Computational Methods
CMSC/AMSC/MAPL 460

Least squares methods

Ramani Duraiswami,
Dept. of Computer Science

Computing the factorization

- QR is useful … so how do we factorize a matrix A?
- In LU we computed a upper triangular matrix by computing adding multiples of other rows so that elements below a given column were zeroed out
- The multipliers were stored in L which gave us A=LU
- Here we want to zero out entries below the diagonal but do it with orthogonal matrices
- Two strategies
- Zero out a column at a time using a matrix Q_1 so that Q^t_1 A gives us all entries below a certain one in a column as zero
  - Householder transformations
  - Result $Q^t_n \ldots Q^t_2 Q^t_1$ A =R or A = $Q_1 \ldots Q_{n-1} Q_n$ R =Q R
- Zero out one specific entry of a column at a time
  - Givens rotations
- Product of orthogonal matrices is orthogonal

To compute QR

- Perform a sequence of orthogonal transformations that zero out elements
- Orthogonal transformations can be rotations or reflections or combinations
- Givens Rotation:
- Givens matrix has elements

$$G = \begin{bmatrix} c & s \\ s & -c \end{bmatrix}$$

- $c^2+s^2=1$
- How do we use a rotation to zero out an element?
- Let z= $[z_1\ z_2]^t$
- We want

$$Gz = \begin{bmatrix} cz_1 + sz_2 \\ sz_1 - cz_2 \end{bmatrix} = xe_1$$

- Eliminate z_2

$$(c^2 + s^2)z_1 = cx\,, \qquad c = z_1/x\,.$$

- Similarly we get $s=z_2/x$, and $z_1^2+z_2^2 = x^2$

Givens QR

- To apply idea to larger matrix, embed the Givens matrix in identity matrix. We will use the notation G_{ij} to denote an $n \times n$ identity matrix with its ith and jth rows modified to include the Givens rotation: for example, if $n = 6$, then

$$G_{25} = \begin{bmatrix} \mathbf{1} & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{c} & 0 & 0 & \mathbf{s} & 0 \\ 0 & 0 & \mathbf{1} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{1} & 0 & 0 \\ 0 & \mathbf{s} & 0 & 0 & -\mathbf{c} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{1} \end{bmatrix},$$

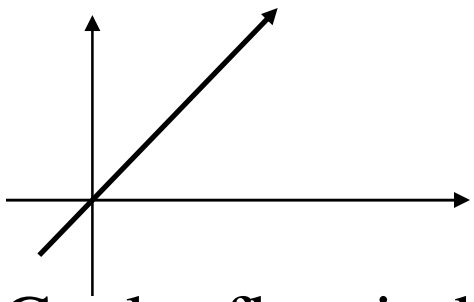
and multiplication of a vector by this matrix leaves all but rows 2 and 5 of the vector unchanged.

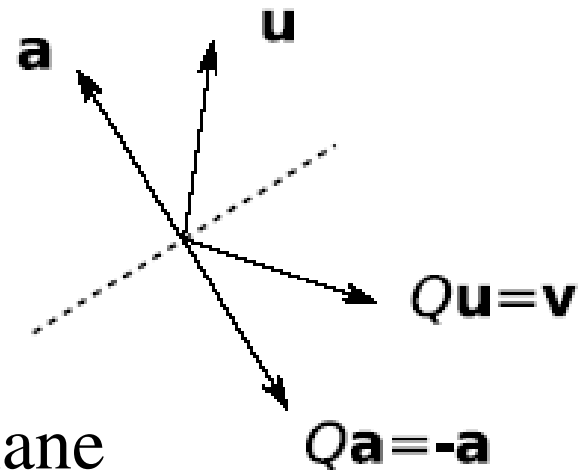
- Algorithm

```
for i=1, ...,n
    for j=i+1, ..., m
        Find Givens matrix G_ij to zero out j,i element of A
        using the the value at position (i,i)
        A=G_ij A
    end
end
```

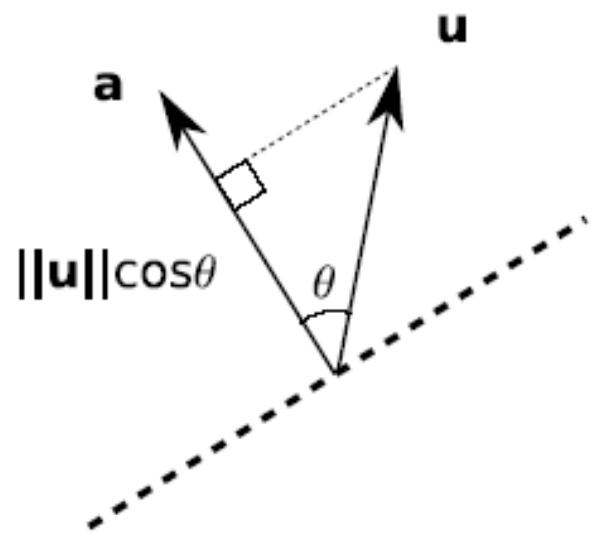
Goal: reflect axial vector through a (hyper)-plane

- Axial vector is any vector through the origin. Let it be **u**





- Goal reflect it through a plane
- Let **a** be the unit vector normal to the plane
- $\mathbf{u} \cdot \mathbf{a} = \|\mathbf{u}\| \cos \theta$
- To get reflected vector in the plane Subtract twice the component of **u** along **a**
- $\mathbf{v} = \mathbf{u} - 2\, \mathbf{a}\|\mathbf{u}\| \cos \theta$
- $\cos \theta = (\mathbf{a} \cdot \mathbf{u})/(\|\mathbf{u}\|)$



Householder transform

- Achieve this reflection via multiplication by an orthogonal matrix
- $\mathbf{v}=\mathbf{Qu}= \mathbf{u} - 2\,\mathbf{a}\|\mathbf{u}\| \cos\theta$

  $= \mathbf{u} - 2\,\mathbf{a}\|\mathbf{u}\|\,(\mathbf{a}\cdot\mathbf{u})/(\|\mathbf{u}\|)$
- $= (\mathbf{I} - 2\,\mathbf{a}\mathbf{a}^t)\mathbf{u}$
- What if **a** is not a unit vector?

$(\mathbf{I} - 2\,(\mathbf{a}\mathbf{a}^t)/(\mathbf{a}^t\,\mathbf{a}))\mathbf{u}$

- $\mathbf{Q}=\mathbf{I} - 2\,(\mathbf{a}\mathbf{a}^t)/(\mathbf{a}^t\,\mathbf{a})$

Householder Transformations

The *Householder transformation* determined by vector v is:

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

outer product, n×n matrix

inner product, scalar

To apply it to a vector x, compute:

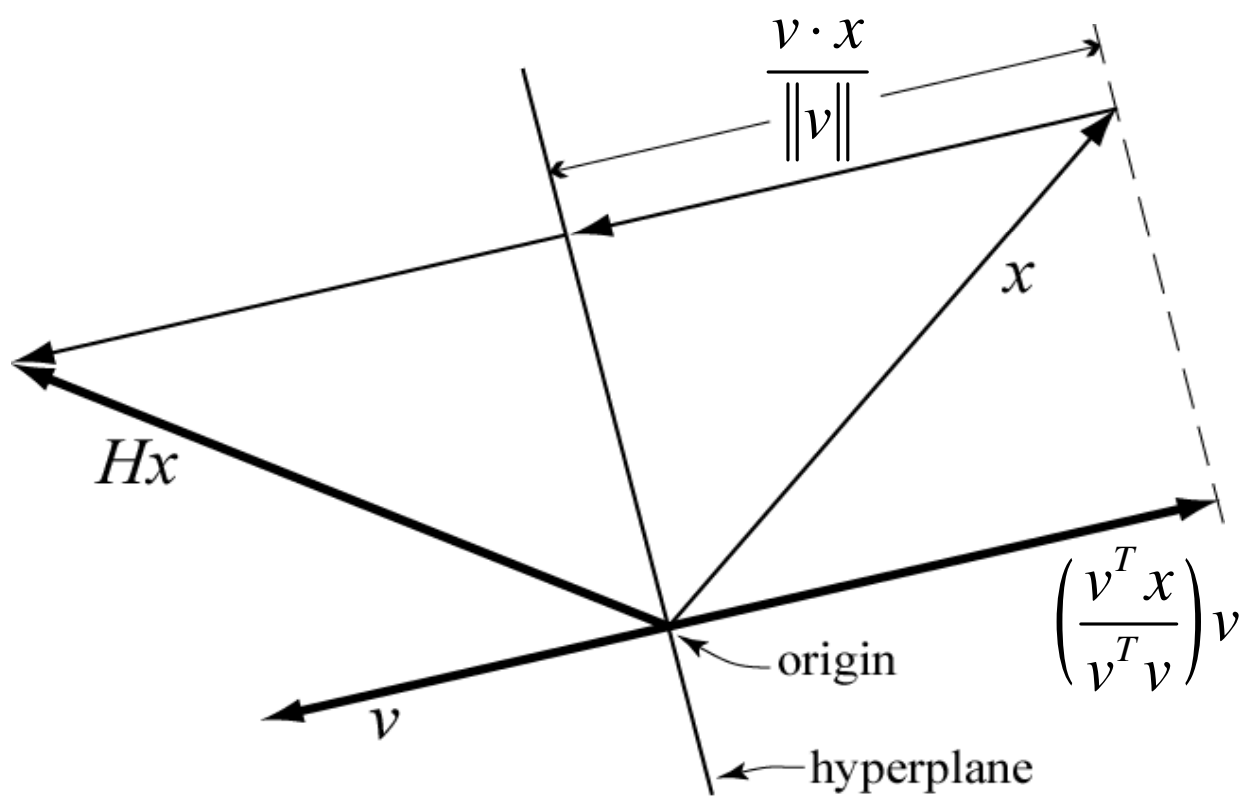
$$Hx = \left(I - 2\frac{vv^T}{v^T v} \right) x = x - 2\frac{v(v^T x)}{v^T v}$$

$$\boxed{Hx = x - \left(2\frac{v^T x}{v^T v} \right) v}$$

scalar

Householder Geometry

- ***Hx* is *x* reflected through the hyperplane perpendicular to *v*** ($p : p^{\mathrm{T}}v=0$)



Householder Properties

- H is symmetric, since

$$H^T = \left(I - 2\frac{vv^T}{v^T v} \right)^T = I^T - 2\frac{(vv^T)^T}{v^T v} = I - 2\frac{v^{T^T} v^T}{v^T v} = H$$

- H is orthogonal, since

$$H^T H = HH = \left(I - 2\frac{vv^T}{v^T v} \right)\left(I - 2\frac{vv^T}{v^T v} \right)$$

$$= I - 4\frac{vv^T}{v^T v} + 4\frac{v(v^T v)v^T}{(v^T v)^2} = I - 4\frac{vv^T}{v^T v} + 4\frac{vv^T}{v^T v} = I$$

and $H^T H = I$ implies $H^T = H^{-1}$

Householder to Zero Matrix Elements

We'll use Householder transformations to zero subdiagonal elements of a matrix.

Given any vector *a*, find the *v* that determines an *H* such that,

$$Ha = \alpha e_1 = \alpha[1, 0, 0, ..., 0]^T$$

Now solve for *v*:

$$Ha = a - \left(2\frac{v^T a}{v^T v}\right)v = a - \mu v = \alpha e_1$$

where μ is parenthesized scalar, related to length of v

$$\Rightarrow \; v = (a - \alpha e_1)/\mu$$

We're free to choose $\mu = 1$, since $\|v\|$ does not affect H

Choosing the Vector v

So $v = a - \alpha e_1$ for some scalar α.

But $\|Ha\|_2 = \|a\|_2$

(prove this by expanding $\|Ha\|_2^2 = (Ha)^T Ha$)

and $\|Ha\|_2 = |\alpha|$ by design, so $\alpha = \pm\|a\|_2$

(either sign will work).

To avoid $v \approx 0$ we choose $\alpha = -\text{sign}(a_1)\|a\|_2$,

so $v = a + \text{sign}(a_1)\|a\|_2 e_1$ is our answer.

Applying Householder Transforms

- Don't compute Hx explicitly, that costs $3n^2$ flops.
- Instead use the formula given previously,

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

  which costs $4n$ flops (if you pre-compute $v^T v$ or pre-normalize $v^T v=2$).
- Typically, when using Householder transformations, you never compute the matrix H; it's only used in derivation and analysis.

QR Decomposition

- Householder transformations are a good way to zero out subdiagonal elements of a matrix.
- A is decomposed:

$$Q^T A = \begin{bmatrix} R \\ 0 \end{bmatrix} \quad \text{or} \quad QQ^T A = A = Q \begin{bmatrix} R \\ 0 \end{bmatrix}$$

- where $Q^{\mathrm{T}}=H_n \ldots H_2 H_1$ is the orthogonal product of Householders and R is upper triangular.
- Overdetermined system $Ax=b$ is transformed into the easy-to-solve

$$\begin{bmatrix} R \\ 0 \end{bmatrix} x = Q^T b$$

Other Norms

- Here we fit using the "least-squares" or L_2 norm
- Could minimize the residual in other norms
- For example we may have more confidence in some data, and want to be sure that their residual is lower
  - Attach a weight to each residual $$\|r\|_w^2 = \sum_1^m w_i r_i^2$$
- Or we may like the 1-norm or infinity norm better

$$\|r\|_1 = \sum_1^m |r_i| \qquad \|r\|_\infty = \max_i |r_i|$$

SVD and Pseudo-Inverse

- $\mathbf{Ax}=\mathbf{b}$ $\mathbf{A}$ is $m\times n$, $\mathbf{x}$ is $n\times 1$ and $\mathbf{b}$ is $m\times 1$.
- $\mathbf{A}=\mathbf{USV}^t$ where $\mathbf{U}$ is m×m, $\mathbf{S}$ is m×n and $\mathbf{V}$ is $n\times n$
- $\mathbf{USV}^t\,\mathbf{x}=\mathbf{b}$. So $\mathbf{SV}^t\,\mathbf{x}=\mathbf{U}^t\mathbf{b}$
- If $\mathbf{A}$ has rank r, then r singular values are significant

  $\mathbf{V}^t\mathbf{x}= \mathrm{diag}(\sigma_1^{-1},\ldots,\sigma_r^{-1},0,\ldots,0)\mathbf{U}^t\mathbf{b}$

  $\mathbf{x}= \mathbf{V}\mathrm{diag}(\sigma_1^{-1},\ldots,\sigma_r^{-1},0,\ldots,0)\mathbf{U}^t\mathbf{b}$

$$\mathbf{x}_r = \sum_{i=1}^{r} \frac{\mathbf{u}_i^t\mathbf{b}}{\sigma_i}\mathbf{v}_i \qquad \sigma_r > \varepsilon, \quad \sigma_{r+1} \le \varepsilon$$

- Pseudoinverse $\mathbf{A}^+=\mathbf{V}\,\mathrm{diag}(\sigma_1^{-1},\ldots,\sigma_r^{-1},0,\ldots,0)\,\mathbf{U}^t$
  - $\mathbf{A}^+$ is a $n\times m$ matrix.
  - If rank $(\mathbf{A}) = n$ then $\mathbf{A}^+=(\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}$
  - If $\mathbf{A}$ is square $\mathbf{A}^+=\mathbf{A}^{-1}$