Fitting Curves to Data

Jake Blanchard
University of Wisconsin - Madison
Spring 2008
The Case

- Suppose we want to project what the US population will be in 2010
- One approach is to fit past data to a curve and extrapolate
## Census Data

<table>
<thead>
<tr>
<th>date</th>
<th>Population (millions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1900</td>
<td>75.995</td>
</tr>
<tr>
<td>1910</td>
<td>91.972</td>
</tr>
<tr>
<td>1920</td>
<td>105.711</td>
</tr>
<tr>
<td>1930</td>
<td>123.203</td>
</tr>
<tr>
<td>1940</td>
<td>131.669</td>
</tr>
<tr>
<td>1950</td>
<td>150.697</td>
</tr>
<tr>
<td>1960</td>
<td>179.323</td>
</tr>
<tr>
<td>1970</td>
<td>203.212</td>
</tr>
<tr>
<td>1980</td>
<td>226.505</td>
</tr>
<tr>
<td>1990</td>
<td>249.633</td>
</tr>
<tr>
<td>2000</td>
<td>281.422</td>
</tr>
</tbody>
</table>
The Curve

\[ \text{population} = Ke^{\alpha t} \]

- Find \( K \) and \( \alpha \) to achieve best fit
Fitting Curves to Data

- Generally curve fitting involves least-squares fits
- We seek parameters in a function that minimize the sum of the squares of the differences between curve and data

\[
F = \sum_{i=1}^{N} \left[ y_i - y(t_i; K, \alpha) \right]^2
\]
Graphical Representation

y(ti)

yi

ti
Linear vs. Nonlinear

- **Linear:**
  
  \[
  y = a + bt \\
  y = a + bt + ct^2 \\
  y = a \sin(t) + b \cos(t) \\
  y = a \sin(3t) \\
  y = ae^{-t}
  \]

- **Nonlinear:**
  
  \[
  y = a \sin(bt) + c \cos(dt) \\
  y = a \sin(bt) \\
  y = ae^{-bt}
  \]
Simple Example

Data:

<table>
<thead>
<tr>
<th>t</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8.38</td>
</tr>
<tr>
<td>2</td>
<td>9.82</td>
</tr>
<tr>
<td>3</td>
<td>10.33</td>
</tr>
<tr>
<td>4</td>
<td>12.14</td>
</tr>
<tr>
<td>5</td>
<td>13.25</td>
</tr>
<tr>
<td>6</td>
<td>14.35</td>
</tr>
<tr>
<td>7</td>
<td>15.57</td>
</tr>
<tr>
<td>8</td>
<td>16.36</td>
</tr>
</tbody>
</table>
Result

![Graph showing a linear relationship between t and y, with a model line and data points.](image-url)
Matlab

- Use `polyfit`
- Fit from figure window
- `fminsearch` for nonlinear fits
Using polyfit in Matlab

- Polyfit fits a polynomial to a set of data
- Polyval allows evaluation of the resulting data in order to plot the results
Sample Commands (straight line)

tdata=1:5;
ydata=[8.38 9.82 10.33 12.14 13.25];
coefs=polyfit(tdata, ydata, 1)
t=1:0.1:5;
y=polyval(coefs,t);
plot(t,y,tdata,ydata,'o')
Demo of Interactive Fit
Practice

• Fit population data to straight line
• What will population be in 2010?
• Repeat for quadratic
• Repeat for cubic
Scaling the “x” data

- Fitting will work better if we “scale” the data
- Our goal is to get a set of x data with a mean of 0 and a standard deviation of 1
- Get this by calculating mean (μ) and std (σ) of the x data and then fit to z, where

\[ z = \frac{x - \mu}{\sigma} \]
More on Scaling

• Wizard for fitting data will do this automatically
Nonlinear Fits

- Nonlinear fits are much more difficult
- There isn’t necessarily a unique solution to the problem
- We have to provide an initial guess for the parameters and then hope the tool can converge to a solution
- This is easily done with the Solver in Excel, but takes a bit more work with Matlab
What we need

- To carry out nonlinear fits, we need the following:
  - A function to evaluate the model for a given set of parameters and for a given time (this is the curve we are fitting to the data)
  - A function to calculate the sum of the squares of the errors between the model and the data (for a given set of fitting parameters)
  - A routine to put everything together
Nonlinear Fits in Matlab (Calling Script)

```matlab
x=[1; 2; 3; 4; 5];
y=[0.9; 7.0; 28.3; 62.1; 122.4];
umpts=max(size(x));
zin(1)=1; %guess for first parameter
zin(2)=3; %guess for second parameter
zout=fminsearch(@(z) sumoferrs(z,x,y), zin)
xplot=x(1):(x(end)-x(1))/(10*numpts):x(end);
yplot=curve(xplot,zout);
plot(x,y,'+',xplot,yplot)
```
function f = curve(x,z)
a = z(1);
n = z(2);
f = a * x.^n;
Routine to Find Sum of Errors

function f=sumoferrs(z, x, y)
f=sum((curve(x,z)-y).^2);
Practice

- Fit population data to exponential
- What will population data be in 2010?

Approach:
- Download `nonlinfit.m`
- Replace data (x and y) in this file with population data from `uspop.m`
- Fix guesses for k and alpha - k=z(1) and alpha=z(2)
- Change curve function to provide 
f=k*exp(alpha*t)
Questions?