Computing for Math and Stats

Lecture 16.

Fitting Polynomials

- One can find a polynomial that approximates given data
- Data is in the form of pairs
  - x_i, y_i
- We can have many such pairs.
- If the number of pairs is the same as the number of unknowns (the coefficients of the polynomial) we can use simple linear equations.
- If we have more data than unknowns we use least squares

Least Squares

- A general method, not only for polynomial fitting
- One of the most used methods in science and engineering
- The basis of most other model fitting methods
- Fast, accurate and well understood
- Many variations exist
- Unfortunately does not apply to all problems

Example: first degree model

$$y(x) = a_1 x + a_0$$

$$(x_0, y_0), (x_1, y_1), \ldots, (x_M, y_M)$$

Finding the coefficients

- Set up an expression that is the sum of squares
- If there was a solution the expression would be zero
- Minimize using derivatives w/r to the unknown coefficients
- Simplify to set up the equations
- Solve the equations using linear algebra techniques
- The equations are always linear for this kind of problems

The Expression

$$S = \sum_{i=1}^{M} \left(p\left(x_i\right) - y_i \right)^2$$

$$p\left(x_i\right) = \sum_{j=0}^{N} a_j x_i^j$$

The derivatives

$$\frac{\partial S}{\partial a_k} = \frac{\partial}{\partial a_k} \sum_{i=1}^{M} \left(p(x_i) - y_i \right)^2 = ¿$$

$$\sum_{i=1}^{M} \frac{\partial}{\partial a_k} \left(p(x_i) - y_i \right)^2 = \sum_{i=1}^{M} 2 \frac{\partial p(x_i)}{\partial a_k} \left(p(x_i) - y_i \right)$$

$$\frac{\partial p(x_i)}{\partial a_k} = \sum_{j=0}^{N} \frac{\partial a_j}{\partial a_k} x_i^j = x_i^k$$

$$\frac{1}{2} \frac{\partial S}{\partial a_k} = \sum_{i=1}^{M} x_i^k \left(p(x_i) - y_i \right) = \sum_{i=1}^{M} \sum_{j=0}^{N} a_j x_i^{j+k} - \sum_{i=1}^{M} x_i^k y_i = \sum_{j=0}^{N} a_j \left(\sum_{i=1}^{M} x_i^{j+k} \right) - \sum_{i=1}^{M} x_i^k y_i$$

The Derivatives

$$\sum_{j=0}^{N} a_j \left(\sum_{i=1}^{M} x_i^{j+k} \right) - \sum_{i=1}^{M} x_i^k y_i = 0$$

$$\sum_{j=0}^{N} a_j q_{jk} - b_k = 0$$

$$q_{jk} = \left(\sum_{i=1}^{M} x_i^{j+k} \right) \qquad b_k = \sum_{i=1}^{M} x_i^k y_i$$

$$QA = B \qquad Q = \begin{bmatrix} q_{00} & q_{01} & q_{02} & & q_{0N} \\ q_{10} & q_{11} & q_{12} & & q_{1N} \\ q_{20} & q_{21} & q_{22} & \cdots & q_{2N} \\ \vdots & \vdots & \vdots & & \vdots \\ x_{N0} & x_{N1} & q_{N2} & & q_{NN} \end{bmatrix} \quad A = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} \quad B = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}$$

The Equations

- Remember these equations?

$$a_0 x_1^0 + a_1 x_1^1 + a_2 x_1^2 + \cdots + a_N x_1^N = y_1$$

$$a_0 x_2^0 + a_1 x_2^1 + a_2 x_2^2 + \cdots + a_N x_2^N = y_2$$

$$a_0 x_3^0 + a_1 x_3^1 + a_2 x_3^2 + \cdots + a_N x_3^N = y_3$$

$$a_0 x_4^0 + a_1 x_4^1 + a_2 x_4^2 + \cdots + a_N x_4^N = y_4$$

$$\vdots$$

$$a_0 x_M^0 + a_1 x_M^1 + a_2 x_M^2 + \cdots + a_N x_M^N = y_M$$

The Equations

- Or this system of equations

$$\begin{bmatrix} x_1^0 & x_1^1 & x_1^2 & & x_1^N \\ x_2^0 & x_2^1 & x_2^2 & & x_2^N \\ x_3^0 & x_3^1 & x_3^2 & \cdots & x_3^N \\ \vdots & \vdots & \vdots & & \vdots \\ x_M^0 & x_M^1 & x_M^2 & & x_M^N \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_M \end{bmatrix}$$

$$X = \begin{bmatrix} x_1^0 & x_1^1 & x_1^2 & & x_1^N \\ x_2^0 & x_2^1 & x_2^2 & & x_2^N \\ x_3^0 & x_3^1 & x_3^2 & \cdots & x_3^N \\ \vdots & \vdots & \vdots & & \vdots \\ x_M^0 & x_M^1 & x_M^2 & & x_M^N \end{bmatrix} \qquad A = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} \qquad Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_M \end{bmatrix} \qquad X\,A = Y$$

The Equations

- The original (over-determined) equations: $X\,A = Y$
- The Least Squares equations: $Q\,A = B$
- And their relation:

$$Q = X^T X$$

$$B = X^T Y$$

Solving the Equations

- Matrix Q is in general invertible (if the points are distinct and have enough of them)
- It is also symmetric and positive definite
- This means it is relatively easy to invert (needs fewer operations and has little round-off error)
  - See mypolyfit.m
- But not always
  - Polynomials involve large powers
  - This means a mix of huge and tiny numbers
- We often use QR decomposition to invert.
  - See mypolyfit1.m

Interpolation

- A common use for polynomial fitting
- Given the value of a function on distinct points find the value of the function in between
- There are many ways to do it
  - Nearest
  - Linear
  - Cubic (spline, Hermite)
- See interpex.m