Least Square Problems

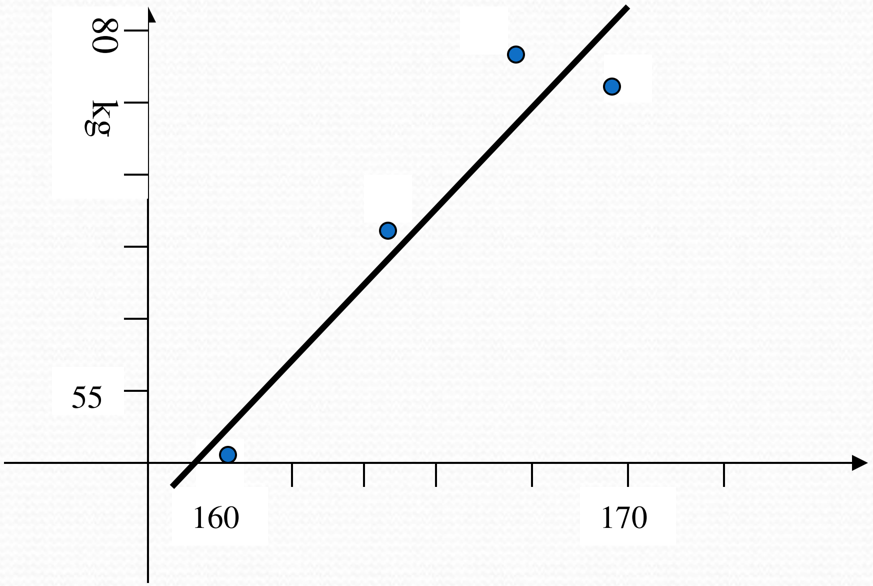
Analisis Numerik

Motivation

Name	Height (cm)	Weight (kg)
Husen	170	75
Ali	165	67
Andi	167	78
Wida	160	50

The best fit line representing the data

$$\left.\begin{aligned} 170a + b &= 75 \\ 165a + b &= 67 \\ 167a + b &= 78 \\ 160a + b &= 50 \end{aligned}\right\}$$



Over-determined system

LS – Problems

Given an over-determined system *Ax*=*b*, find *x* satisfying

$$\min_x \| Ax - b \|_2$$

Consider now

$$\begin{aligned}\phi(x) = \| Ax - b \|_2^2 &= (Ax-b)^T (Ax-b) \\ &= (x^T A^T - b^T)(Ax-b) \\ &= x^T A^T Ax - x^T A^T b - b^T Ax + b^T b \\ &= x^T A^T Ax - 2x^T A^T b + b^T b\end{aligned}$$

then

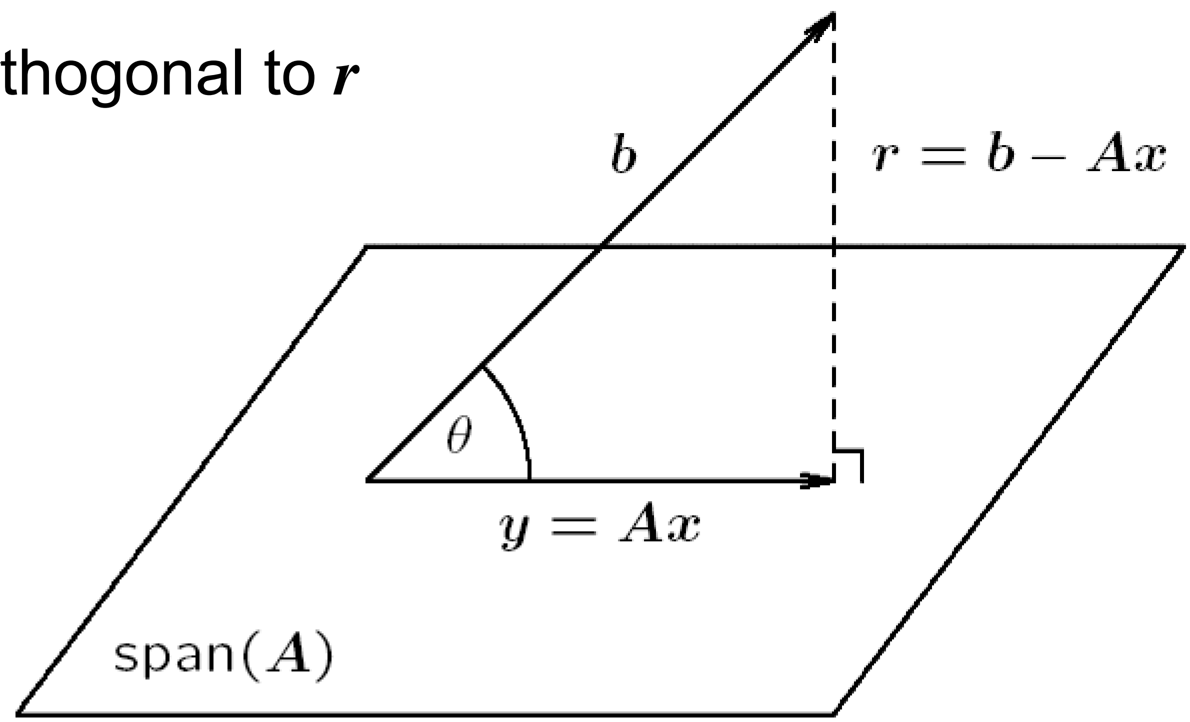
$$\frac{d\phi(x)}{dx} = 0;$$

$$A^T Ax - A^T b = 0$$

$$A^T Ax = A^T b \longleftarrow \text{Normal equation}$$

Normal Equation

$\boldsymbol{Ax}$ is orthogonal to $\boldsymbol{r}$



thus

$$0 = A^T r = A^T (Ax - b)$$

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

The matrix coefficient is symmetric;
If A is full-rank then $\boldsymbol{A^T A}$ is positive definite

Limitation of Normal Equation

- If $rank(A) < n$ then the normal equation cannot be solved (A^TA is singular)
- The condition number of A^TA can be considerably large
  - Thus the normal equation is in general sensitive

Orthogonal Transformation

- *A* matrix *Q* is orthogonal iff $\boldsymbol{Q^TQ=QQ^T=I}$
- If *Q* is orthogonal and *x* is a non-zero vector, then $||Qx||_2 = ||x||_2$
    - *Q* preserve Euclidean length
- Now consider

$$\| r \|_2 = \| Q^T r \|_2 = \| Q^T (Ax - b) \|_2 = \| Q^T Ax - Q^T b \|_2$$

- Orthogonal transformation can serve the function to introduce zero

QR Transformation

Given an *nxm* matrix *A*, there exists an orthogonal matrix *Q* such that

$$Q^T A = \begin{pmatrix} R \\ O \end{pmatrix}$$

where *R* is an upper triangular matrix and *O* is zero matrix.

In this case

$$\| r \|_2 = \| Q^T r \|_2 = \| Q^T Ax - Q^T b \|_2 = \left\| \begin{pmatrix} Rx \\ 0 \end{pmatrix} - \begin{pmatrix} \tilde{b}_1 \\ \tilde{b}_2 \end{pmatrix} \right\|_2$$

$$= \left\| Rx - \tilde{b}_1 \right\|_2 + \left\| \tilde{b}_2 \right\|_2$$

Householder Matrix

Let v be a non-zero matrix. Define the matrix

$$H = I - \frac{2}{v^T v} v v^T$$

It can be shown that H is symmetric and orthogonal

$$\begin{aligned} H^T &= (I - \frac{2}{v^T v} v v^T)^T \\ &= I - \frac{2}{v^T v} (v v^T)^T \\ &= I - \frac{2}{v^T v} v v^T = H \end{aligned}$$

$$\begin{aligned} H^2 &= (I - \frac{2}{v^T v} v v^T)(I - \frac{2}{v^T v} v v^T) \\ &= I - \frac{4}{v^T v} v v^T + \frac{4}{(v^T v)^2} (v v^T)(v v^T) \\ &= I - \frac{4}{v^T v} v v^T + \frac{4}{(v^T v)^2} v (v^T v) v^T \\ &= I - \frac{4}{v^T v} v v^T + \frac{4(v^T v)}{(v^T v)^2} v v^T \\ &= I \end{aligned}$$

Householder Transformation

Let *x* be a non-zero vector. Then

$$Hx = (I - \frac{2}{v^T v} v v^T) x$$

$$= x - \frac{2 v^T x}{v^T v} v = x - \alpha v \quad \text{where} \quad \alpha = \frac{2 v^T x}{v^T v}$$

Thus, *v* can be chosen in such a way that

$$Hx = \begin{pmatrix} c \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad \text{i.e.} \quad v = x + sign(x_1) \| x \|_2 \, e_1$$

Householder Algorithm

1. Set x= the first column of A,
2. Construct v and H_1;
3. Compute $A^1=H_1*A$
4. Next, set x= second column of A^1 starting from the second element.
5. Construct v and H_2; since H is of dimension $(m-1)$x$(m-1)$, need to adjust

$$H_2 = \begin{pmatrix} 1 & O^T \\ O & H \end{pmatrix}$$

6. Compute $A^2=H_2*A^1$
7. And so on to have

$$A^n = H_n H_{n-1} \cdots H_1 A$$

where

$$A^n = \begin{pmatrix} R \\ O \end{pmatrix}$$

Need to compute also

$$b^n = H_n H_{n-1} \cdots H_1 b$$

An example

Let

$$A = \begin{pmatrix} 170 & 1 \\ 165 & 1 \\ 167 & 1 \\ 160 & 1 \end{pmatrix}$$

so

$$\begin{cases} x = (170,165,167,160)^T \\ v=x+\|x\|e1=(501.08,165,167,160)T. \end{cases}$$

Thus

$$H_1 = \begin{pmatrix} -0.5135 & -0.4984 & -0.5044 & -0.4833 \\ -0.4984 & 0.8359 & -0.1661 & -0.1591 \\ -0.5044 & -0.1661 & 0.8319 & -0.1611 \\ -0.4833 & -0.1591 & -0.1611 & 0.8457 \end{pmatrix}$$

then

$$H_1 A = \begin{pmatrix} -331.08 & -1.9995 \\ 0 & 0.0123 \\ 0 & 0.0003 \\ 0 & 0.0422 \end{pmatrix}$$

$x=(0.0123, 0.0003, 0.0422)^T$.

An example (cont')

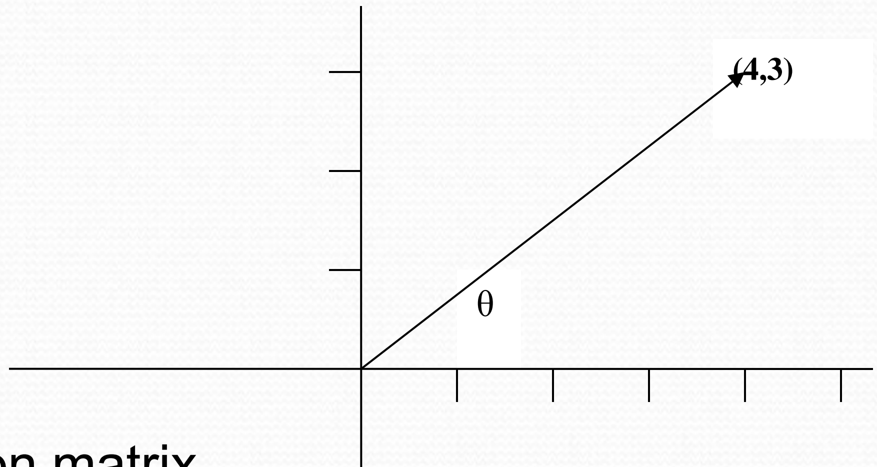
$$\tilde{H}_2 = \begin{pmatrix} -0.2795 & -0.0073 & -0.9601 \\ -0.0073 & 1.0000 & -0.0055 \\ -0.9601 & -0.0055 & 0.2296 \end{pmatrix} \rightarrow H_2 = \begin{pmatrix} 1 & 0.0000 & 0.0000 & 0.0000 \\ 0 & -0.2795 & -0.0073 & -0.9601 \\ 0 & -0.0073 & 1.0000 & -0.0055 \\ 0 & -0.9601 & -0.0055 & 0.2296 \end{pmatrix}$$

Thus

$$\begin{array}{cc} H_2 H_1 A = \left(\begin{array}{cc} -331.08 & -1.9995 \\ 0 & -0.0440 \\ \hdashline 0 & 0 \\ 0 & 0 \end{array}\right) & \begin{array}{c} \boldsymbol{R} \\ \\ \\ \boldsymbol{O} \end{array} \end{array}$$

Note:
In practice, there is no need to explicitly store *H*;
HA is instead the one we need

Givens' Rotation



The rotation matrix

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix}$$

Can be used to rotate any point to the axis (zeroing principle)

e.g.

$$\begin{pmatrix} \frac{4}{5} & \frac{3}{5} \\ -\frac{3}{5} & \frac{4}{5} \end{pmatrix} \begin{pmatrix} 4 \\ 3 \end{pmatrix} = \begin{pmatrix} 5 \\ 0 \end{pmatrix}$$

QR - Givens

A more general form of rotation matrix is given by (say in 5-D)

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & c & 0 & s & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -s & 0 & c & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{pmatrix} = \begin{pmatrix} a_1 \\ \alpha \\ a_3 \\ 0 \\ a_5 \end{pmatrix}$$

where

$$c = \frac{a_2}{\sqrt{a_2^2 + a_4^2}}; s = \frac{a_4}{\sqrt{a_2^2 + a_4^2}}; \alpha = \sqrt{a_2^2 + a_4^2}$$

Note that only two elements are affected, 2^{nd} and 4^{th}.

QR-Givens Algorithm

Given A is an mxn matrix, the following algorithm overwrites A with $Q^TA=R$, where R is an upper triangular matrix

```
for i=2,...,m
    for j=1,2,....,min(i-1,n)
        find c and s (rotation factor for [a(i,i), a(j,i)]
        A(i,:)=c*A(i,:) + s*A(j,:)
        A(j,:)=c*A(j,:) – s*A(i,:)
    end
end
```

This algorithm requires $2n^2(m\text{-}n/3)$ flops

Note: No matrix multiplication is needed; simply rows update!

Rank Deficient Problems

- The previous discussion assumes that *A* has full-rank ($rank(A)=n$)
- If $rank(A)<n$ then *R* is not invertible
- Need columns pivoting to make all the linearly independent columns sit in the same block
- Columns pivoting: use the columns with highest norm as the pivot

$$PQ^T A = \begin{pmatrix} R & C \\ O_1 & O_2 \end{pmatrix}$$

Exercise

1. A particle is believed to travel according to a parabolic curve $s(t)=at^2+bt+c$, where $s(t)$ is the position of the particle at time t. Some measurement indicates the following data:

t	0	1	2	4	7	10	15	30
S	5	8	12	34	40	65	50	2

   Find the best fit parabolic curve to the above data. Use the normal equation and *QR* factorization, compare your result.