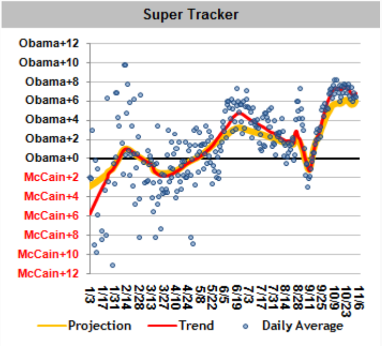
least-squares

L. Olson

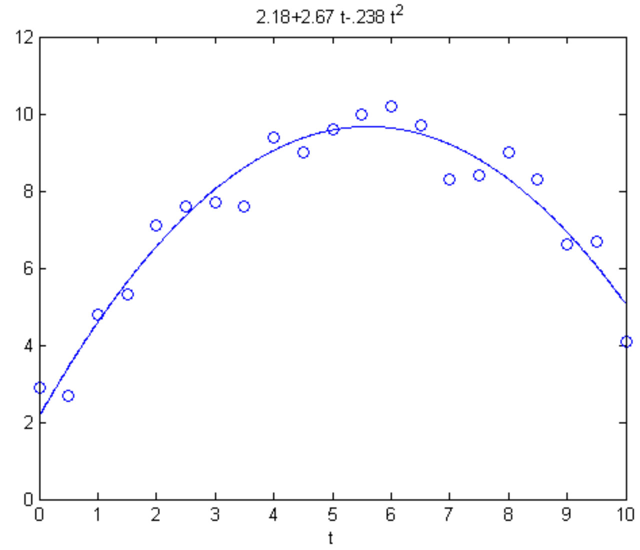
Department of Computer Science
University of Illinois at Urbana-Champaign

polling data

Suppose we are given the data $\{(x_1, y_1), ..., (x_n, y_n)\}$ and we want to find a curve that *best fits* the data.



fitting curves



fitting a line

Given n data points $\{(x_1, y_i), ..., (x_n, y_n)\}$ find a and b such that

$$y_i = ax_i + b \quad \forall i \in [1, n].$$

In matrix form, find a and b that solves

$$\begin{bmatrix} x_1 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

Systems with more equations than unknowns are called **overdetermined**

overdetermined systems

If A is an $m \times n$ matrix, then in general, an $m \times 1$ vector b may not lie in the column space of A. Hence $Ax = b$ may not have an exact solution.

Definition

The **residual** vector is

$$r = b - Ax.$$

The **least squares** solution is given by minimizing the square of the residual in the 2-norm.

normal equations

Writing $r = (b - Ax)$ and substituting, we want to find an x that minimizes the following function

$$\phi(x) = \|r\|_2^2 = r^T r = (b - Ax)^T (b - Ax) = b^T b - 2x^T A^T b + x^T A^T Ax$$

From calculus we know that the minimizer occurs where $\nabla\phi(x) = 0$.

The derivative is given by

$$\nabla\phi(x) = -2A^T b + 2A^T Ax = 0$$

Definition

The system of **normal equations** is given by

$$A^T Ax = A^T b.$$

solving normal equations

Since the normal equations forms a symmetric system, we can solve by computing the Cholesky factorization

$$A^T A = LL^T$$

and solving $Ly = A^T b$ and $L^T x = y$.

Consider

$$A = \begin{bmatrix} 1 & 1 \\ \epsilon & 0 \\ 0 & \epsilon \end{bmatrix}$$

where $0 < \epsilon < \sqrt{\epsilon_{mach}}$. The normal equations for this system is given by

$$A^T A = \begin{bmatrix} 1+\epsilon^2 & 1 \\ 1 & 1+\epsilon^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

normal equations: conditioning

The normal equations tend to worsen the condition of the matrix.

Theorem

$$cond(A^T A) = (cond(A))^2$$

```
>> A = rand(10,10);
>> cond(A)
   43.4237
>> cond(A'*A)
   1.8856e+03
```

How can we solve the least squares problem without squaring the condition of the matrix?

other approaches

- *QR* factorization.
  - For $A \in \mathbb{R}^{m \times n}$, factor $A = QR$ where
    - Q is an $m \times m$ orthogonal matrix
    - R is an $m \times n$ upper triangular matrix (since R is an $m \times n$ upper triangular matrix we can write $R = \begin{bmatrix} R' \\ 0 \end{bmatrix}$ where R is $n \times n$ upper triangular and 0 is the $(m - n) \times n$ matrix of zeros)

- SVD - singular value decomposition
  - For $A \in \mathbb{R}^{m \times n}$, factor $A = USV^T$ where
    - U is an $m \times m$ orthogonal matrix
    - V is an $n \times n$ orthogonal matrix
    - S is an $m \times n$ diagonal matrix whose elements are the singular values.

orthogonal matrices

Definition

A matrix Q is orthogonal if

$$Q^T Q = QQ^T = I$$

Orthogonal matrices preserve the Euclidean norm of any vector v,

$$||Qv||_2^2 = (Qv)^T(Qv) = v^T Q^T Qv = v^T v = ||v||_2^2.$$

using qr factorization for least squares

Now that we know orthogonal matrices preserve the euclidean norm, we can apply orthogonal matrices to the residual vector without changing the norm of the residual.

$$\|r\|_2^2 = \|b - Ax\|_2^2 = \left\| b - Q \begin{bmatrix} R \\ 0 \end{bmatrix} x \right\|_2^2 = \left\| Q^T b - Q^T Q \begin{bmatrix} R \\ 0 \end{bmatrix} x \right\|_2^2 = \left\| Q^T b - \begin{bmatrix} R \\ 0 \end{bmatrix} x \right\|_2^2$$

If $Q^T b = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$ and $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ then

$$\left\| Q^T b - \begin{bmatrix} R \\ 0 \end{bmatrix} x \right\|_2^2 = \left\| \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} - \begin{bmatrix} Rx_1 \\ 0 \end{bmatrix} \right\|_2^2 = \left\| \begin{bmatrix} c_1 - Rx_1 \\ c_2 \end{bmatrix} \right\|_2^2 = ||c_1 - Rx_1||_2^2 + ||c_2||_2^2$$

Hence the least squares solution is given by solving $\begin{bmatrix} R \\ 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$. We can solve $Rx_1 = c_1$ using back substitution and the residual is $\|r\|_2 = \|c_2\|_2$.

gram-schmidt orthogonalization

One way to obtain the QR factorization of a matrix A is by Gram-Schmidt orthogonalization.

We are looking for a set of orthogonal vectors q that span the range of A.

For the simple case of 2 vectors $\{a_1, a_2\}$, first normalize a_1 and obtain

$$q_1 = \frac{a_1}{||a_1||}.$$

Now we need q_2 such that $q_1^T q_2 = 0$ and $q_2 = a_2 + cq_1$. That is,

$$R(q_1, q_2) = R(a_1, a_2)$$

Enforcing orthogonality gives:

$$q_1^T q_2 = 0 = q_1^T a_2 + cq_1^T q_1$$

gram-schmidt orthogonalization

$$q_1^T q_2 = 0 = q_1^T a_2 + c q_1^T q_1$$

Solving for the constant c.

$$c = -\frac{q_1^T a_2}{q_1^T q_1}$$

reformulating q_2 gives.

$$q_2 = a_2 - \frac{q_1^T a_2}{q_1^T q_1} q_1$$

Adding another vector a_3 and we have for q_3,

$$q_3 = a_3 - \frac{q_2^T a_3}{q_2^T q_2} q_2 - \frac{q_1^T a_3}{q_1^T q_1} q_1$$

Repeating this idea for n columns gives us Gram-Schmidt orthogonalization.

gram-schmidt orthogonalization

Since R is upper triangular and $A = QR$ we have

$$\begin{aligned} a_1 &= q_1 r_{11} \\ a_2 &= q_1 r_{12} + q_2 r_{22} \\ \vdots &= \vdots \\ a_n &= q_1 r_{1n} + q_2 r_{2n} + ... + q_n r_{nn} \end{aligned}$$

From this we see that $r_{ij} = \frac{q_i^T a_j}{q_i^T q_i}, j > i$

orthogonal projection

The orthogonal projector onto the range of q_1 can be written:

$$\frac{q_1 q_1^T}{q_1^T q_1}$$

. Application of this operator to a vector a orthogonally projects a onto q_1. If we subtract the result from a we are left with a vector that is orthogonal to q_1.

$$q_1^T (I - \frac{q_1 q_1^T}{q_1^T q_1}) a = 0$$

orthogonal projection

figs/projection

gram-schmidt orthogonalization

```
function [Q,R] = gs_qr (A)

m = size(A,1);
n = size(A,2);

for i = 1:n
    R(i,i) = norm(A(:,i),2);
    Q(:,i) = A(:,i)./R(i,i);
    for j = i+1:n
        R(i,j) = Q(:,i)' * A(:,j);
        A(:,j) = A(:,j) - R(i,j)*Q(:,i);
    end
end

end
```

using svd for least squares

Recall that a singular value decomposition is given by

$$A = \begin{bmatrix} \vdots & \vdots & \vdots \\ u_1 & \dots & u_m \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_r & & \\ & & & \ddots & \\ & & & & 0 \end{bmatrix} \begin{bmatrix} \dots & v_1^T & \dots \\ \dots & \vdots & \dots \\ \dots & v_n^T & \dots \end{bmatrix}$$

where σ_i are the singular values.

using svd for least squares

Assume that A has rank k (and hence k nonzero singular values σ_i) and recall that we want to minimize

$$||r||_2^2 = ||b - Ax||_2^2.$$

Substituting the SVD for A we find that

$$||r||_2^2 = ||b - Ax||_2^2 = ||b - USV^T x||_2^2$$

where U and V are orthogonal and S is diagonal with k nonzero singular values.

$$||b - USV^T x||_2^2 = ||U^T b - U^T USV^T x||_2^2 = ||U^T b - SV^T x||_2^2$$

using svd for least squares

Let $c = U^T b$ and $y = V^T x$ (and hence $x = Vy$) in $\|U^T b - SV^T x\|_2^2$. We now have

$$\|r\|_2^2 = \|c - Sy\|_2^2$$

Since S has only k nonzero diagonal elements, we have

$$\|r\|_2^2 = \sum_{i=1}^{k} (c_i - \sigma_i y_i)^2 + \sum_{i=k+1}^{n} c_i^2$$

which is minimized when $y_i = \frac{c_i}{\sigma_i}$ for $1 \leqslant i \leqslant k$.

using svd for least squares

Theorem

Let A be an $m \times n$ matrix of rank r and let $A = USV^T$, the singular value decomposition. The least squares solution of the system $Ax = b$ is

$$x = \sum_{i=1}^{r} (\sigma_i^{-1} c_i) v_i$$

where $c_i = u_i^T b$.