Sensitivity Analysis and Least Squares Parameter Estimation for an Epidemic Model

Alun L. Lloyd

Department of Mathematics
Biomathematics Graduate Program
Center for Quantitative Sciences in Biomedicine
North Carolina State University

Aims of this Practical

1. Learn about simple epidemic model, how it behaves and how to simulate it
2. Derive sensitivity equations for the model
3. Fit the model to data from an outbreak, estimating model parameters
4. Obtain measures of uncertainty for these estimated parameters

Aims of this Practical

1. Learn about simple epidemic model, how it behaves and how to simulate it
   simulating differential equation models in MATLAB
2. Derive sensitivity equations for the model
3. Fit the model to data from an outbreak, estimating model parameters
   minimizing a function in MATLAB
4. Obtain measures of uncertainty for these estimated parameters

SIR Model for Spread of Infection

Compartmental model: Susceptibles, Infectives, Recovereds

$$\boxed{S} \xrightarrow{\text{infection}} \boxed{I} \xrightarrow{\text{recovery}} \boxed{R}$$

$$dS/dt = -\beta SI/N$$
$$dI/dt = \beta SI/N - \gamma I$$
$$dR/dt = \gamma I$$

Ignore births and deaths (e.g. short-lived outbreak)

"Standard incidence" term $\beta SI/N$ $\quad$ β : "transmission parameter"

"well-mixed" population

Assume constant per-capita recovery rate of γ

$1/\gamma$ is average duration of infectiousness

Note: $S + I + R = N$ (constant), so need only worry about S and I

Behavior of SIR Model

Behavior is governed by the value of the ratio $R_0 = \beta/\gamma$

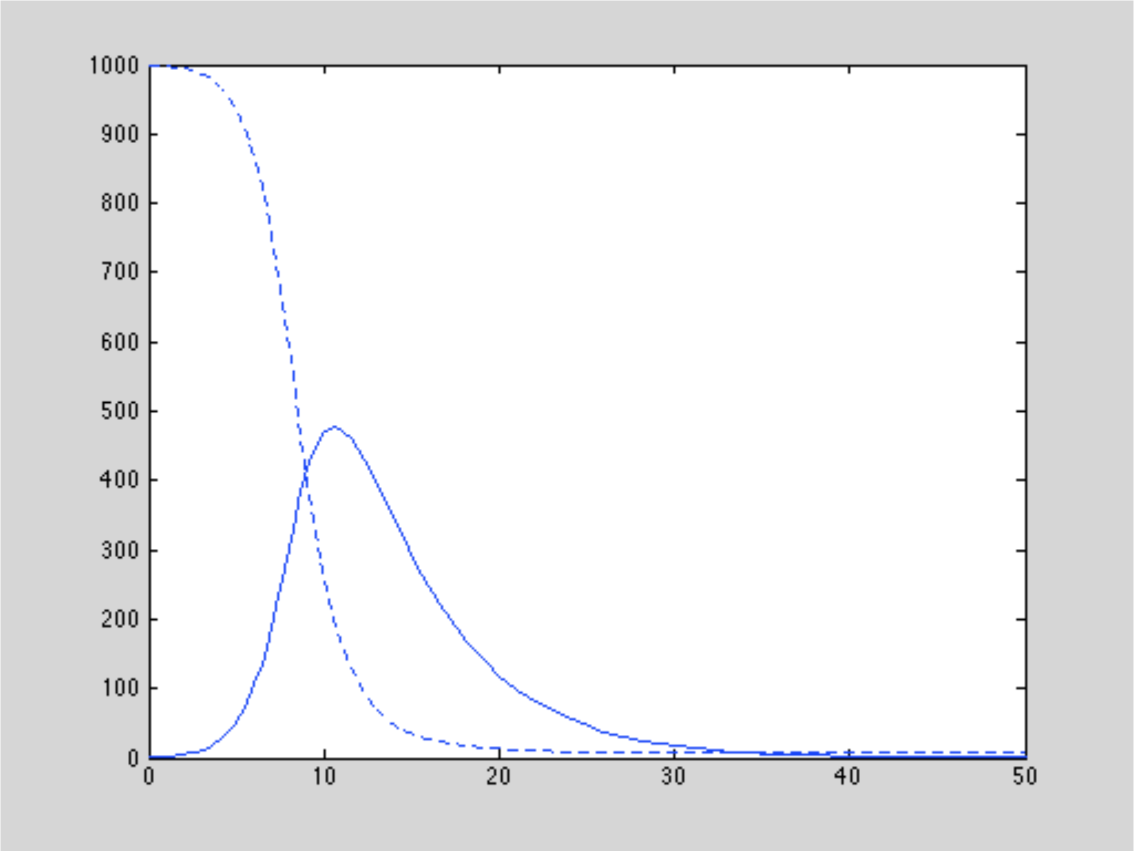
Outbreak can occur if $R_0 > 1$, cannot occur if $R_0 < 1$

$R_0 > 1$ plot:

$\beta = 1$, $\gamma = 0.2$, $N = 1000$
$S(0) = 999$, $I(0) = 1$

$S(t)$: dashed line

$I(t)$: solid line



Behavior of SIR Model

Behavior is governed by the value of the ratio $R_0 = \beta/\gamma$

Outbreak can occur if $R_0 > 1$, cannot occur if $R_0 < 1$

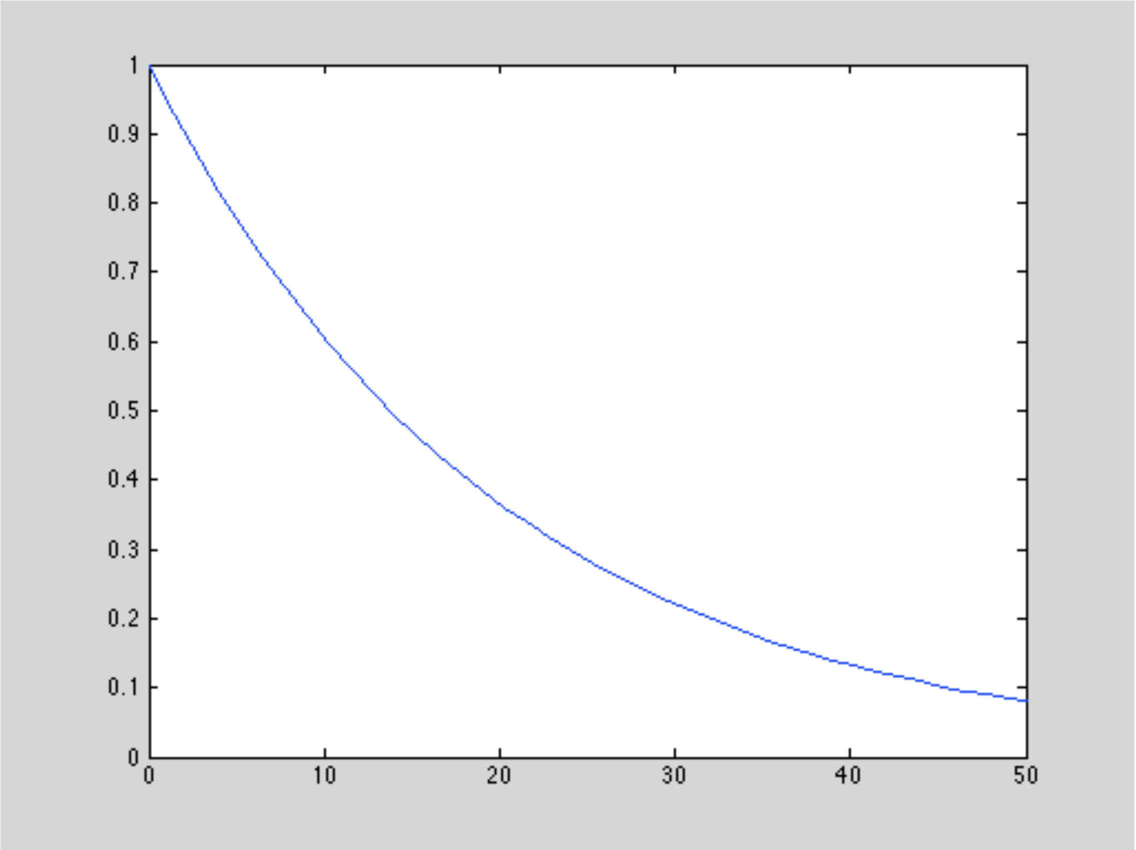
$R_0 < 1$ plot:

$\beta = 0.15$, $\gamma = 0.2$, $N = 1000$
$S(0) = 999$, $I(0) = 1$

$S(t)$: not shown (remains close to 999)

$I(t)$: solid line

note different scale on vertical axis



Simple Analysis of SIR Model in Terms of R_0

Consider dI/dt :

$$\begin{aligned}\frac{dI}{dt} &= \beta SI/N - \gamma I \\ &= \gamma\left(\frac{\beta}{\gamma}\frac{S}{N} - 1\right) I \\ &= \gamma\left(R_0\{S/N\} - 1\right) I \qquad (*)\end{aligned}$$

per-capita transmission maximized when $S \approx N$:

$$\frac{dI}{dt} = \gamma\left(R_0 - 1\right) I$$

I increases if $R_0 > 1$, decreases if $R_0 < 1$

R_0 : basic reproductive number = β x $1/\gamma$ = β x (av. duration of infection)

average number of secondary infections caused by an infectious individual when the population is almost entirely susceptible

Epidemiological Importance of R_0

Can control infection if we can reduce R_0 ($= \beta/\gamma$) below one

(e.g. reduce β or increase γ)

Alternatively, from (*) on previous slide, if we can reduce S/N below $1/R_0$

e.g. vaccinate $p_c = 1 - 1/R_0$ or more of the population

Control is more difficult for a highly infectious agent (e.g. measles, with $R_0 \approx 15\text{-}18$) than for a less infectious agent (e.g. smallpox with $R_0 \approx 5\text{-}7$)

Critical for epidemiologists to estimate R_0 (i.e. β and γ), preferably also getting some idea of reliability of estimate(s)

Typical method used: fit model to some dataset

The Data

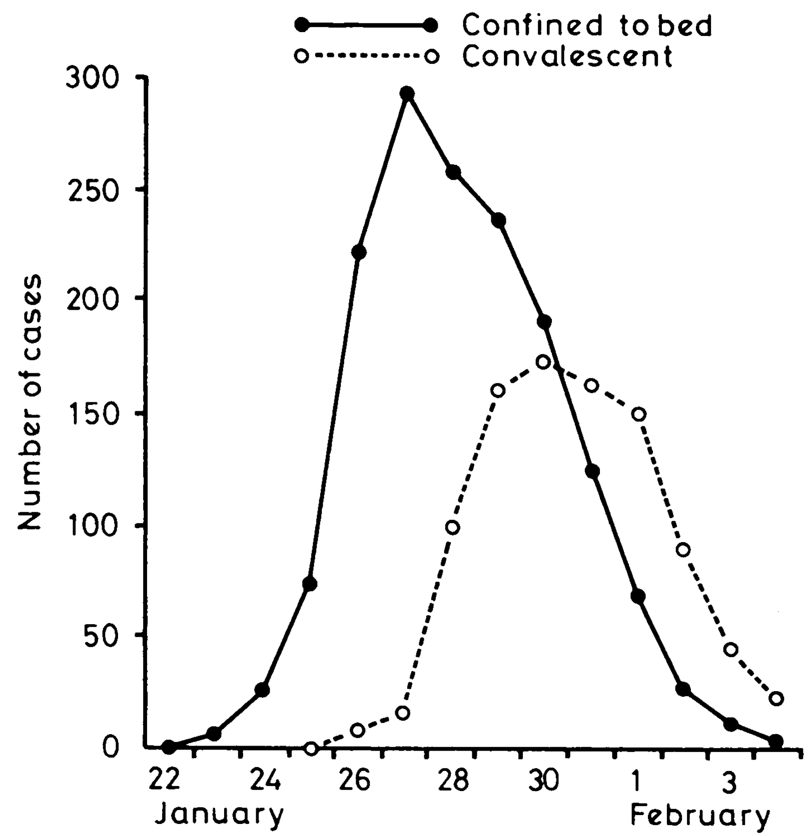
BRITISH MEDICAL JOURNAL 4 MARCH 1978

EPIDEMIOLOGY

Influenza in a boarding school

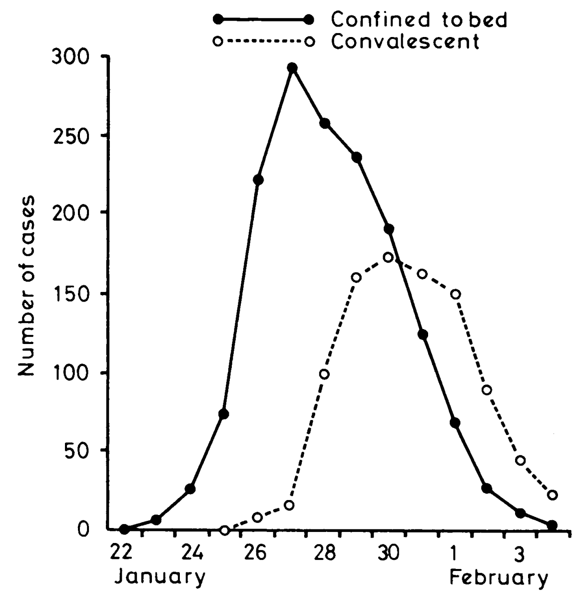
The following notes are compiled by the Communicable Disease Surveillance Centre (Public Health Laboratory Service) and the Communicable Diseases (Scotland) Unit from reports submitted by microbiological laboratories, community physicians, and environmental health officers.

During January an epidemic of influenza occurred in a boarding school in the north of England. A total of 763 boys between the ages of 10 and 18 were at risk, all except 30 being full boarders; the staff were from the surrounding villages. There were 113 boys between the ages of 10 and 13 in the junior house, while the rest were divided into 10 houses of about 60 boys each.



Fitting the SIR Model to Data

We shall take the "confined to bed" time series and view it as the observed trajectory of $I(t)$



14 data points, but we shall imagine that the first one provides us with the true initial condition, leaving us with 13 data points

$N = 763$, $S(0) = 760$, $I(0) = 3$

Seek the values of β and γ that provide the "best fit" to the data

"Best fit" in the sense of minimizing the sum of the squared errors ("error sum of squares"):

$$F(\beta, \gamma) = \sum_i \{I^{\text{observed}}(t_i) - I^{\text{predicted}}(t_i; \beta, \gamma)\}^2$$

Quick Start… If You Have Already Used MATLAB to do Least Squares Fitting

1. Write a function that simulates SIR model

2. Write a function that takes a vector `pars=[beta, gamma]` as input, simulates model for this pair of parameters, compares to data and returns error sum of squares (see notes on slide 20)

3. Minimize this function to find best-fitting values of beta and gamma (slides 23-27)

4. Derive sensitivity equations (slides 16,17), implement them in MATLAB (slide 18) and explore their behavior (see notes on slide 19)

5. Use sensitivity equations with asymptotic statistical theory to obtain estimates of uncertainty in estimated parameters (slides 28,29)

SIR Model : Forward Simulation

Nonlinearity of the transmission term means we cannot find an analytic solution of the model for *S* and *I* in terms of time

Numerically integrate (simulate) model in MATLAB, given a set of parameters and initial values for *S* and *I*

We shall use the `ode45` routine in MATLAB

MATLAB works with vectors, so we shall use the first element (e.g. `y(1)`) to denote *S* and the second (e.g. `y(2)`) to denote *I*

`ode45`

```
[t,y]=ode45(@odefun,tspan,y0,options,pars);
```

`odefun` the name of the function that gives the right sides of our differential equations (replace "odefun" with something more descriptive, but keep "@")

`tspan` vector that specifies the interval of times over which to integrate:

`tspan = [t_initial, t_final]`

or a vector of times at which we wish to obtain output :

`tspan = [t_initial, t1, t2, … , t_final]`

`y0` column vector of initial states (i.e. initial conditions) : `y0 = [ S0 ; I0 ]`

`options` options for the ODE solver, e.g. solution tolerances
use `[ ]` for no options; see `odeset` for information on options

`pars` a vector of parameter values that gets passed to `odefun`

`t` (returned) column vector of times at which output is given

`y` (returned) matrix of numerically calculated values of state variables over time

each row refers to a different time point, each column to a different state variable
e.g. `y(1,:)` are initial states, `y(end,:)` final states,
`y(:,2)` is a column vector of I values at all times — this is what we want to make an $I(t)$ vs t plot

odefun

`function f = odefun(t,y,pars)`

Function `odefun` returns the entries of the right sides of the differential equations, $f(t,\mathbf{y})$, as a column vector

`t` (scalar) value of time at which to evaluate f

`y` column vector containing values of state variables

`pars` a vector of parameter values that gets passed to `odefun`

```
function f = sir_rhs(t,y,pars)

    f=zeros(2,1);

    beta=pars(1);
    gamma=pars(2);
    N=pars(3);

    S=y(1);
    I=y(2);

    f(1)=-beta*S*I/N;
    f(2)=beta*S*I/N-gamma*I;
end
```

need to return a column vector

could eliminate a number of these lines if we worked with y(1), pars(1) etc in the f(1) and f(2) lines

SIR Model Simulation

```
function sir_simulation

    beta=1.0;
    gamma=1.0/5.0;     % five day infectious period
    N=1000.0;

    pars=[beta,gamma,N];

    tspan=[0,50];       % simulate for 50 days

    y0=[999;1];         % one initial infective

    [t,y]=ode45(@sir_rhs,tspan,y0,[],pars);

    plot(t,y(:,2));    % plot prevalence of infection over time

end

function f = sir_rhs(t,y,pars)
    f=zeros(2,1);
    f(1)=-pars(1)*y(1)*y(2)/pars(3);
    f(2)=pars(1)*y(1)*y(2)/pars(3)-pars(2)*y(2);
end
```

Sensitivity Equations

Sensitivities: partial derivatives of state variables with respect to parameters

e.g. $\frac{\partial I}{\partial \beta}(t)$

For $\frac{dx}{dt} = f(x, t; \theta)$, where *x* and *f* are *m* dimensional,
and θ is a *p* dimensional vector of parameters

the *m* by *p* matrix of sensitivities, $\frac{\partial x}{\partial \theta}(t)$, satisfies the ODE system

$$\frac{d}{dt}\frac{\partial x}{\partial \theta} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial \theta} + \frac{\partial f}{\partial \theta}$$

with initial conditions $\frac{\partial x}{\partial \theta}(0) = 0_{m \times p}$

Sensitivity Equations

Sensitivities: partial derivatives of state variables with respect to parameters

e.g. $\frac{\partial I}{\partial \beta}(t)$

For $\frac{dx}{dt} = f(x, t; \theta)$, where *x* and *f* are *m* dimensional,
and θ is a *p* dimensional vector of parameters

the *m* by *p* matrix of sensitivities, $\frac{\partial x}{\partial \theta}(t)$, satisfies the ODE system

$$\frac{d}{dt}\frac{\partial x}{\partial \theta} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial \theta} + \frac{\partial f}{\partial \theta}$$

with initial conditions $\frac{\partial x}{\partial \theta}(0) = 0_{m \times p}$

The matrix $\frac{\partial f}{\partial x}$ is the Jacobian matrix

- differentiate RHS of ODE w.r.t. state vars.

$\frac{\partial f}{\partial \theta}$ is derivative of RHS w.r.t. params

Banks's notation : $s(t) = \frac{\partial x}{\partial \theta}$

Appendix of Capaldi et al. (2012) gives sensitivity equations for SIR model

Numerical Implementation of Sensitivity Equations

Need to solve $\dfrac{d}{dt}\dfrac{\partial x}{\partial \theta} = \dfrac{\partial f}{\partial x}\dfrac{\partial x}{\partial \theta} + \dfrac{\partial f}{\partial \theta}$ together with $\dfrac{dx}{dt} = f(x, t; \theta)$

In total, that's *mp* + *m* equations

For SIR model with 2 states and 2 parameters of interest : 6 equations/quantities to track, arranged as a column vector in MATLAB, with entries

`y(1)`	`y(2)`	`y(3)`	`y(4)`	`y(5)`	`y(6)`
S	I	$\partial S/\partial\beta$	$\partial S/\partial\gamma$	$\partial I/\partial\beta$	$\partial I/\partial\gamma$

Your tasks: 1. Work out the sensitivity equations for the SIR model

2. Code up the sensitivity equations (together with the 2 of the original SIR model) in MATLAB

Behavior of the Sensitivity Equations?

Once you have the sensitivity equations running…

Plot curves of $\partial I/\partial\beta$ and $\partial I/\partial\gamma$ on the same graph

Compare their shapes in the following situations:

1. R_0 just above one, e.g. $R_0 = 1.2$ (take β=0.24, γ=0.2, integrate for 300 time units)
2. Intermediate R_0, e.g. $R_0 = 5$ (take β=1, γ=0.2, integrate for 50 time units)
3. Large R_0, e.g. $R_0 = 12$ (take β=2.4, γ=0.2, integrate for 50 time units)

Does the plot in case (1) say something interesting about our ability to separately estimate β and γ?

Fitting the SIR Model to Data

Two steps:

1. Create function that calculates error sum of squares given values of β and γ
2. Find values of β and γ that minimize this function

Step 1 is a simple modification of the code already created to simulate the SIR model

```
function ESS = error_sum_of_squares(input_pars)
   beta=input_pars(1);
   gamma=input_pars(2);

   tspan=[0:13];      % this vector has entries 0, 1, 2, … , 12, 13 ,
                      % so we get output for each day
   data=[3;6;25;73;222;294;258;237;191;125;69;27;11;4];

   N=763;
   y0=[760 ; 3];
   [t,y]=ode45(@sir_rhs,tspan,y0,[],[beta,gamma,N]);

   diff=data-y(:,2);      % calculate differences between data and predictions
   ESS=sum(diff.^2);      % square entries of diff ( .^2 operator) and then sum
end
```

Fitting the SIR Model to Data

What does the error sum of squares function look like?

Because it's a function of two variables, it's relatively easy to visualize, e.g. using a 3D plot or a contour plot

Might be interesting to look at this before doing minimization…

```
beta_range=[1:0.05:3];
gamma_range=[0.15:0.025:1];

% set up grid of values
[GAMMA,BETA]=meshgrid(gamma_range,beta_range);

% calculate error sum of squares for each point on grid
for i=1:numel(beta_range)
       for j=1:numel(gamma_range)
            ESS(i,j)=error_sum_of_squares([BETA(i,j),GAMMA(i,j)]);
        end
    end

 % do contour plot, with gamma on horizontal, beta on vertical
 figure(1)
 contour(GAMMA,BETA,ESS,20)
```

Fitting the SIR Model to Data

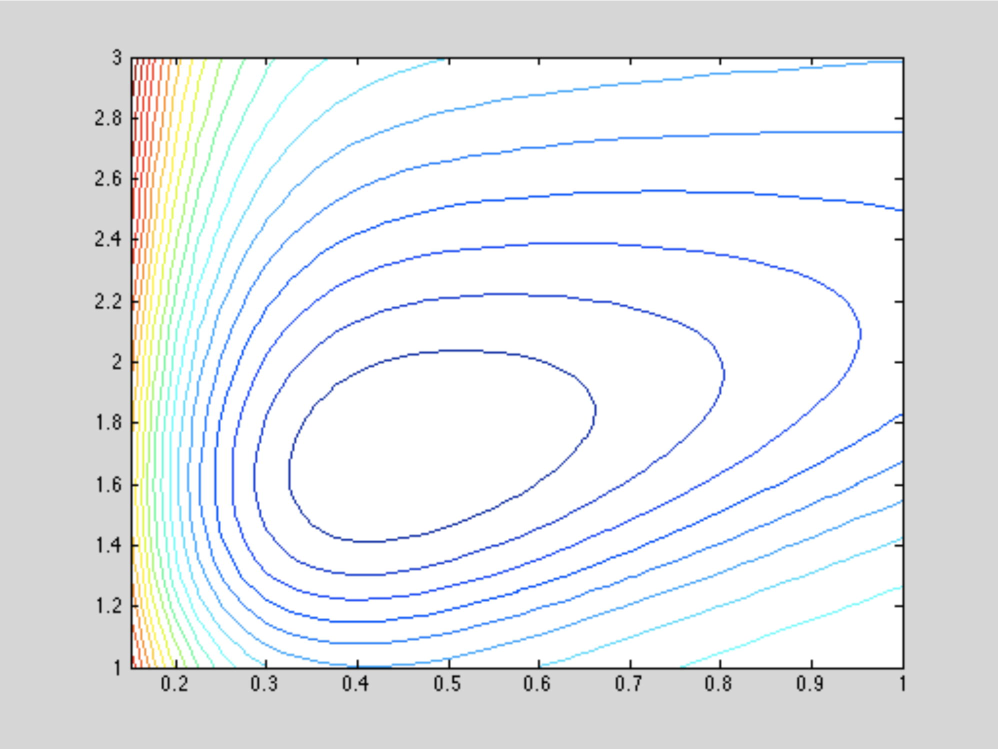
What does the error sum of squares function look like?

Because it's a function of two variables, it's relatively easy to visualize, e.g. using a 3D plot or a contour plot

Might be interesting to look at this before doing minimization…

β on vertical axis

γ on horizontal



Minimizing a Function: `fminsearch`

Optimization is a big area, with lots of different methods that could be used

We shall use MATLAB's fminsearch , which implements the Nelder-Mead direct search simplex algorithm (Nelder & Mead, 1965; see also Walters et al. 1991, Lagarias et al. 1998)

Worth keeping in mind the difficulties (i.e. things that can and do go wrong) with minimization, particularly the possibility that a function has multiple local minima

(Our error sum of squares function looks nice, so we wade in without worrying too much…)

Minimizing a Function: `fminsearch`

```
[x,fval]=fminsearch(@func,x0,options,extra_pars)
```

`func(pars)` is the function whose value is to be minimized
(e.g. our `error_sum_of_squares`)
`pars` is a *p* dimensional vector

`x0` initial guess for the *p* dimensional vector of parameters

`options` contains options for optimization routine (e.g. tolerances, number of allowed iterations and/or function evaluations)

use [ ] if we want to use defaults; see `optimset` for more info

`extra_pars` a vector of other (fixed) parameters we may wish to pass

Returned values:

`x` vector of parameters that minimizes function

`fval` value of function at returned value of x

Example of use of fminsearch

```
function test_minimization
    x0=[1,4];
    [x,fval]=fminsearch(@simple_function,x0)
            % as I don't want to specify options or extra parameters
            % we can skip those arguments
end

function f = simple_function(pars)
    a=pars(1);
    b=pars(2);
    f= 2*(a-2)^2+3*(b-3)^2;
        % embarrassingly simple function, whose minimum is at (2,3)
end
```

Task: Fit SIR Model to Data

Use `fminsearch` on your `error_sum_of_squares` function to find the best-fitting values of β and γ and the error sum of squares

Hint for initial guess at parameters: average duration of influenza infection is about 4 days, and R_0 might be in the ballpark of 8

alternatively: did you get any idea from the contour plot?

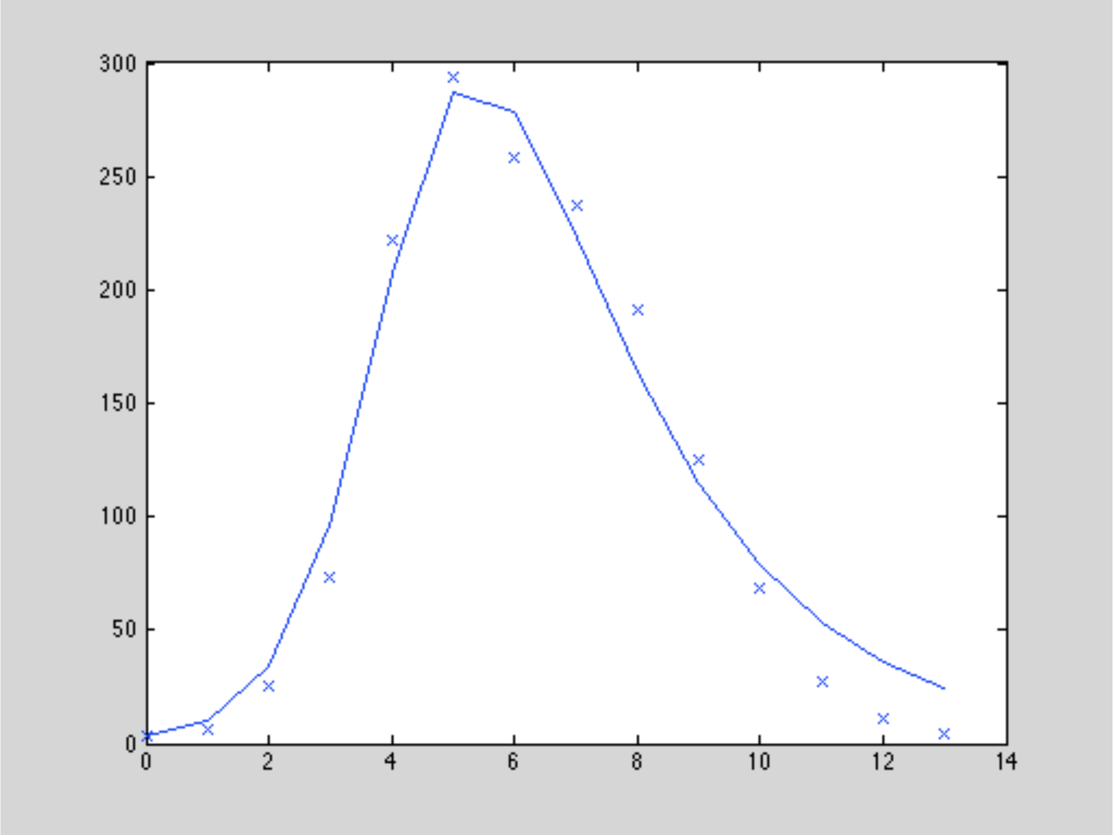
```
[theta_hat,ess]=fminsearch(@error_sum_of_squares,[1,0.2])
```

Plot data and best fitting curve on the same graph

What is our best guess at the value of R_0?

Task: Fit SIR Model to Data

Plot data and best fitting curve on the same graph:



What about uncertainty in our estimates of parameters? Bring the statistical machinery into play…

Uncertainty Estimates for Parameters

Using theory from this morning's talk, our estimate of the variance-covariance matrix for the vector of estimated parameters is

$$\Sigma = \hat{\sigma}^2 \left(\chi^{(n)}(\hat{\theta})^{\mathrm{T}} \chi^{(n)}(\hat{\theta}) \right)^{-1}$$

Here, $\hat{\sigma}^2$ is (minimized value of error sum of squares) / (*n* – *p*)

n (13) data points, *p* (2) estimated parameters

$\chi^{(n)}(\hat{\theta})$ is the *n* x *p* matrix of sensitivities, with entries $\chi^{(n)}(\hat{\theta})_{i\,j} = \dfrac{\partial I(t_i;\hat{\theta})}{\partial \theta_j}$

Need sensitivities of *I* with respect to β and γ at each time point

Then: $\mathrm{SE}(\hat{\beta}) = \sqrt{\Sigma_{11}}$, $\mathrm{SE}(\hat{\gamma}) = \sqrt{\Sigma_{22}}$, $\mathrm{cov}(\hat{\beta},\hat{\gamma}) = \Sigma_{12}$

Uncertainty Estimates for Parameters

Task: Calculate standard errors for estimates of β and γ
and corresponding coefficients of variation (SE/estimate)

Calculate correlation between parameter estimates using $\rho = \frac{\mathrm{cov}(\hat{\beta}, \hat{\gamma})}{\mathrm{SE}(\hat{\beta})\,\mathrm{SE}(\hat{\gamma})}$

Qu.: How does uncertainty in of β and γ translate into uncertainty of $R_0 = \beta/\gamma$?

Non-trivial…

approximate result: $$\mathrm{Var}\left(\frac{\hat{\beta}}{\hat{\gamma}}\right) \approx \left(\frac{\beta_0}{\gamma_0}\right)^2 \left(\frac{\mathrm{Var}(\hat{\beta})}{{\beta_0}^2} + \frac{\mathrm{Var}(\hat{\gamma})}{{\gamma_0}^2} - \frac{2\mathrm{cov}(\hat{\beta}, \hat{\gamma})}{\beta_0 \gamma_0}\right)$$

here, β_0 and γ_0 are our estimates of β and γ

Where to Go Next?

Many possible directions..

1. Include uncertainty in initial condition

    We took $I(0) = 3$. Instead estimate $I(0)$ together with β and γ

        (now have 14 data points)

    Need to include sensitivity of $I(t)$ with respect to $I(0)$

        theory very similar to parameter sensitivities

        see equation 3.62 in Banks's notes

2. What is the appropriate model?

    SEIR model? (individuals have some delay before becoming infectious)

    SEICR model? (model "confinement to bed")

    Time varying parameters? (e.g. action taken to control spread)

    * These models have more parameters… can we estimate them all from 14 data points? identifiability

    * More complex models are more flexible, so tend to fit better: How do we determine if increased fit justifies increased complexity of model? information criteria

References

Anonymous (1978). Influenza in a boarding school. Brit. Med. J. **6112**, 587.

Banks, H.T., et al. (2014). Modeling and Inverse Problems in the Presence of Uncertainty. CRC Press.

Capaldi, A., et al. (2012). Parameter estimation and uncertainty quantification for an epidemic model. Math. Biosci. Eng. **9**, 553-576.

Lagarias, J.C., et al. (1998). Convergence properties of the Nelder-Mead simplex method in low dimensions. SIAM J. Optim. **9**, 112-147.

Nelder, J.A. & Mead, R. (1965). A simplex method for function minimization. Comp. J. **7**, 308-313.

Walters, F.H., et al. (1998). Sequential Simplex Optimization. CRC Press