UNSW, School of Mathematics and Statistics

MATH2089 – Numerical Methods

Week 05 – Orthogonal Matrices, Least Squares

- MATLAB M-files
  - angex.m
  - orthex.m
  - eigex.m
  - qrex.m
  - svdex.m
  - lsqex.m
  - caellifit.m
  - caelli.dat

Angle between vectors

- The inner (dot) product of two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ is
$$\mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^{n} a_i b_i = \mathbf{a}^T \mathbf{b}.$$
- The angle $\theta \in [0, \pi]$ between two non-zero vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ satisfies
$$\cos(\theta) = \frac{\mathbf{a}^T \mathbf{b}}{\|\mathbf{a}\|_2 \, \|\mathbf{b}\|_2}.$$
- Two non-zero vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ are orthogonal $\iff$ the angle between them is $\pi/2$ $\iff$ $\mathbf{a}^T \mathbf{b} = 0$.
- The vector $\mathbf{a}$ is a unit vector $\iff$ $\|\mathbf{a}\|_2 = 1$.

Example (MATLAB M-file angex.m)

Find the angles between the vectors
$$\mathbf{a} = (3, 2, 4, 5)^T, \quad \mathbf{b} = (-2, 3, 0, 2)^T, \quad \mathbf{c} = (1, 0, -2, 1)^T.$$

Orthogonal Matrices

Definition (Orthogonal matrices)

$Q \in \mathbb{R}^{m \times n}$ is orthogonal $\iff$ $Q^T Q = I_n$

- Orthonormal columns $Q = [\mathbf{q}_1 \; \mathbf{q}_2 \; \cdots \; \mathbf{q}_n]$, $\quad \mathbf{q}_j \in \mathbb{R}^m$
$$\mathbf{q}_i^T \mathbf{q}_j = 0, \quad i \neq j, \quad \text{orthogonal}$$
$$\mathbf{q}_i^T \mathbf{q}_i = \|\mathbf{q}_i\|_2^2 = 1, \quad \text{unit length}$$
- $Q \in \mathbb{R}^{m \times n}$, $m > n \implies QQ^T \neq I$ (in general, matrix multiplication not commutative: $AB \neq BA$)
- $Q \in \mathbb{R}^{n \times n}$ square $\implies Q^T Q = I = QQ^T \implies$
  - $Q^{-1} = Q^T$
  - $\det(Q) = \pm 1$
    - Use $\det(AB) = \det(A)\det(B)$, $\quad \det(A^T) = \det(A)$
    - $\det(Q^T Q) = \det(Q^T)\det(Q) = (\det(Q))^2 = \det(I) = 1$.
  - $\|Q\|_2 = 1$, $\|Q^{-1}\|_2 = \|Q^T\|_2 = 1 \implies \kappa_2(Q) = 1$.

Orthogonal matrices - example

Example (Orthogonal matrix)

$$Q = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ 0 & \frac{2}{\sqrt{6}} \end{bmatrix}$$

Solution

- MATLAB *M-file* *orthex.m*

- Benefits
  - Ideal condition number $\kappa_2(Q) = 1$
  - Q square, to solve $Q\mathbf{x} = \mathbf{b}$ just calculate $\mathbf{x} = Q^T\mathbf{b}$
  - Q orthogonal $\Longrightarrow \|QA\|_2 = \|A\|_2$
  - Least squares problems
- Issues
  - Slightly more expensive to calculate
  - Orthogonal matrices usually not sparse

Eigenvalues and Eigenvectors – Spectral factorization

Definition (Spectral factorization)

$A \in \mathbb{R}^{n\times n}$ diagonalizable $\iff$ there exist non-singular $V \in \mathbb{C}^{n\times n}$ and diagonal $D \in \mathbb{C}^{n\times n}$:

$$A = VDV^{-1}$$

- $A = VDV^{-1} \iff AV = VD \iff V^{-1}AV = D$
- $D = \text{diag}(\lambda_1, \dots, \lambda_n)$, eigenvalues λ_j, real or complex conjugates
  - Can have repeated/multiple eigenvalues
  - Not all matrices are diagonalizable
- $V = [\mathbf{v}_1\ \mathbf{v}_2\ \cdots\ \mathbf{v}_n]$, columns eigenvectors $\mathbf{v}_j$, $A\mathbf{v}_j = \lambda_j\mathbf{v}_j$
- $\det(A) = \prod_{j=1}^{n} \lambda_j, \qquad \text{trace}(A) \equiv \sum_{j=1}^{n} a_{jj} = \sum_{j=1}^{n} \lambda_j$
- MATLAB command `eig`, MATLAB M-file `eigex.m`

```
ev = eig(A)
[V, D] = eig(A)
```


Spectral factorization – Symmetric matrices

Definition (Spectral factorization – Symmetric matrices)

$A \in \mathbb{R}^{n\times n}$ symmetric $\iff \exists$ orthogonal $Q \in \mathbb{R}^{n\times n}$, diagonal $D \in \mathbb{R}^{n\times n}$:

$$A = QDQ^T$$

- Columns of Q are orthonormal set of eigenvectors $AQ = QD$
- $D = \text{diag}(\lambda_1, \dots, \lambda_n)$, eigenvalues, real
- A symmetric
  - positive definite $\iff \lambda_i > 0$ for all $i = 1, \dots, n$
  - positive semi-definite $\iff \lambda_i \geq 0$ for all $i = 1, \dots, n$
  - negative definite $\iff \lambda_i < 0$ for all $i = 1, \dots, n$
  - negative semi-definite $\iff \lambda_i \leq 0$ for all $i = 1, \dots, n$
  - indefinite $\iff$ there exist $\lambda_i < 0, \quad \lambda_j > 0$
- A^TA symmetric, positive semi-definite $\Longrightarrow \lambda_j$ real, $\lambda_j \geq 0$
- $A \in \mathbb{R}^{m\times n}, \quad \|A\|_2 = \sqrt{\max_{j=1,\dots,n} \lambda_j(A^TA)}$

QR factorization

Definition (QR factorization)

$A \in \mathbb{R}^{m\times n}$, $m > n \implies$

$$A = Q \begin{bmatrix} R \\ 0 \end{bmatrix}$$

$Q \in \mathbb{R}^{m\times m}$ orthogonal, $R \in \mathbb{R}^{n\times n}$ upper triangular

- Partition $Q = \begin{bmatrix} Y & Z \end{bmatrix}$, $Y \in \mathbb{R}^{m\times n}$, $Z \in \mathbb{R}^{m\times(m-n)} \implies A = YR$
- Q orthogonal $\implies$

$$Q^TQ = \begin{bmatrix} Y^T \\ Z^T \end{bmatrix} \begin{bmatrix} Y & Z \end{bmatrix} = \begin{bmatrix} Y^TY & Y^TZ \\ Z^TY & Z^TZ \end{bmatrix} = \begin{bmatrix} I_n & 0 \\ 0 & I_{m-n} \end{bmatrix}$$

so $Y^TY = I_n, \quad Z^TZ = I_{m-n}, \quad Y^TZ = 0$

- MATLAB M-file `qrex.m`

Singular Value Decomposition

Definition (Singular Value Decomposition SVD)

$A \in \mathbb{R}^{m\times n}$ *then*

$$A = U \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} V^T$$

- $U \in \mathbb{R}^{m\times m}$ orthogonal ($U^TU = I_m$)
- $V \in \mathbb{R}^{n\times n}$ orthogonal ($V^TV = I_n$)
- $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_n)$
- Singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$ always ordered

- Singular values of A are square roots of eigenvalues of A^TA
- A full rank $\iff$ columns of A linearly independent
  - A full rank $\iff$ $\sigma_n > 0$ numerically preferable
- condition number $\kappa_2(A) = \sigma_1/\sigma_n$
- MATLAB function `svd`, M-file `svdex.m`

Linear Least Squares – Linear fit to data

Example (Line of best fit)

Find the line $\ell(t) = \alpha + \beta t$ of best fit to the data (t_j, y_j) for $j = 1, \ldots, m$

- Parameter vector $\mathbf{x}$, coefficient matrix A, data vector $\mathbf{y}$

$$\mathbf{x} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \quad A = \begin{bmatrix} 1 & t_1 \\ 1 & t_2 \\ \vdots & \vdots \\ 1 & t_m \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix},$$

MATLAB
```
x = A \ y
lsqex.m
```

- The residual is

$$\mathbf{r} = A\mathbf{x} - \mathbf{y} = (\alpha + \beta t_1 - y_1, \ldots, \alpha + \beta t_m - y_m)^T$$

- Least squares: Minimize $F = \|\mathbf{r}\|_2^2 = \mathbf{r}^T\mathbf{r} = \sum_{j=1}^m (\alpha + \beta t_j - y_j)^2$
- Set $\partial F/\partial\alpha = 0$ and $\partial F/\partial\beta = 0$ we obtain

$$\begin{cases} \alpha m + \beta \sum_{j=1}^m t_j & = \sum_{j=1}^m y_j \\ \alpha \sum_{j=1}^m + \beta \sum_{j=1}^m t_j^2 & = \sum_{j=1}^m y_j t_j \end{cases}$$

Normal equations

- Alternatively, write $F = \mathbf{r}^T\mathbf{r} = \mathbf{x}^T A^T A\mathbf{x} - 2\mathbf{x}^T A^T\mathbf{y} + \mathbf{y}^T\mathbf{y}$ and set $\partial F/\partial\alpha = \partial F/\partial\beta = 0$, we also obtain:
- Normal equations $(A^T A)\,\mathbf{x} = A^T\mathbf{y}$ (2 by 2 linear system)

$$A^T A = \begin{bmatrix} m & \sum_{j=1}^m t_j \\ \sum_{j=1}^m t_j & \sum_{j=1}^m t_j^2 \end{bmatrix}, \quad A^T\mathbf{y} = \begin{bmatrix} \sum_{j=1}^m y_j \\ \sum_{j=1}^m t_j y_j \end{bmatrix}$$

Example

Use MATLAB to find the line $y(t) = \alpha + \beta t$ of best fit to the data

j	1	2	3	4	5
t_j	0	0.5	1	1.5	2
y_j	1.0	0.6	0.4	0.1	0.08

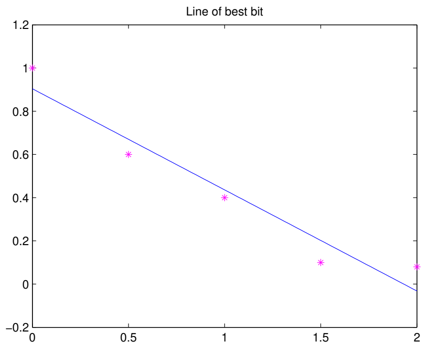
In this case, $m = 5$ and $n = 2$.

```
tdat = [0:0.5:2];
ydat = [1.0 0.6 0.4 0.1 0.08];
tdat = tdat(:); % change to column vector
ydat = ydat(:);
A = [ones(size(tdat)) tdat];
x = A \ ydat
```

After running, we get $x(1) = 0.9040$ ans $x(2) = -0.4680$. The line of best fit is $y(t) = 0.904 - 0.468t$.

Line of best fit – Example

```
t = linspace(0,2,100);
y = x(1) + x(2)*t;
plot(t,y,'b-',tdat,ydat,'m*');
```



MATLAB lsqex.m

Linear least squares

- $A \in \mathbb{R}^{m\times n}$, $m > n$, over determined (more equations than variables)
- Residual $\mathbf{r}(\mathbf{x}) = A\mathbf{x} - \mathbf{b}$, $\quad \mathbf{r}(\mathbf{x}) = \mathbf{0} \iff A\mathbf{x} = \mathbf{b}$
- Least squares Minimize $\|\mathbf{r}(\mathbf{x})\|_2^2 = \mathbf{r}(\mathbf{x})^T\mathbf{r}(\mathbf{x}) = \sum_{j=1}^m (r_j(\mathbf{x}))^2$
- Normal equations A full rank $\Longrightarrow$ least squares solution

$$\mathbf{x} = (A^TA)^{-1}A^T\mathbf{b}$$

  - In practice solve $(A^TA)\,\mathbf{x} = A^T\mathbf{b}$
  - Symmetric positive definite coefficient matrix – Cholesky factorization
  - Issue $\kappa_2(A^TA) = \kappa_2(A)^2$ squaring condition number

Using QR factorization

Numerically preferable Use QR factorization

$$A = Q\begin{bmatrix} R \\ 0 \end{bmatrix}$$

$Q \in \mathbb{R}^{m\times m}$ orthogonal, $\quad R \in \mathbb{R}^{n\times n}$ upper triangular. Partition
$Q = \begin{bmatrix} Y & Z \end{bmatrix}$, $Y \in \mathbb{R}^{m\times n}$, $Z \in \mathbb{R}^{m\times(m-n)} \Longrightarrow A = YR$

- A full rank $\iff$ R nonsingular
- $A^TA = R^TR \Longrightarrow$ the normal equation is equivalent to $R^TR\mathbf{x} = R^TY^T\mathbf{b}$
- solve $R\mathbf{x} = Y^T\mathbf{b}$ by back-substitution.
- Condition number $\kappa_2(A) = \kappa_2(R)$
- MATLAB `x = A \ b`

Example

Use MATLAB to find the quadratic $y(t) = \alpha + \beta t + \gamma t^2$ of best fit to the data in the previous example.

In this case, $m = 5$ and $n = 3$. The residual is

$$\mathbf{r} = (\alpha + \beta t_1 + \gamma t_1^2 - y_1, \ldots, \alpha + \beta t_m + \gamma t_m^2 - y_m)^T.$$

We can write $\mathbf{r} = A\mathbf{x} - \mathbf{y}$ where the coefficient matrix A is

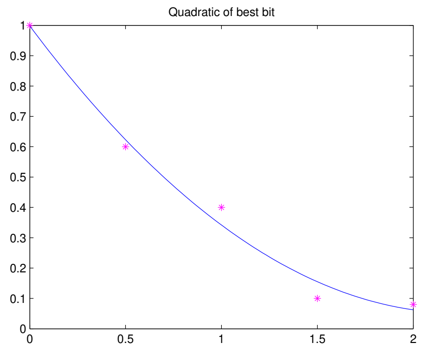
$$\begin{bmatrix} 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ \vdots & \vdots & \vdots \\ 1 & t_m & t_m^2 \end{bmatrix}$$

Define tdat and ydat as before. Then

```
A = [ones(size(tdat)) tdat  tdat.^2];
x = A \ ydat
```

We get $\mathtt{x} = [0.9983; -0.8451; 0.1886]$. Thus the quadratic of best fit is $y(t) = 0.9983 - 0.8451t + 0.1886t^2$.

```
t = linspace(0,2,100);
y = x(1) + x(2)*t + x(3)*t.^2;
plot(t,y,'b-',tdat,ydat,'m*');
```



Using exponential functions

Example

Approximate the data from the previous examples by an exponential function $y(t) = \lambda e^{\mu t}$.

Convert to a linear problem

$$y(t) = \lambda e^{\mu t} \Longrightarrow \ln y(t) = \ln(\lambda e^{\mu t}) = \ln \lambda + \mu t.$$

The data values yield a system of equations in λ and μ

$$\begin{cases} \ln \lambda + \mu t_1 & = \ln y_1 \\ \ln \lambda + \mu t_2 & = \ln y_2 \\ \vdots & \vdots \\ \ln \lambda + \mu t_m & = \ln y_m \end{cases}$$

The coefficient matrix

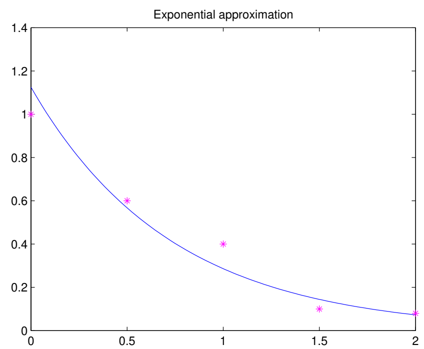
$$A = \begin{bmatrix} 1 & t_1 \\ 1 & t_2 \\ \vdots & \vdots \\ 1 & t_m \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} \ln \lambda \\ \mu \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} \ln y_1 \\ \ln y_2 \\ \vdots \\ \ln y_m \end{bmatrix}$$

```
A = [ones(size(tdat)) tdat];
x = A \ log(ydat)
lam = exp(x(1))
mu = x(2)
```

`lam = 1.1247` and `mu = -1.3686`. The exponential approximation is $y(t) = 1.1247e^{-1.3686t}$.

Plotting

```
t = linspace(0,2,100);
y = lam * exp(mu*t);
plot(t,y,b-,tdat,ydat,m*);
```



Example: Caelli data

Example (Caelli data – linear least squares)

Approximate the Caelli data by (MATLAB M-file `caellifit.m`)

$$\phi(\mathbf{x};t) = x_1 e^{-\alpha t} + x_2 e^{-v_1(t-\mu_1)^2} + x_3 e^{-v_2(t-\mu_2)^2} + x_4 e^{-v_3(t-\mu_3)^2}$$