

Mathematica for Module 8.3 on "Empirical Models"
File: *EmpiricalModels.nb*

Introduction to Computational Science: Modeling and Simulation for the Scier
Angela B. Shiflet and George W. Shiflet
Wofford College
© 2006 by Princeton University Press

Linear Empirical Model

Norris.dat

National Institute of Standards and Technology monitors," where x is "NIST's measurement of customer's measurement" (<http://www.itl.nist.g>

■ Some of the data

```
pts = {{0.2, 0.1}, {0.4, 0.3}, {0.3, 0.3}}
```

- Alternatively, data in two lists. Form a list of ordered pairs

```
xLst = {0.2, 0.4, 0.3, 0.3};  
yLst = {0.1, 0.3, 0.3, 0.6};
```

- Plot the points (Figure 8.3.1)

```
lp = ListPlot[pts,  
  PlotStyle -> {PointSize[.03]}, AxesLa  
  PlotRange -> {{0, 0.62}, {0, 0.62}}]
```

- Fit function

```
fiteq = Fit[pts, {1, x}, x]
```

- Plot function (Figure 8.3.2)

```
pltfit = Plot[fiteq, {x, 0, .62}]
```

```
Show[lp,pltfiit]
```

■ Plot point at (0.34, 0.365) (Figure 8.3.3)

```
plotPt = ListPlot[{{0.34, 0.365}},  
  PlotStyle -> {PointSize[.04]}, AxesLa  
  PlotRange -> {{0, 0.62}, {0, 0.62}}]
```

```
Show[lp,pltfiit, plotPt]
```

■ Complete set of data

```
x = {0.2, 337.4, 118.2, 884.6, 10.1, 226.5,  
  0.4, 0.6, 775.5, 666.9, 338.0, 447.5,  
  120.2, 0.3, 0.3, 556.8, 339.1, 887.2,  
  669.1, 448.9, 0.5};  
y = {0.1, 338.8, 118.1, 888.0, 9.2, 228.1,  
  0.3, 0.1, 778.1, 668.8, 339.3, 448.9,  
  119.6, 0.3, 0.6, 557.6, 339.3, 888.0,  
  668.4, 449.2, 0.2};
```



```
lp = ListPlot[pts, AxesLabel → {" x ", " y "},  
PlotStyle → {PointSize[.03]}]
```

- Plotting data with a line through the first and last lines helps

```
ln = Show[Graphics[Line[{{xLst[[1]], yLst[[1]]}, {xLst[[Length[xLst]], yLst[[Length[yLst]]}]]],
```

```
Show[lp, ln]
```

```
lp = ListPlot[pts, AxesLabel → {" x ", " y "},  
PlotStyle → {PointSize[.03]}]
```

- Squaring x coordinates does not seem quite enough (Figure 8.

```
lp2 = ListPlot[Transpose[{xLst^2, yLst}],  
PlotStyle → {PointSize[.03]}]
```

- Cubing x coordinates does not seem quite enough (Figure 8.3.

```
lpPower = ListPlot [Transpose [ {xLst ^ 3, yLst} ],  
PlotStyle → {PointSize [ .03 ]}]
```

- Raising x coordinates to the fourth power seems too much (Fi

```
lpPower = ListPlot [Transpose [ {xLst ^ 4, yLst} ],  
PlotStyle → {PointSize [ .03 ]}]
```

- Raising x coordinates to the power 3.5 seems better (Figure 8.

```
lpPower = ListPlot [Transpose [ {xLst ^ 3.5,  
PlotStyle → {PointSize [ .03 ]}]
```

- Plot linear regression line with points (Figure 8.3.10)

```
fitData = Fit [Transpose [ {xLst ^ 3.5, yLst} ],
```

```
plFit = Plot[fitData, {z, 0, 7.5}]
```

```
Show[lpPower, plFit]
```

- Always, eventually plot actual and predicted data together (F

```
Clear[f, x]  
f[x_] := fitData /. z → x3.5
```

```
f[x]
```

```
plData = Plot[f[x], {x, 0, 1.7}]
```

```
Show[lp, plData]
```

```
plData = Plot[f[x], {x, 1.3, 1.7}]
```

```
Show[lp, plData]
```

Solving for y in a One-Term Model

Misra1a.dat

The data is from NIST dental research by D. Misra where x represents pressure and y volume.

(<http://www.itl.nist.gov/div898/strd/nls/data/Misra1a.dat>)

■ Plot data (Figure 8.3.13)

```
xLst = {77.6, 114.9, 141.1, 190.8, 239.9,
        536.8, 593.1, 689.1, 760.0};
yLst = {10.07, 14.73, 17.94, 23.93, 29.61,
        61.01, 66.40, 75.47, 81.78};

pts = Transpose[{xLst, yLst}];
```

```
lp = ListPlot[pts, AxesLabel → {"x", "y"},
              PlotStyle → {PointSize[.03]}]
```

- Raising y coordinates to the power $6/5$ seems linear (Figure 8.

```
lpPower = ListPlot [Transpose [ {xLst, yLst  
PlotStyle → {PointSize [.03]} ]
```

```
fitData = Fit [Transpose [ {xLst, yLst6/5 } ],
```

```
plFit = Plot [fitData, {z, 0, 800}]
```

```
Show [lpPower, plFit]
```

- Always, eventually plot actual and predicted data together. S

```
Clear [f, x]
```

```
f [x_] := (-5.48629 + 0.267869 x)5/6
```

```
f [x]
```

```
plData = Plot[f[x], {x, 77, 770}]
```

```
Show[lp, plData]
```

Multi-term Models

Filip.dat

The data is from NIST research.

(<http://www.itl.nist.gov/div898/strd/nls/data/filip>)

■ Plot data (Figure 8.3.16)

```
xLst = {-6.860120914, -4.324130045, -4.3  
-6.661145254, -6.355462942, -6.11810  
-6.519993057, -6.204119983, -5.85387  
-5.482672118, -5.171791386, -4.85170  
-3.709075441, -3.499489089, -6.30076  
-5.031376979, -4.680685696, -4.32984  
-8.363211311, -8.107682739, -7.82390  
-6.920818754, -6.628932138, -6.32394  
-8.663140179, -8.473531488, -8.24733  
-7.352812702, -7.072065318, -6.77417  
-6.835647144, -6.53165267, -6.224098  
-5.290645224, -4.974284616, -4.64454  
-3.408378962, -3.13200249, -8.726767  
-8.165388579, -7.886056648, -7.58804
```

```

-6.691862621, -6.392544977, -6.06737
-6.065855188, -5.752272167, -5.13241
-3.66174277, -3.2644011};
yLst = {0.8116, 0.9072, 0.9052, 0.9039, 0
0.7975, 0.8162, 0.8515, 0.8766, 0.888
0.8971, 0.9021, 0.909, 0.9139, 0.9199
0.8977, 0.9035, 0.9078, 0.7675, 0.770
0.7971, 0.8329, 0.8641, 0.8804, 0.766
0.7749, 0.7796, 0.7897, 0.8131, 0.849
0.8856, 0.8919, 0.8934, 0.894, 0.8957
0.7739, 0.7681, 0.7665, 0.7703, 0.770
0.8602, 0.8809, 0.8301, 0.8664, 0.883
0.9119, 0.9228};
pts = Transpose[{xLst, yLst}];

```

```

lp = ListPlot[pts,
PlotStyle -> {PointSize[.02]}]

```

■ Model with fourth degree polynomial (Figure 8.3.17)

```

fitData = Fit[pts, {1, z, z2, z3, z4}, z

```

```
plFit = Plot[fitData, {z, -9, -3}]
```

```
Show[lp, plFit]
```

■ Model with tenth degree polynomial (Figure 8.3.18)

```
fitData = Fit[pts, {1, z, z2, z3, z4, z5,
```

```
plFit = Plot[fitData, {z, -9, -3}]
```

```
Show[lp, plFit]
```

■ Plot of tenth degree polynomial model inside and outside range

```
plFit = Plot[fitData, {z, -10, -2}]
```