Approximation

Linear Least-Squares

Sachin Shanbhag

Department of Scientific Computing
Florida State University,
Tallahassee, FL 32306.

References

- Burden and Faires, Numerical Analysis, 1993
- Pal, Numerical Analysis for Scientists and Engineers, 2007.

Contents

Least Squares

Recall, for polynomial interpolation, given

$$x_0, x_1, \cdots, x_i, \cdots x_n$$

and the "values" at those points

$$f_0, f_1, \cdots, f_i, \cdots f_n$$

we sought an order n polynomial, which passed exactly though all the $\{x_i, f_i\}$

$$f_i = \sum_{j=0}^{n} a_j x_i^j$$

Least Squares

- Written in full, we wanted a polynomial $p_n(x)$ to pass through the $n+1$ points by solving the linear system

$$\begin{aligned} p_n(x_0) = & \quad a_0 + a_1 x_0 + \cdots + a_n x_0^n = & f_0 \\ p_n(x_1) = & \quad a_0 + a_1 x_1 + \cdots + a_n x_1^n = & f_1 \\ & \qquad\qquad \vdots & \\ p_n(x_n) = & \quad a_0 + a_1 x_n + \cdots + a_n x_n^n = & f_n \end{aligned}$$

- In all, $n+1$ equations, $n+1$ unknowns (the a_i)

$$\mathbf{Xa} = \mathbf{f}$$

where $\mathbf{X}$ was the (square) Vandermonde matrix.

Least Squares: Motivation

- As n increases, the interpolating function becomes more complex
  - often picks up undesirable features
  - oscillations (Runge's phenomenon)
  - begins fitting noise, instead of the signal
- In least squares approximation
  - we do not require the approximating function to pass through points
  - we require it to lie as close as possible to the data "in some sense"
  - the approximating function is "coarser" than the data

Coarser Object

Suppose we wanted to fit a lower order polynomial p_m through $n+1$ points such that $m < n$.

Written in full, we seek a polynomial $p_m(x)$ that "passes" through the $n+1$ points by solving the linear system

$$\begin{aligned} a_0 + a_1 x_0 + \cdots + a_m x_0^n &= f_0 \\ a_0 + a_1 x_1 + \cdots + a_m x_1^n &= f_1 \\ &\vdots \\ a_0 + a_1 x_n + \cdots + a_m x_n^n &= f_n \end{aligned}$$

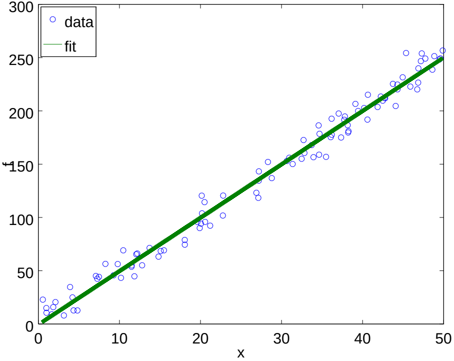
In all, $n+1$ equations, $m+1$ unknowns (the a_i)

$$\mathbf{A}\mathbf{a} = \mathbf{f}.$$

Q: If $n+1 = 100$ and $m+1 = 2$, what is the size of $\mathbf{A}$?

Least Squared Error

- In this case, we have too many equations, and too few unknowns.



- We can't "pass" or interpolate a straight-line through the 100 points!
- We can, however, try to "minimize" the distance between the straight line and the scatter of points.

Least Squared Error

For any line (defined by $\mathbf{a}$), we define the squared-error as:

$$\epsilon^2 = ||\mathbf{A}\mathbf{a} - \mathbf{f}||_2^2$$

Written out more explicitly,

$$\begin{aligned} \epsilon^2 &= (\mathbf{A}\mathbf{a} - \mathbf{f})^T(\mathbf{A}\mathbf{a} - \mathbf{f}) \\ &= \mathbf{a}^T\mathbf{A}^T\mathbf{A}\mathbf{a} - 2\mathbf{f}^T\mathbf{A}\mathbf{a} + \mathbf{f}^T\mathbf{f} \end{aligned}$$

We want to minimize the squared error, so we set:

$$\frac{\partial \epsilon^2}{\partial \mathbf{a}} = 0.$$

This yields:

$$2\mathbf{A}^T\mathbf{A}\mathbf{a} - 2\mathbf{A}^T\mathbf{f} = \mathbf{0}$$

Least Squared Error

The least-squares solution $\hat{\mathbf{a}}$ can be obtained by solving the so-called "normal equations":

$$\boxed{\mathbf{A}^T\mathbf{A}\hat{\mathbf{a}} = \mathbf{A}^T\mathbf{f}}$$

Question:
If $n + 1 = 100$ and $m = 2$ as before, what is the size of:

- $\mathbf{A}^T\mathbf{A}$?
- $\mathbf{A}^T\mathbf{f}$?

Are the normal equations over-determined?

Summary: Discrete Polynomial Approximation

- Given an over-determined system,

$$\begin{bmatrix} 1 & x_0 & x_0^2 & ... & x_0^m \\ 1 & x_1 & x_1^2 & ... & x_1^m \\ ... & ... & ... & ... & ... \\ 1 & x_n & x_n^2 & ... & x_n^m \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ ... \\ a_m \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ ... \\ f_n \end{bmatrix}$$

$$\boxed{\mathbf{A}_{(n+1)\times(m+1)}\mathbf{a}_{(m+1)\times 1} = \mathbf{f}_{(n+1)\times 1}}$$

- In typical regression problems, $n \gg m$, and hence the matrix $\mathbf{A}$ is tall.
- In the usual sense, this corresponds to a case with too many equations $(n+1)$, and too few unknowns $(m+1 < n+1)$

Discrete Polynomial Approximation

- The normal equations balance this mismatch by seeking to minimize the squared-error:

$$\mathbf{A}^T\mathbf{A}\hat{\mathbf{a}} = \mathbf{A}^T\mathbf{f}$$

- Essentially, by pre-multiplying both sides of the original equation by $\mathbf{A}^T$, we get a “square” linear system, where the number of equations is $m+1$
- This can be solved using methods from linear algebra.
- Let us consider a simple example.

Example

Problem: Consider the following data generated by adding "white noise" according to the equation

$$f = 5x + 1 + 2.5N(0,1)$$

x	f
1.00	3.97
2.00	9.66
3.00	14.41
4.00	19.38
5.00	22.10
6.00	31.00
7.00	38.70
8.00	35.53
9.00	44.99
10.00	54.54

Code

```
n = 9;
x = (1:1:n+1)';
f = 5*x + 1 + 2.5 * randn(n+1,1); % adds white noise
```

We want to set the matrix $\mathbf{A}$ as

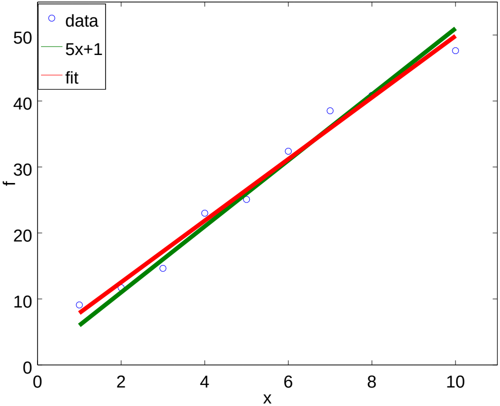
$$\mathbf{A} = \begin{bmatrix} 1 & x_0 \\ 1 & x_1 \\ \dots & \\ 1 & x_n \end{bmatrix}$$

and solve the linear system $\mathbf{A}^T\mathbf{A}\hat{\mathbf{a}} = \mathbf{A}^T\mathbf{f}$.

```
A = [ones(n+1,1) x];
ahat = (A'*A)\(A'*f)
fhat = A * ahat;
```

where we have also finally set $\hat{\mathbf{f}} = \mathbf{A}\hat{\mathbf{a}}$

Solution



Best fit $f = 5.30x - 1.74$

Example: Lorenz Function

- Given a function of the type

$$f(x) = \frac{a}{1 + x^2} + \epsilon N(0, 1), \qquad (\epsilon = 0.01)$$

  where $N(0, 1)$ is the standard normal distribution.
- Given (x_i, f_i), where $f_i = f(x_i)$, how can we get a smooth fit $f(x)$ of the noisy data?
- In other words, can we use the machinery we just learned to find the (scalar) parameter a?
- We want to minimize $||\mathbf{A}\mathbf{a} - \mathbf{f}||^2$.
- Let's ponder over the shape of the problem a bit.

Application: Lorenz Function

- The original system:

$$\mathbf{A}\mathbf{a} = \mathbf{f}$$

written out it full:

$$\begin{bmatrix} \frac{1}{1+x_1^2} \\ \frac{1}{1+x_2^2} \\ \vdots \\ \frac{1}{1+x_n^2} \end{bmatrix} [a] = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix}$$

- Again, we have too many equations, so we try to form the normal equations:

$$\mathbf{A}^T \mathbf{A} \hat{a} = \mathbf{A}^T \mathbf{f}$$

Q: What is the size of the problem?

Example

Consider the following data which we would like to fit to a Lorenz function:

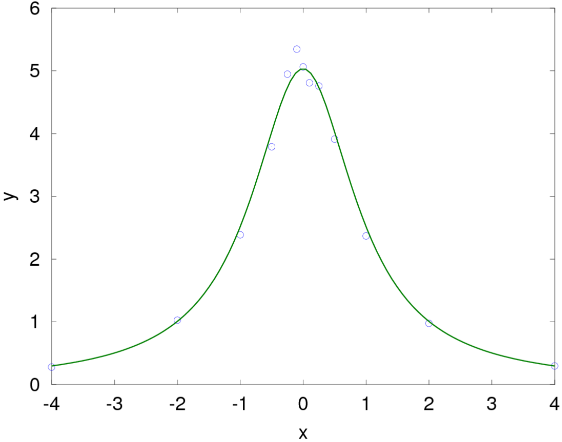
$$f(x) = \frac{a}{1+x^2}$$

x	f	x	f
-4.00	0.29191	0.10	5.15839
-2.00	1.02558	0.25	4.71492
-1.00	2.53256	0.50	3.86039
-0.50	3.79663	1.00	2.44546
-0.25	4.71709	2.00	0.99122
-0.10	4.87615	4.00	0.29183
0.00	4.71467		

Alternatively generate data using the code:

```
x1 = [-4 -2 -1 -0.5 -0.25 -0.1]';
x  = [x1;0;-flipud(x1)]
f = 5./(1+x.^2) + 0.05*5./(1+x.^2).*randn(length(x),1);
```

Solution



Best fit

$$\hat{f} = \frac{5.0339}{1 + x^2}$$