CS368 MATLAB Programming

Lecture 7

Young Wu

Based on lecture slides by Michael O'Neill and Beck Hasti

March 9, 2022

Coordination Game

Quiz

~~Q1~~

~~Q2~~

will repeat next next week,

- No lecture next week.
- For the quiz grade next week (out of 2):
- A : If $\geqslant$ 95% of you chooses A, everyone gets 3 points.
- B : If $\geqslant$ 75% of you chooses B, everyone gets 2 points.
- C : If $\geqslant$ 50% of you chooses C, everyone gets 1 point.
- Otherwise, everyone gets 0.

Schedule

Admin

- No lecture next week.
- There will be office hours (all on Zoom). P1 – P3
- Code similarity check. Grade maybe changed to 0 if: on Monday

1. No code submission on Canvas (either .txt or .*m*). Allowed to resubmit.
2. Almost identical code without attribution. Not allowed to resubmit.

- Midterm grades on Canvas. Not going to submit official midterm grades.

Polynomials

Math

- A degree n polynomial, $c_n x^n + c_{n-1} x^{n-1} + c_{n-2} x^{n-2} + \ldots + c_2 x^2 + c_1 x + c_0$ is specified by a list of coefficients $c_n, c_{n-1}, \ldots, c_1, c_0$ usually stored in a vector.
- The computation can be done efficiently using Horner's method: $((((c_n x + c_{n-1}) x + c_{n-2}) x + \ldots + c_2) x + c_1) x + c_0$.

Exponential and Trigonometry Functions

Math

- Many elementary function can be evaluated and approximated using polynomials.

1. $e^x \approx 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{x^5}{120} + \ldots$
2. $\sin(x) \approx x - \frac{x^3}{6} + \frac{x^5}{120} + \ldots$
3. $\cos(x) \approx 1 - \frac{x^2}{2} + \frac{x^4}{24} + \ldots$

- They are also called Taylor polynomials.

Polynomial Evaluation

Code

- Suppose c and x are row vectors with length n.
- *dot(c, x .^ ((n − 1):−1:0))* evaluates the polynomial with coefficients c at x.
- *polyval(c, x)* evaluates the same polynomial but faster.

Polynomial Evaluation, Trig

Quiz

Q3

6 term

$\leq 0 \ldots 1$

8

- (Compute $\cos(x) \approx 1 - \frac{x^2}{2} + \frac{x^4}{24} + \ldots - \frac{x^{10}}{10!}$ at $x = 1$.)
- **0.5403**
- C : *polyval(((−1) .^ (5:−1:0)) ./ (factorial (10:−2:0)), 1)*

$10 \cdot 9 \cdot 8 \cdot \ldots \cdot 2 \cdot 1$

$(-1)^{5, 4, 3, 2, 1, 0}$

10 8 6 4 2 0

−1 +1 −1

Lagrange Polynomial

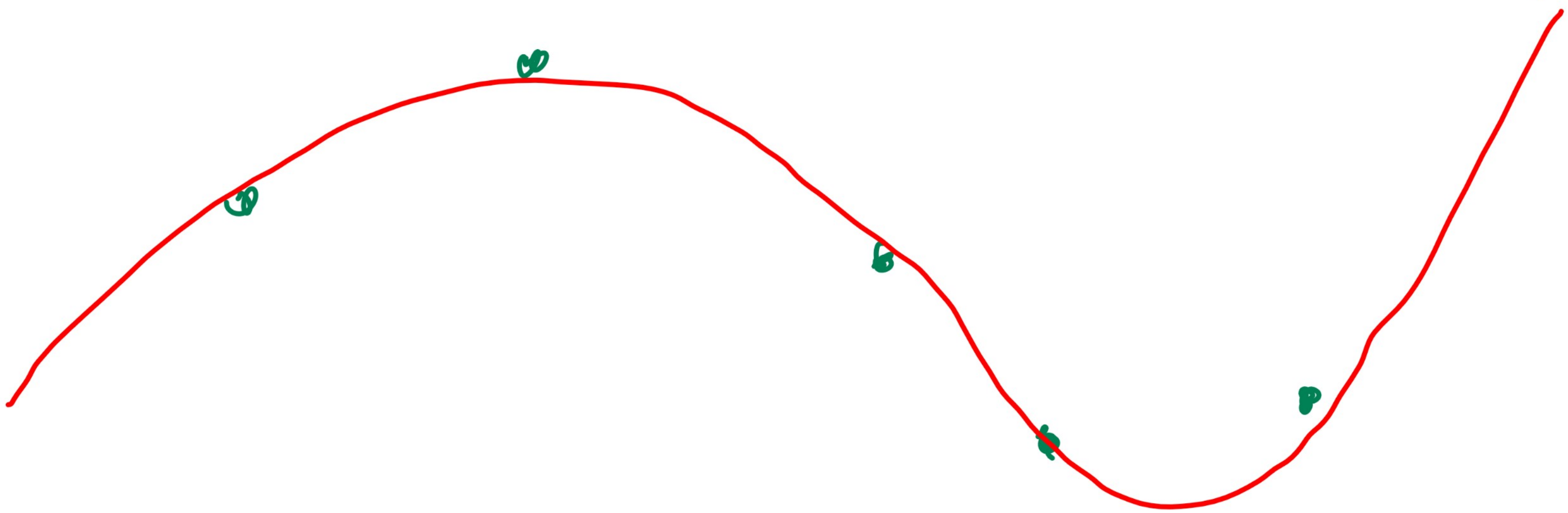
Math

- A set of n points (with distinct x values) uniquely determine a degree $n-1$ polynomial that passes through all the points.
- The coefficients of the leading terms can be zero, so technically the degree of the polynomial can be less than $n-1$.

Lagrange Polynomial Diagram

Math

$$C_4 x^4 + C_3 x^3 + \dots + C_0$$



Vandermonde Matrix

Math

(x_1, y_1)

- The coefficients can be found by solving a system of linear equations,

$$\begin{bmatrix} x_1^{n-1} & x_1^{n-2} & \dots & 1 \\ x_2^{n-1} & x_2^{n-2} & \dots & 1 \\ \dots & \dots & \dots & \dots \\ x_n^{n-1} & x_n^{n-2} & \dots & 1 \end{bmatrix} \begin{bmatrix} c_{n-1} \\ c_{n-2} \\ \dots \\ c_0 \end{bmatrix} = \begin{bmatrix} \boxed{y_1} \\ y_2 \\ \dots \\ y_n \end{bmatrix}.$$

(x_2, y_2)

$c_{n-1} x_1^{n-1} + c_{n-2} x_1^{n-2} + \dots + c_0 = y_1$

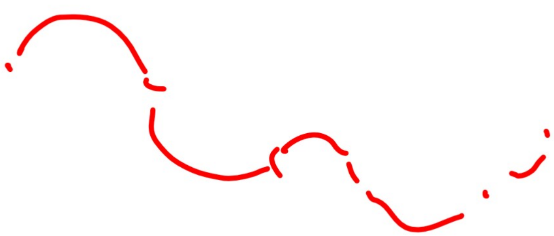
Polyfit

Code

- Suppose x, y are two row vectors with length n.
- `repmat(x', 1, n) .^ ((n − 1):−1:0)` is the Vandermonde matrix, and `vander(x)` creates the same matrix.
- `vander(x) \ y'` solves for the coefficients using the Vandermonde matrix.
- `polyfit(x, y, n − 1)` finds the same coefficients.

degree

spline



Spline

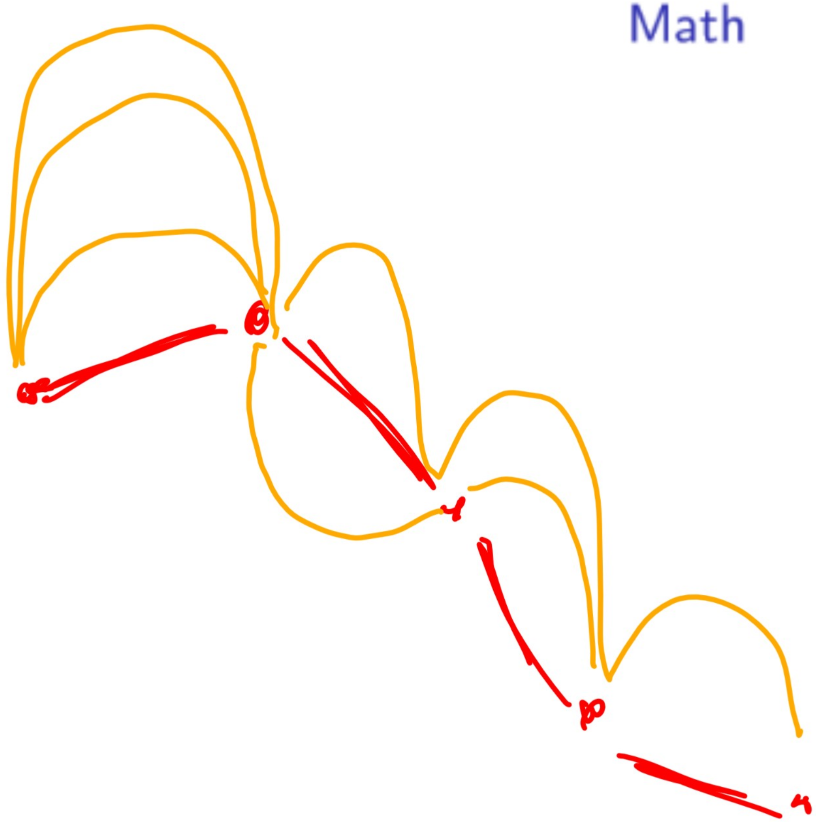
Math

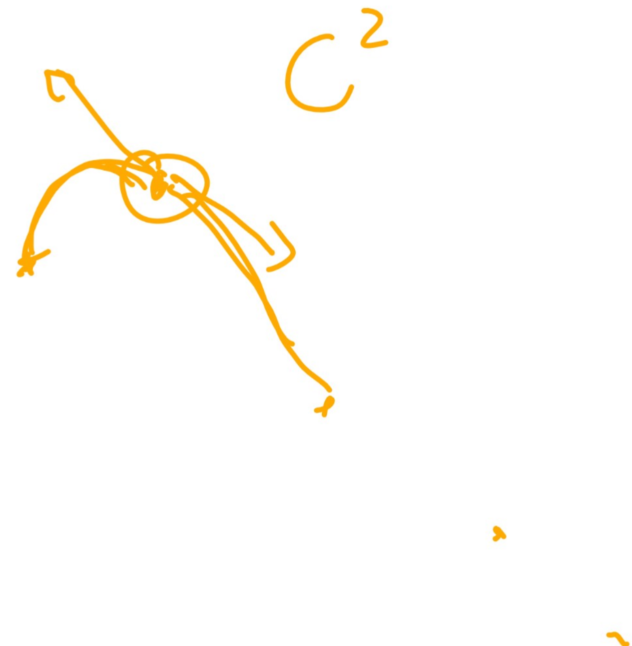
- A set of n points (with distinct x values, and ordered according to x) can define $n-1$ cubic polynomial segments satisfying the following conditions.

1. The i-th polynomial segment connects point i and point $i+1$.
2. The segments form a smooth curve. Technically, C^2 smoothness is desired: the first and second derivatives should be continuous.

Spline Diagram

Math





Tridiagonal Matrix

Math

- With n points, there are $4n - 4$ coefficients to compute. Suppose the coefficient for segment i connecting (x_i, y_i) and (x_{i+1}, y_{i+1}) is $c_{i1}, c_{i2}, c_{i3}, c_{i4}$.

1. $c_{i1}x_i^3 + c_{i2}x_i^2 + c_{i3}x_i + c_{i4} = y_i$ for $i = 1, 2, ..., n-1$ so that the segment passes through the left end point.
2. $c_{i1}x_{i+1}^3 + c_{i2}x_{i+1}^2 + c_{i3}x_{i+1} + c_{i4} = y_{i+1}$ for $i = 1, 2, ..., n-1$ so that the segment passes through the right end point.
3. $3c_{i,1}x_{i+1}^2 + 2c_{i2}x_{i+1} + c_{i3} = 3c_{i+1,1}x_{i+1}^2 + 2c_{i+1,2}x_{i+1} + c_{i+1,3}$ for $i = 1, 2, ..., n-2$ so that the first derivative is continuous.
4. $6c_{i1}x_{i+1} + 2c_{i2} = 6c_{i+1,1}x_{i+1} + 2c_{i+1,1}$ for $i = 1, 2, ..., n-2$ so that the second derivative is continuous.

- There are $4n - 6$ equations. 2 more equations are needed.

End Point Conditions

Math

- There are many ways to specify the 2 extra equations.

1. If the third derivative for the first and last two segments are continuous, $c_{1,1} = c_{2,1}$ and $c_{n-1,1} = c\,(n-2,1)$, the resulting spline is called a not-a-knot spline.
2. If the second derivative at the end points are 0, or $6c_{1,1}x_1 + 2c_{1,2} = 0$ and $6c_{n-1,1}x_n + 2c_{n-1,2} = 0$, the resulting spline is called a natural spline.

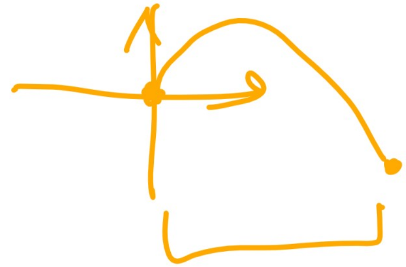
Spline

Code

- Suppose x, y are two row vectors with length n.
- *spline (x, y). coefs* finds the coefficients of the cubic spline that interpolates (x, y) while satisfying the not-a-knot condition.
- *spline (x, y, x0)* evaluates the cubic spline interpolating (x, y) at $x = x_0$.
- *csape(x, y, ' variational '). coefs* finds the coefficients of the natural cubic spline. It requires MATLAB's Curve Fitting Toolbox.

Local Coefficient

Code



- *spline(x, y).coefs* are local coefficients meaning the equation for segment i and $x \in [x_i, x_{i+1}]$ is $c_{i1}(x - x_i)^3 + c_{i2}(x - x_i)^2 + c_{i3}(x - x_i) + c_{i4}$.

P4

- The polynomial can be expanded to find the coefficients of the original spline problem.

Interpolation, Polyfit

Quiz

Q4

$1 \cdot x^4 + 1 \cdot x^3 + 1 \cdot x^2 + 1 \cdot x + 1$

1, 1, 1, 1, 1

1. $x = 1{:}5;\ y = polyval(ones(5,\ 1),\ x);\ \ polyfit\ (x,\ y,\ 4)$

degree

- B : **1 1 1 1 1**
- C : **0 0 0 0 1**

Interpolation, Polyfit Too

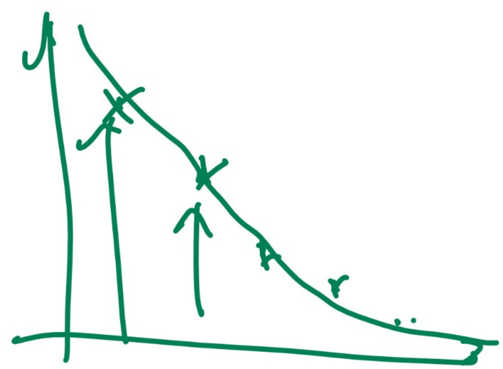
Quiz

Q5

1:5

1. *x = 1:5; y = 5:−1:1; polyval(polyfit (x, y, 4), x)*

- *C* : **5 4 3 2 1**
- *D* : **1 2 3 4 5**



Interpolation, Spline

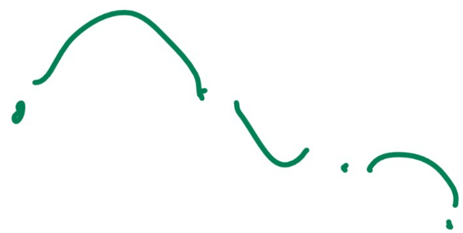
Quiz

Q6

don't use

→ polyval

1. $x = 1{:}5;\ y = 5{:}-1{:}1;\ spline(x,\ y,\ x)$
   - C : **5 4 3 2 1**
   - D : **1 2 3 4 5**



Interpolation, Spline Too

Quiz

1. $x = 1{:}5;\ y = 5{:}-1{:}1;\ spline(x,\ y).coefs(:,4)$

→ cubic polynomial Q?

- C: 4 3 2 1 D: 5 4 3 2

$x = x_i$

x_i x_{i+1}

$$f(x_i) = \underbrace{c_1 (x - x_i)^3}_{0} + \underbrace{c_2 (x - x_i)^2}_{0} + \underbrace{c_3 (x - x_i)}_{0} + c_4$$

Polynomial Regression

Math

- *polyfit (x, y, d)* with $d < n$, finds the polynomial that is the closest to the points (x, y).
- The total distance from the points to the polynomial is defined as the sum of the squared differences between the y value at x and the value of the polynomial at x,

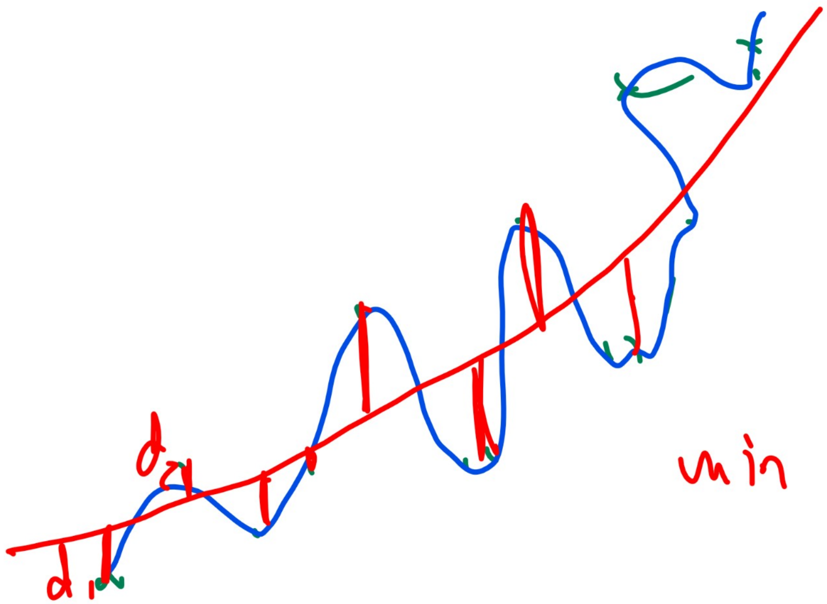
$$D = \sum_{i=1}^{n} \left(y_i - \sum_{j=0}^{d} c_j x_i^j \right)^2$$

$$= \sum_{i=1}^{n} \left(y_i - c_d x_i^d - c_{d-1} x_i^{d-1} - \ldots - c_1 x_i - c_0 \right)^2 .$$

- *polyfit* finds the coefficients of the polynomial that minimizes D. These coefficients are also called regression coefficients.

Regression Diagram

Math



$$\min \quad d_1^2 + d_2^2 + d_3^2 + \dots d_n^2$$

General Linear Regression

Math

- *regress (y ', X)* finds the regression coefficients for $c_n x_n + c_{n-1} x_{n-1} + ... + c_1 x_1 + c_0 x_0$, where $x_0, x_1, ..., x_n$ are the columns of X. It requires MATLAB's Statistics and Machine Learning Toolbox.
- *regress (y ', X)* solves $(X^T X) c = X^T y$ instead of $Xc = y$. This is because $X^T X$ is always invertible (meaning $(X^T X)^{-1}$ always exists), so $c = (X^T X)^{-1} X^T y$.
- When X is the degree d Vandermonde matrix for x, meaning $x_i = x^i$ for $i = d, d-1, ..., 0$, *regress (y, X)* finds the same coefficients as *polyfit (x, y, d)*.

Non-Polynomial Regression

Code

- *polyfit (f(x), y, 1)* finds the regression coefficients for $c_1 f(x) + c_0$, for example, *polyfit (sin (x), y, 1)* finds c_1, c_0 such that the total distance from the points to the function $\boxed{y = c_1 \sin(x) + c_0}$ is minimized.

$C_1 \sin(x) + C_2 \sin^2(x) + C_3$

- *regress (y', [f1(x') f2(x') ...])* finds the regression coefficients for $c_1 f_1(x) + c_2 f_2(x) + \ldots$, for example, *regress (y', [sin (x') cos(x') ones(length (x), 1)])* finds c_2, c_1, c_0 such that the total distance from the points to the function $\boxed{y = c_2 \sin(x) + c_1 \cos(x) + c_0}$ is minimized.

Regression, Polynomial

Quiz

- (The values of c, d such that the curve $y = cx^2 + d$ best fits (x, y) .)
- **1 1**

$y = cx^2 + bx + d$

1. $x = 1{:}10\ ;\ y = 1 + x\ .\hat{}\ 2;$

- A : *polyfit (x, y, 1)* $y = cx + d$
- B : ~~*polyfit*~~ *(x, y, 2)*
- C : *polyfit (x .^ 2, y, 1)*

polyfit (x, y , 2)

maybe $b \neq 0$

Q8

$c \cdot f(x) + d$

Regression, Unknown Degree

Quiz

$y = a f(x) + b$

Q9

- (The values of c, d such that the curve $y = cx^d$ best fits (x, y) .)
- **1 1**

$y = c x^d$

$\log y = \log c + d \log x$

$V(2)$ $V(1)$ $[c, d]$

1. $x = 1{:}10$; $y = x$;

- A : v = polyfit (log(x), log(y), 1); [exp(v(2)) v(1)]
- B : v = polyfit (x, log(y), 1); [exp(v(2)) v(1)]
- C : v = polyfit (log(x), log(y), 1); [v(2) v(1)]
- D : v = polyfit (x, log(y), 1); [v(2) v(1)]

Regression, Non-Polynomial

Quiz

Q 10

- (The values of c, d such that the curve $\boxed{y = ce^{dx}}$ best fits (x, y) .)
- **1 1**

$\log y = \boxed{\log c} + \boxed{d}x$

v(2) v(1)

1. $x = 1{:}10$; $y = exp(x)$;

- A : $v = polyfit\,(log(x),\ log(y),\ 1);\ [exp(v(2))\ \ v(1)]$
- B : $v = polyfit\,(x,\ log(y),\ 1);\ [exp(v(2))\ \ v(1)]$
- C : $v = polyfit\,(log(x),\ log(y),\ 1);\ [v(2)\ \ v(1)]$
- D : $v = polyfit\,(x,\ log(y),\ 1);\ [v(2)\ \ v(1)]$

spline

P4 due next Wed.

P1 - P3 code Monday.

Blank Slide