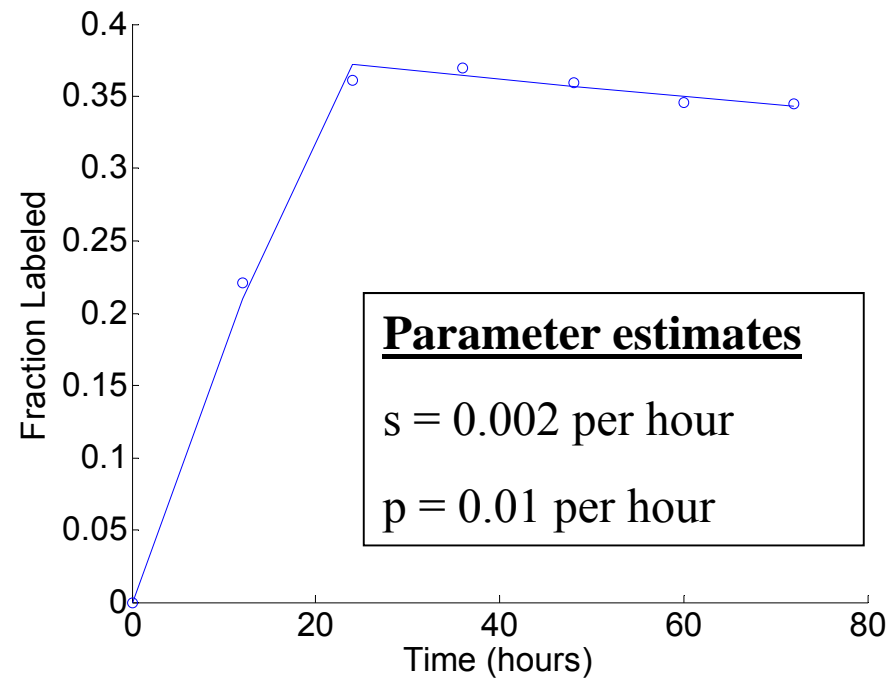
Estimate uncertainty given limited number of experimental observations

Accuracy of Estimated Model Parameters

Underlying true set of model parameters ($\mathbf{a}_{\text{true}}$) known to Mother Nature but hidden from the experimenter

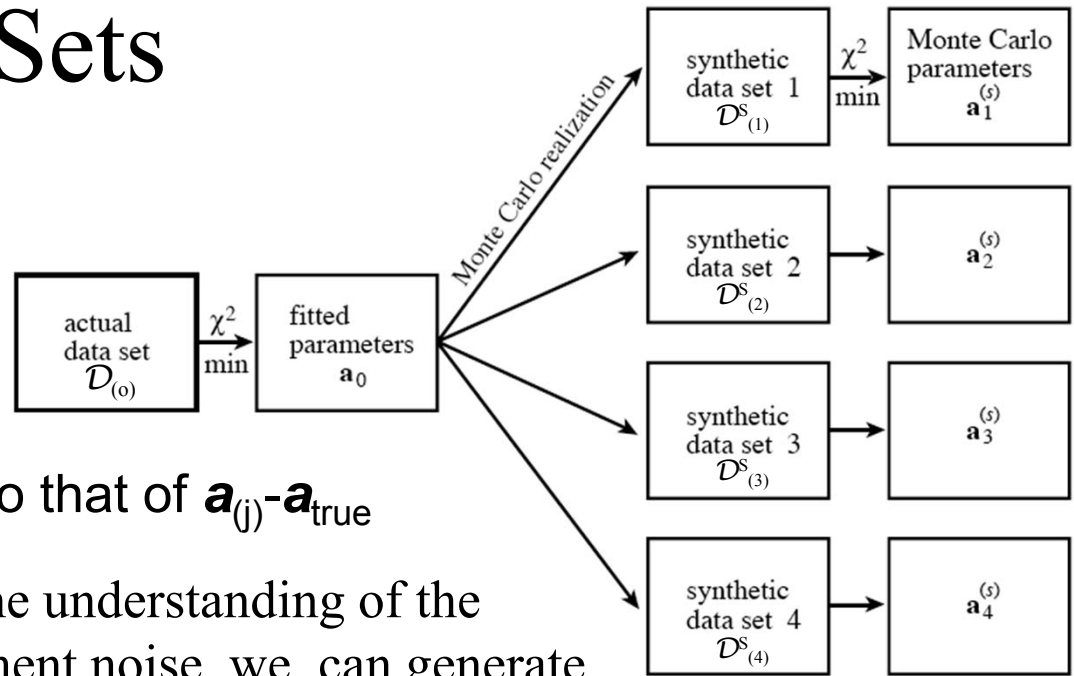- True parameters are statistically realized as measured data set $\mathcal{D}_{(o)}$
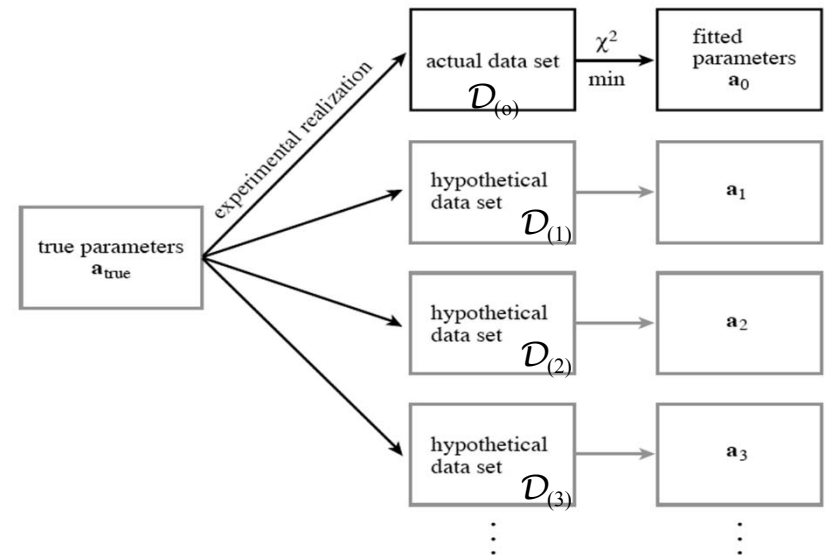
from Numerical Recipes online

- Fitting $\mathcal{D}_{(o)}$ yields estimated model parameters $\mathbf{a}_{(o)}$
- Other experiments could have resulted in data sets $\mathcal{D}_{(1)}$, $\mathcal{D}_{(2)}$, etc. which would have yielded model parameters $\mathbf{a}_{(1)}$, $\mathbf{a}_{(2)}$, etc.
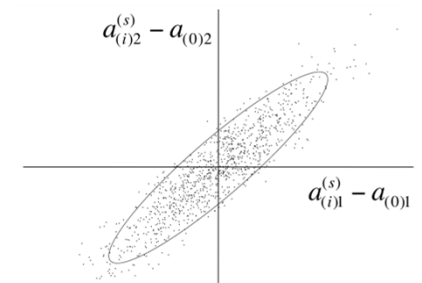
Estimate probability distribution of $\mathbf{a}_{(i)}$ - $\mathbf{a}_{\text{true}}$ without knowing $\mathbf{a}_{\text{true}}$

Monte Carlo Simulation of Synthetic Data Sets

from Numerical Recipes online

- Assume that if $a_{(0)}$ is a reasonable estimate of a_{true}, then the distribution of $a_{(j)}-a_{(0)}$ should be similar to that of $a_{(j)}-a_{true}$

- With the assumed $a_{(0)}$, and some understanding of the characteristics of the measurement noise, we can generate "synthetic data sets" $\mathcal{D}^S_{(1)}, \mathcal{D}^S_{(2)},\ldots$ at the same x_i values as the actual data set, $\mathcal{D}_{(0)}$, that have the same relationship to $a_{(0)}$ as $\mathcal{D}_{(0)}$ has to a_{true}

- For each $\mathcal{D}^S_{(j)}$, perform a model fit to obtain corresponding $a^S_{(j)}$, yielding one point $a^S_{(j)} - a_{(0)}$ for simulating the desired M-dimensional probability distribution. **This is a very powerful technique!!**

2-parameter probability distribution for 1,000 Monte Carlo simulations

(Costa, Kleinstein and Hershberg, Sci Signal. 2011)

The Bootstrap Method

Estimating generalization error based on "resampling":
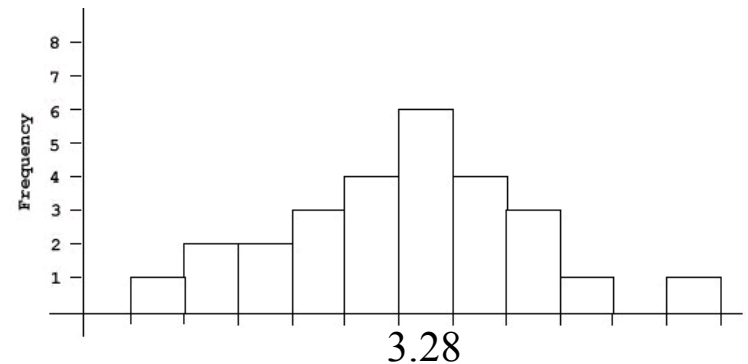Randomly draw datasets with replacement from training data

- If don't know enough about the measurement errors (i.e. cannot even say they are normally distributed) so Monte Carlo simulation cannot be used.

- Bootstrap Method uses actual data set $\mathcal{D}_{(o)}$, with its N data points, to generate synthetic data sets $\mathcal{D}^S_{(1)}$, $\mathcal{D}^S_{(2)}$,… also with N data points.

- Randomly select N data points from $\mathcal{D}_{(o)}$ *with replacement*, which makes $\mathcal{D}^S_{(j)}$ differ from $\mathcal{D}_{(o)}$ with a fraction of the original points replaced by *duplicated* original points.

- Fitting the $\mathcal{D}^S_{(j)}$ data yields model parameter sets $\mathbf{a}^S_{(j)}$ using actual measurement noise.

If sample is good approximation of population, bootstrap method will provide good approximation of sampling distribution of original statistic.

Bootstrap Methods

Randomly draw datasets with replacement from training data

- D = [3.0, 2.8, 3.7, 3.4, 3.5] → average = 3.28
- Bootstrap samples D_N could be:
  - [2.8, 3.4, 3.7, 3.4, 3.5] → 3.36
  - [3.5, 3.0, 3.4, 2.8, 3.7] → 3.28
  - [3.5, 3.5, 3.4, 3.0, 2.8] → 3.24
  - ...

If sample is good approximation of population, bootstrap method will provide good approximation of sampling distribution of original statistic.