Overview

Motivation: data points might be affected by errors. The interpolation criterion $y_i = f(x_i) = g(x_i)$, i.e., exactly recover the function values is not appropriate.

- Minimal norm approximation criteria
- Solving linear least squares problems
- Nonlinear least squares problems

- Problem: approximate $f: [a, b] \to \mathbb{R}$ by $g: [a, b] \to \mathbb{R}$, $f(t) \cong g(t; c)$, c are approximant parameters
- Best approximants can be defined in terms of error minimization

$$\varepsilon = \| f - g \|, \min_c \varepsilon(c)$$

- Examples. Assume f known by sample $\mathcal{D} = \{(x_i, y_i = f(x_i)), i = 1, \ldots, m\}$
  - Interpolation: $\| f \|_i = \sum_{i=1}^m | f(x_i)|$. Verify norm properties:
    - $\to$ $a \in \mathbb{R}$, $\|a f \|_i = \sum_{i=1}^m |a f(x_i)| = |a| \sum_{i=1}^m | f(x_i)| = |a| \, \| f \|_i$ ✓
    - $\to$ $\| f \|_i = 0 = \sum_{i=1}^m | f(x_i)| \Rightarrow f(x_i) = 0 \Rightarrow f = 0$ on grid $\{x_1, \ldots, x_m\}$
    - $\to$ $\| f + g \| = \sum_{i=1}^m | f(x_i) + g(x_i)| \leqslant \sum_{i=1}^m (| f(x_i)| + |g(x_i)|) = \| f \|_i + \| g \|_i$
  - Least-sqaures: use 2-norm $\| f \|_2 = \sum_{i=1}^m (f(x_i))^2$
  - Min-max: use inf norm $\| f \|_\infty = \max_{a \leqslant t \leqslant b} | f(t)|$

$$\min_c \varepsilon(c) \Rightarrow \min_c \max_{a \leqslant t \leqslant b} | f(t) - g(t; c)|.$$

- Based on two ideas:
  1. Approximant is a linear combination, $g = c_1 g_2 + \cdots + c_n g_n$
  2. Error is measured by 2-norm

$$\min_{\boldsymbol{c} \in \mathbb{R}^n} \| f - g \| \Rightarrow \min_{c \in \mathbb{R}^n} \sum_{i=1}^{m} (y_i - g(x_i; c))^2.$$

- Choose a basis set:
  - Monomial $\mathcal{M}_n(t) = \{1, t, t^2, \ldots, t^{n-1}\}$
  - Trigonometric $\mathcal{T} = \{1, \cos t, \sin t, \ldots\}$, $f \colon [0, 2\pi] \rightarrow \mathbb{R}$, $f(t + 2\pi) = f(t)$
  - Exponential $\mathcal{E}_n(t) = \{1, e^t, e^{2t}, \ldots, e^{(n-1)t}\}$
- Evaluate basis set at grid points $\boldsymbol{x} \in \mathbb{R}^m$, obtain $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ matrix

$$\mathcal{M}_n(\boldsymbol{x}) = \boldsymbol{A} = \left[\begin{array}{ccccc} \mathbf{1} & \boldsymbol{x} & \boldsymbol{x}^2 & \ldots & \boldsymbol{x}^{n-1} \end{array} \right], \boldsymbol{x}^k = \left[\begin{array}{ccc} x_1^k & \ldots & x_m^k \end{array} \right]^T$$

- Seek $\boldsymbol{c} \in \mathbb{R}^n$ $\min_{\boldsymbol{c} \in \mathbb{R}^n} \| \boldsymbol{y} - \boldsymbol{A}\boldsymbol{c} \|_2$

- Projection of $\boldsymbol{v} \in \mathbb{R}^m$ along direction $\boldsymbol{u} \in \mathbb{R}^m$, $\|\boldsymbol{u}\| = 1$

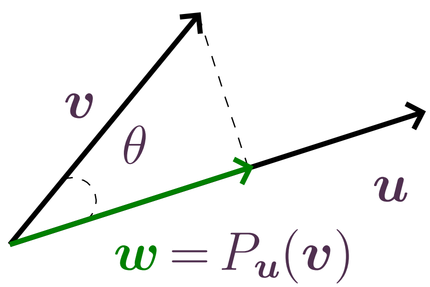


Figure 1. Projection operation

- $\boldsymbol{w} = (\|\boldsymbol{v}\| \cos\theta)\, \boldsymbol{u} = \left(\|\boldsymbol{v}\| \frac{\boldsymbol{u} \cdot \boldsymbol{v}}{\|\boldsymbol{u}\| \, \|\boldsymbol{v}\|} \right) \boldsymbol{u} = (\boldsymbol{u}^T \boldsymbol{v}) \boldsymbol{u} = \boldsymbol{u}(\boldsymbol{u}^T \boldsymbol{v}) = (\boldsymbol{u}\, \boldsymbol{u}^T) \boldsymbol{v} \Rightarrow$
- Projection matrix $\boldsymbol{P}_{\boldsymbol{u}} = \boldsymbol{u}\, \boldsymbol{u}^T$

- Projection along $\boldsymbol{e}_1$ direction in $\mathbb{R}^2$

$$\boldsymbol{P}_{\boldsymbol{e}_1} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} [1 \quad 0] = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \boldsymbol{P}_{\boldsymbol{e}_1} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ 0 \end{bmatrix}.$$

- Projection along direction of vector $\boldsymbol{w} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ in $\mathbb{R}^3$:
  - First obtain a vector of unit norm $\boldsymbol{u} = \boldsymbol{w} / \|\boldsymbol{w}\| = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$
  - Projection matrix

$$\boldsymbol{P}_{\boldsymbol{u}} = \boldsymbol{u}\boldsymbol{u}^T = \frac{1}{3} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} [\, 1 \quad 1 \quad 1 \,] = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

Definition. *The Dirac delta symbol δ_{ij} is defined as*

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

Definition. *A set of vectors $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n\}$ is said to be orthonormal if*

$$\boldsymbol{u}_i^T \boldsymbol{u}_j = \delta_{ij}$$

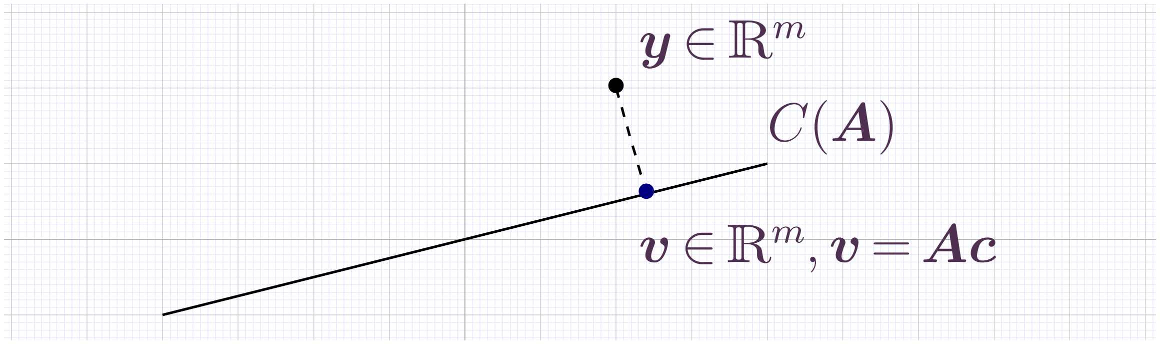
- The column vectors of the identity matrix are orthonormal

$$\boldsymbol{I} = \left(\begin{array}{ccc} \boldsymbol{e}_1 & \ldots & \boldsymbol{e}_m \end{array} \right), \boldsymbol{e}_i^T \boldsymbol{e}_j = \delta_{ij}, \boldsymbol{I}^T \boldsymbol{I} = \boldsymbol{I}$$

- Columns of $\boldsymbol{Q} = [\begin{array}{ccc} \boldsymbol{q}_1 & \ldots & \boldsymbol{q}_n \end{array}] \in \mathbb{R}^{m \times n}$ are orthonormal if

$$\boldsymbol{Q}^T \boldsymbol{Q} = \left[\begin{array}{c} \boldsymbol{q}_1^T \\ \vdots \\ \boldsymbol{q}_n^T \end{array} \right] [\begin{array}{ccc} \boldsymbol{q}_1 & \ldots & \boldsymbol{q}_n \end{array}] = \left[\begin{array}{cccc} \boldsymbol{q}_1^T \boldsymbol{q}_1 & \boldsymbol{q}_1^T \boldsymbol{q}_2 & \ldots & \boldsymbol{q}_1^T \boldsymbol{q}_n \\ \boldsymbol{q}_2^T \boldsymbol{q}_1 & \boldsymbol{q}_2^T \boldsymbol{q}_2 & \ldots & \boldsymbol{q}_2^T \boldsymbol{q}_n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{q}_n^T \boldsymbol{q}_1 & \boldsymbol{q}_n^T \boldsymbol{q}_2 & \ldots & \boldsymbol{q}_n^T \boldsymbol{q}_n \end{array} \right] = \boldsymbol{I}_n$$

- Linear least squares problems (LLSQ) easily solved by Pythagorean theorem



- Solution is orthgonal projection onto $C(\boldsymbol{A})$

$$\boldsymbol{Q}\boldsymbol{R} = \boldsymbol{A}, \boldsymbol{P}_{C(\boldsymbol{A})} = \boldsymbol{Q}\boldsymbol{Q}^T, \boldsymbol{v} = (\boldsymbol{Q}\boldsymbol{Q}^T)\boldsymbol{y}$$

- The LLSQ parameter vector $\boldsymbol{c}$ is found by back-substitution from

$$\boldsymbol{v} = (\boldsymbol{Q}\boldsymbol{Q}^T)\boldsymbol{y} = (\boldsymbol{Q}\boldsymbol{R})\boldsymbol{c} \Rightarrow \boldsymbol{R}\boldsymbol{c} = \boldsymbol{Q}^T\boldsymbol{y}.$$

- Consider data $\mathcal{D} = \{(x_i, y_i), i = 1, \dots, m\}$
- The *polynomial interpolant* of degree $m-1$ passes through the data points

$$p_{m-1}(x) = c_0 + c_1 x + \cdots + c_{m-1} x^{m-1} = \begin{bmatrix} 1 & x & \dots & x^{m-1} \end{bmatrix} \boldsymbol{c}$$

- Impose conditions $p_{m-1}(x_i) = y_i, i = 1, \dots, m$ to obtain linear system

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{m-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{m-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \dots & x_m^{m-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{m-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{bmatrix} \Rightarrow \boldsymbol{A}\boldsymbol{c} = \boldsymbol{y}, \boldsymbol{A} \in \mathbb{R}^{m \times m}$$

```
>> m=4; x=transpose(1:m); y=1 - 2*x + 3*x.^2 - 4*x.^3;
```

```
>> A=[x.^0 x.^1 x.^2 x.^3]; [Q,R]=qr(A); c = R \ (Q'*y);␣c'
```

- Consider data $\mathcal{D} = \{(x_i, y_i), i = 1, \ldots, m\}$
- The *polynomial interpolant* of degree $m-1$ passes through the data points

$$p_{m-1}(x) = c_0 + c_1 x + \cdots + c_{m-1} x^{m-1} = \begin{bmatrix} 1 & x & \ldots & x^{m-1} \end{bmatrix} \boldsymbol{c}$$

- Impose conditions $p_{m-1}(x_i) = y_i, i = 1, \ldots, m$ to obtain linear system

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^{m-1} \\ 1 & x_2 & x_2^2 & \ldots & x_2^{m-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \ldots & x_m^{m-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{m-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{bmatrix} \Rightarrow \boldsymbol{A}\boldsymbol{c} = \boldsymbol{y}, \boldsymbol{A} \in \mathbb{R}^{m \times m}$$

```
>> m=4; x=transpose(1:m); y=1 - 2*x + 3*x.^2 - 4*x.^3;
```

```
>> A=[x.^0 x.^1 x.^2 x.^3]; [Q,R]=qr(A); c = R \ (Q'*y); c'
```

$$\begin{pmatrix} 1.0000 & -2.0000 & 3.0000 & -4.0000 \end{pmatrix}$$

- Consider noisy data containing many measurements $\mathcal{D} = \{(x_i, y_i), i = 1, \ldots, m\}$
- Assume data without noise would conform to some polynomial law

$$y_i = z_i + r_i = c_0 + c_1 x_i + r_i, i = 1, \ldots, m$$

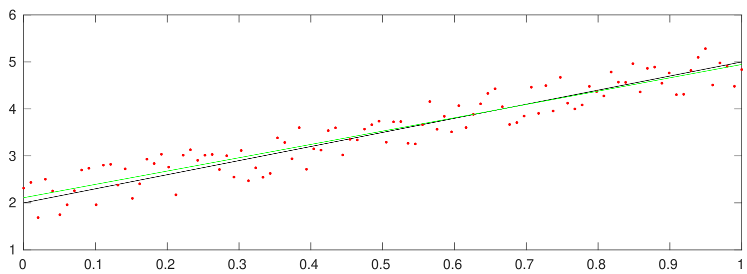
with r_i a random number uniformly distributed within $(-R, R)$.

```
>> m=100; x=linspace(0,1,m)';
```

```
>> c0=2; c1=3; z=c0+c1*x;  y=(z+rand(m,1)-0.5);
```

```
>> A=ones(m,2); A(:,2)=x(:); [Q,R]=qr(A); c = R \ (Q'*y);␣c'
```

```
>> w=A*c; plot(x,z,'k',x,y,'.r',x,w,'g');
```



- Consider noisy data containing many measurements $\mathcal{D} = \{(x_i, y_i), i = 1, \ldots, m\}$
- Assume data without noise would conform to some polynomial law

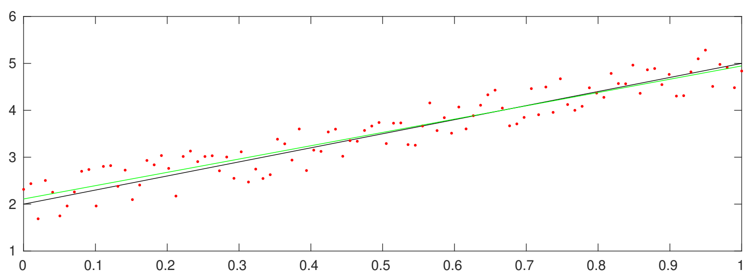
$$y_i = z_i + r_i = c_0 + c_1 x_i + r_i, i = 1, \ldots, m$$

with r_i a random number uniformly distributed within $(-R, R)$.

```
>> m=100; x=linspace(0,1,m)';
>> c0=2; c1=3; z=c0+c1*x;  y=(z+rand(m,1)-0.5);
>> A=ones(m,2); A(:,2)=x(:); [Q,R]=qr(A); c = R \ (Q'*y);␣c'
```

(1.9506 3.0310)

```
>> w=A*c; plot(x,z,'k',x,y,'.r',x,w,'g');
```

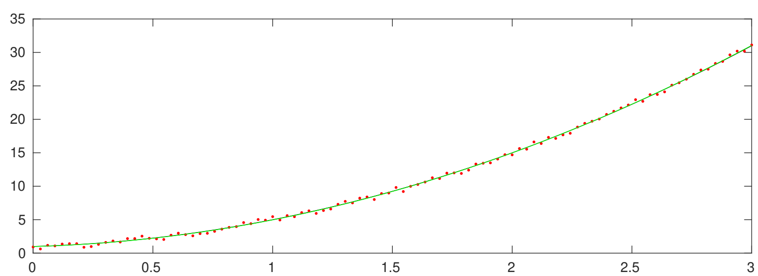


- Consider noisy data containing many measurements $\mathcal{D} = \{(x_i, y_i), i = 1, \ldots, m\}$
- Assume data without noise would conform to some polynomial law

$$y_i = z_i + r_i = c_0 + c_1 x_i + r_i, i = 1, \ldots, m$$

with r_i a random number uniformly distributed within $(-R, R)$.

```
>> m=100; x=linspace(0,3,m)';
>> c0=1; c1=1; c2=3; z=c0+c1*x+c2*x.^2;  y=(z+rand(m,1)-0.5);
>> A=ones(m,3); A(:,2)=x(:); A(:,3)=x(:).^2; [Q,R]=qr(A); c=R\(Q'*y); c'
>> w=A*c; plot(x,z,'k',x,y,'.r',x,w,'g');
```



- Consider noisy data containing many measurements $\mathcal{D} = \{(x_i, y_i), i = 1, \ldots, m\}$
- Assume data without noise would conform to some polynomial law

$$y_i = z_i + r_i = c_0 + c_1 x_i + r_i, i = 1, \ldots, m$$

with r_i a random number uniformly distributed within $(-R, R)$.

```
>> m=100; x=linspace(0,3,m)';
```

```
>> c0=1; c1=1; c2=3; z=c0+c1*x+c2*x.^2;  y=(z+rand(m,1)-0.5);
```

```
>> A=ones(m,3); A(:,2)=x(:); A(:,3)=x(:).^2; [Q,R]=qr(A); c=R\(Q'*y);␣c'
```

(1.0047 0.9641 3.0143)

```
>> w=A*c; plot(x,z,'k',x,y,'.r',x,w,'g');
```