Matematikk 2 – øving 5 vår

Tore Bergebakken

Skrivebok først, MATLAB sist.

Sidene er scannet med
`scanimage --device "blabla" --format=png --output-file XX.png`
(jeg innser at kvaliteten ikke ble den beste, første gang jeg bruker scanimage...)

PDFene er stitchet sammen med `pdftk *.pdf cat output submit.pdf`

Oppgave 3.4.3a ble litt rotete siden jeg ikke kom på at vi skulle beregne c før jeg hadde gjort resten. Hadde også glemt at det bare var a-oppgaven som skulle gjøres…

Øving 5

Kapittel 3.1: 1a, 2a, 3, CP3

1a) Lagrange gjennom
$(0,1)$, $(2,3)$, $(3,0)$

3 punkter → $3-1=2$ graders polynom.

$$P_2(x) = y_1 \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} + y_2 \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} + \text{en til.}$$

$$= 1 \cdot \frac{(x-2)(x-3)}{(0-2)(0-3)} + 3 \cdot \frac{(x-0)(x-3)}{(2-0)(2-3)} + 0 \cdot \frac{(x-0)(x-2)}{(3-0)(3-2)}$$

$$= \frac{x^2-2x-3x+6}{-6} + \frac{3(x^2-3x)}{2(-1)} + 0$$

$$= -\frac{1}{6}x^2 - \frac{3}{2}x^2 + x + \frac{9}{2}x - 1$$

$$= -\left(\frac{1+9}{6}\right)x^2 + \frac{11}{2}x - 1 = \underline{\underline{-\frac{5}{3}x^2 + \frac{11}{2}x - 1}}$$

ez.

2a) $f[x_1 \ldots x_n]$ = koeffisient for x^{n-1} i interpolerende polynom.

Rekursjon på papir?

x_i	y_i		
0	1		
		$\frac{3-1}{2-0} = \frac{2}{2} = 1$	
2	3		$\frac{-3-1}{3-0} = -\frac{4}{3}$
		$\frac{0-3}{3-2} = -3$	
3	0		

$$P_2(x) = 1 + 1(x-0) - \frac{4}{3}(x-0)(x-2)$$
$$= 1 + (x-0)\left(1 + (x-2)\left(-\frac{4}{3}\right)\right)$$
$$= 1 + x\left(-\frac{4}{3}x + 1 + 2\cdot\frac{4}{3}\right)$$
$$= -\frac{4}{3}x^2 + \frac{3+8}{3}x + 1 = \underline{\underline{-\frac{4}{3}x^2 + \frac{11}{3}x + 1}}$$

3) Setter opp dividerte differanser:

aach mellomregning...

-1	3			
		$\frac{1-3}{1-(-1)} = \frac{-2}{2} = -1$		
1	1		$\frac{2-(-1)}{2-(-1)} = 1$	
		$\frac{3-1}{2-1} = 2$		$\frac{1-1}{3-(-1)} = 0$ ...
2	3		$\frac{4-2}{3-1} = 1$	
		$\frac{7-3}{3-2} = 4$		
3	7			

NEI en av dem er

~~Ingen av koeffisientene er 0~~

=> et 2. gradspolynom kan ikke interpolere punktene

a) d=2 dagen

3) forts.

a) $d = 2$: ett, siden kun ett polynom av grad 3 eller mindre vil interpolere – og m. av koeff. er 0

$$P_2(x) = 3 - 1(x+1) + 1\cdot(x+1)(x-1) + 0$$

b) $d = 3$: Ingen, siden fjerde koeffisient blir 0.

c) $d = 6$: uendelig mange.

$$P_6(x) = P_2(x) + c\,(x+1)\,x^3\,(x-1)(x-2)$$

for $c \neq 0$.

5)

a)

x	y			
-2	8			
		$\frac{4-8}{0-(-2)} = -2$		
0	4		$\frac{-2-(-2)}{1-(-2)} = 0$	
		$\frac{2-4}{1-0} = -2$		$\frac{0-0}{3-(-2)} = 0$
1	2		$\frac{-2-(-2)}{3-0} = 0$	
		$\frac{-2-2}{3-1} = \frac{-4}{2} = -2$		
3	-2			

$$P(x) = 8 - 2(x-(-2)) = -2x + 4$$ (rett)

b) kun av grad 4

$$P_4(x) = P(x) + c(x+2)\,x\,(x-1)(x-3)$$ for $c \neq 0$

Kapitte 3.2 - this is straight forw.

1a) Punktene

0	0
$\frac{\pi}{2}$	1
π	0

$$\frac{1-0}{\pi/2-0} = \frac{2}{\pi}$$

$$\frac{0-1}{\pi-\pi/2} = -\frac{2}{\pi}$$

$$\frac{-2/\pi - 2/\pi}{\pi - 0} = \frac{-2\cdot 2}{\pi\cdot\pi} = -\frac{4}{\pi^2}$$

$$P_2(x) = 0 + \frac{2}{\pi}(x-0) - \frac{4}{\pi^2}(x-0)(x-\frac{\pi}{2})$$

$$= \frac{2}{\pi}x - \frac{4}{\pi^2}x^2 + \frac{4}{\pi^2}\cdot\frac{\pi}{2}x$$

$$= \frac{4}{\pi}x - \frac{4}{\pi^2}x^2 = \frac{4}{\pi}x(1-\frac{x}{\pi})$$

1b) $P_2(\pi/4) = \frac{4}{\pi}\cdot\frac{\pi}{4} - \frac{4}{\pi^2}\cdot\frac{\pi^2}{16} = 1 - \frac{1}{4} = \frac{3}{4}$

$= 0,75$

1c) Makestells

$$|\sin x - P_2(x)| \le \left|\frac{(x-0)(x-\frac{\pi}{2})(x-\pi)}{3!}\cdot(\sin c)'''\right|$$

$$|\sin x - P_2(x)| \le \frac{|x(x-\frac{\pi}{2})(x-\pi)|}{6}|-\sin c|$$

$|\sin c|$ er maksimalt lik 1.

$$|\sin\frac{\pi}{4} - P_2(\frac{\pi}{4})| \le \frac{|\frac{\pi}{4}(\frac{\pi}{4}-\frac{\pi}{2})(\frac{\pi}{4}-\pi)|}{6}\cdot 1$$

$$= \frac{1}{6}|\frac{\pi}{4}(-\frac{\pi}{4})(-\frac{3\pi}{4})| \simeq 0,2422$$

1d) Kalkulator:

$$\sin(\pi/4) = \frac{\sqrt{2}}{2} \approx 0{,}7071$$

Feil sammenliknet m/kalk:

$$\left|\sin\left(\frac{\pi}{4}\right) - P_2\left(\frac{\pi}{4}\right)\right| = \left|\frac{\sqrt{2}}{2} - \frac{3}{4}\right| \approx \underline{0{,}04289}$$

Sammenliknet m/feilgrense:

$$\frac{0{,}04289}{0{,}2422} \approx 0{,}1770$$

$\Rightarrow$ ca. **18%** mindre feil enn forventet!

Kapittel 3,4:

$$(c(x-1)^2)'' = (2c(x-1))' = (2cx-2c)' = 2c$$

3 a) Natural spline?

$$S_1''(x_1) = 0 \quad \text{og} \quad S_{n-1}''(x_n) = 0$$

Her

$$S(x) = \begin{cases} 4 - \frac{11}{4}x + \frac{3}{4}x^3 & i\ [0,1] \\ 2 - \frac{1}{2}(x-1) + c(x-1)^2 - \frac{3}{4}(x-1)^3 & i\ [1,2] \end{cases}$$

så vi tester

$$S_1''(x) = \frac{3}{4}3\cdot 2x$$

$$S_1''(x_1) = S_1''(0) = \frac{9}{2}\cdot 0 = \underline{0} \text{ yess}$$

$$S_2''(x) = \frac{3}{4}\cdot 3\cdot 2(x-1) + 2c = \frac{9}{2}(x-1) + 2c$$

$$S_2''(2) = -\frac{9}{2}(2-1) + 2c = -\frac{9}{2} + 2c \Rightarrow c = \frac{9}{4}$$

$\Rightarrow$ ikke natural spline.

hvis $c \neq \frac{9}{4}$

oppfyller natural spline siden $c = \frac{9}{4}$, se neste side.

3a forts.)

Parabolisk termineret?

Første og siste spline
er max grad 2.

$\Rightarrow$ siden $S_2(x)$ og $S_1(x)$
er tredjegradsfunksjoner, er
ikke dette oppfylt.

Not-a-knot?

Må ha $S_1'''(x_2) = S_2'''(x_2)$
og $S_{n-2}'''(x_{n-1}) = S_{n-1}'''(x_{n-1})$

Siden $n-1=2$ og $n-2=1$
i dette tilfellet, er
dette likningene ikke.

$S_1'''(x) = \frac{3}{4} \cdot 3 \cdot 2 \cdot 1 = \frac{9}{2}$

$S_2'''(x) = -\frac{3}{4} \cdot 3 \cdot 2 \cdot 1 = -\frac{9}{2}$

$S_1'''(1) = \frac{9}{2} \neq -\frac{9}{2} = S_2'''(1)$

så ikke not-a-knot.

---

men hva er c?

$S_2''(x_1) = S_2''(1) = \frac{9}{2}(1-1) + 2c = 2c$

$S_1''(1) = \frac{9}{2} \cdot 1 = \frac{9}{2}$

$S_2''(x_1) = S_1''(x_1)$

$2c = \frac{9}{2}$

$c = \frac{9}{4} \Rightarrow$ er natural spline...

3b) $$S(x) = \begin{cases} 3 - 4x + 4x^2 & i\ [0,1] \\ -2-(x-1)+c(x-1)^2 & i\ [1,2] \end{cases}$$

Natural spline?

$S''(x) = 4 \cdot 2 \cdot 1 = 8 \neq 0$ for alle x

så... nei.

Parabolsk terminert?

Ja, begge endesplinene er andregradsfunksjoner.

Not-a-knot?

Ja, siden S_1 og S_2 er av andre grad vil $S_1(x)''' = S_2(x)''' = 0$.

3c) c er koeffisienten til 2.gradsleddet.

$S_2''(x) = 2c$, ifølge egenskap 3

må $S_1''(x_2) = S_2''(x_2)$:

$S_1''(1) = 8 = S_2''(1) = 2c$

$\Rightarrow c = \frac{8}{2} = 4$

3c) $S(x)=\begin{cases} -2-\frac{3}{2}x+\frac{7}{2}x^2-x^3 & i\ [0,1] \\ -1+c(x-1)+\frac{1}{2}(x-1)^2-(x-1)^3 & i\ [1,2] \\ 1+\frac{7}{2}(x-2)-\frac{5}{2}(x-2)^2-(x-2)^3 & i\ [2,3] \end{cases}$

$n=4$ her?

egenskap 2 =

$S'_{i-1}(x_i)=S'_i(x_i)$ for $i=1,2,3$

$S'_1(x_2)=S'_2(x_2)$ og $S'_2(x_3)=S'_3(x_3)$

$S'_1(x)=-\frac{3}{2}+7x-3x^2$

$S'_2(x)=c+2(x-1)-3(x-1)^2$

$S'_3(x)=\frac{7}{2}-5(x-2)-3(x-2)^2$

$S_1'(1)=-\frac{3}{2}+7-3=4-\frac{3}{2}=\frac{8-3}{2}=\frac{5}{2}$

$S_2'(1)=c+0-0=c$

$\Rightarrow \underline{\underline{c=\frac{5}{2}}}$, done.

---

natural spline?

$S_1''(x)=7-6x$

$S_1''(0)=7-0=7\neq 0$

$\Rightarrow$ <u>nei</u>.

parabolsk terminert? Nei, alle splines er 3. grads!

Ferdig.

3c) forts.

not-a-knot?

$S_1'''(1) = S_2'''(1)$ og $S_2'''(2) = S_3'''(2)$

$S_1'''(x) = -3 \cdot 2 \cdot 1 = -6$ => Ikke for alle x

$S_2'''(x) = -3 \cdot 2 \cdot 1 = -6$ => Ikke ~~for~~ alle x

$S_3'''(x) = -3 \cdot 2 \cdot 1 = -6$

Yes, not-a-knot.

---

kapittel 3.5: Bezierkurver?

1a) Har endepunkter $(x_1, y_1) = (0, 0)$
og $(x_4, y_4) = (1, 0)$

og kontrollpunkter $(x_2, y_2) = (0, 2)$
og $(x_3, y_3) = (2, 0)$

koeffisientene blir

$b_x = 3(x_2 - x_1) = 3(0-0) = \underline{0}$

$c_x = 3(x_3 - x_2) - b_x = 3(2-0) - 0 = \underline{6}$

$d_x = x_4 - x_1 - b_x - c_x = 1 - 0 - 6 - 0 = -5$

$b_y = 3(y_2 - y_1) = 3(2-0) = \underline{6}$

$c_y = 3(y_3 - y_2) - b_y = 3(0-2) - 6 = \underline{-12}$

$d_y = y_4 - y_1 - b_y - c_y = 0 - 0 - 6 - (-12) = \underline{6}$

1a) forts.

$$(x(t), y(t)) = \left(\cancel{x_1} + \cancel{b_x t} + c_x t^2 + d_x t^3,\ \cancel{x_1} + b_y t + c_y t^2 + d_y t^3\right)$$

$$= \underline{\underline{\left(6t^2 - 5t^3,\ 6t - 12t^2 + 6t^3\right)}}$$

2a)

Kurven $\begin{cases} x(t) = 1 + 6t^2 + 2t^3 \\ y(t) = 1 - t + t^3 \end{cases}$

Konstantleddene gir at første endepkt. er $(x_1, y_1) = \underline{(1, 1)}$

$b_x = 3(x_2 - x_1)$

$0 = 3(x_2 - 1) = 3x_2 - 3$

$3 = 3x_2 \iff \underline{x_2 = 1}$

$b_y = 3(y_2 - y_1)$

$-1 = 3(y_2 - 1)$

$-1 + 3 = 3y_2 \iff y_2 = \frac{2}{3}$

Første kontrollpkt. $(x_2, y_2) = \underline{\underline{(1, \frac{2}{3})}}$

$6 = c_x = 3(x_3 - x_2) - b_x = 3(x_3 - 1) - 0$

$6 + 3 = 3x_3 \iff x_3 = \frac{9}{3} = \underline{3}$

$0 = c_y = 3(y_3 - y_2) - b_y = 3(y_3 - \frac{2}{3}) - (-1)$

$0 = 3y_3 - 3 \cdot \frac{2}{3} + 1 = 3y_3 - 1$

$\iff y_3 = \frac{1}{3}$

$\Rightarrow$ tredje kontr.pkt.: $(x_3, y_3) = \underline{\underline{(3, \frac{1}{3})}}$

2a) forts.

Siste løkningene...

$$d_x = x_4 - x_1 - b_x - c_x$$

$$2 = x_4 - 1 - 0 - 6$$

$$x_4 - 7 = 2$$

$$x_4 = 2 + 7 = \underline{9}$$

$$d_y = y_4 - y_1 - b_y - c_y$$

$$1 = y_4 - 1 - (-1) - 0$$

$$y_4 - 1 + 1 = 1$$

$$y_4 = \underline{1}$$

Andre endepunktet:

$$\underline{(x_4, y_4) = (9, 1)}$$

Oppsummert har vi pkt. ene

$$\underline{\underline{(1,1),\ (1, \tfrac{2}{3}),\ (3, \tfrac{1}{3}) \text{ og } (9,1)}}$$

(opprinnelig fant jeg andre kontr.pkt. som $(3, \underline{1})$ men det så rart ut i Geogebra.)

3.1 Interpolating Polynoms

CP3

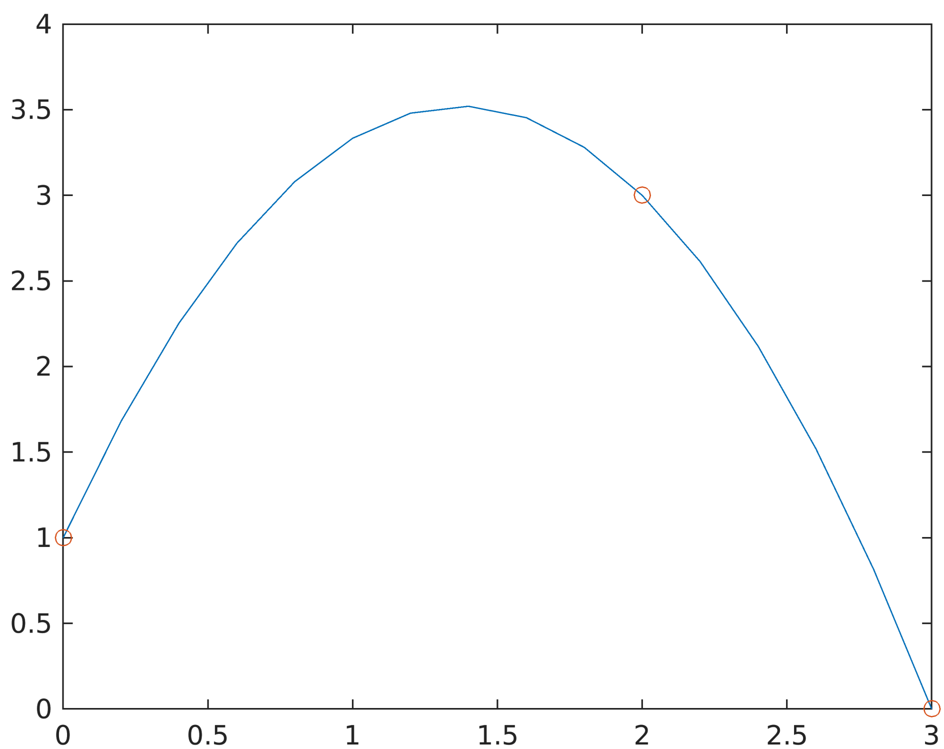
Verifying that my function works by checking that inputting x_i gives the correct y_i

```
x=[0 2 3]';
y=[1 3 0]';
for i=1:3
    sprintf('at x=%d, %.4f should be %d', x(i), polyinterp(x,y,x(i)), y(i))
end
```

```
ans =
'at x=0, 1.0000 should be 1'
ans =
'at x=2, 3.0000 should be 3'
ans =
'at x=3, 0.0000 should be 0'
```

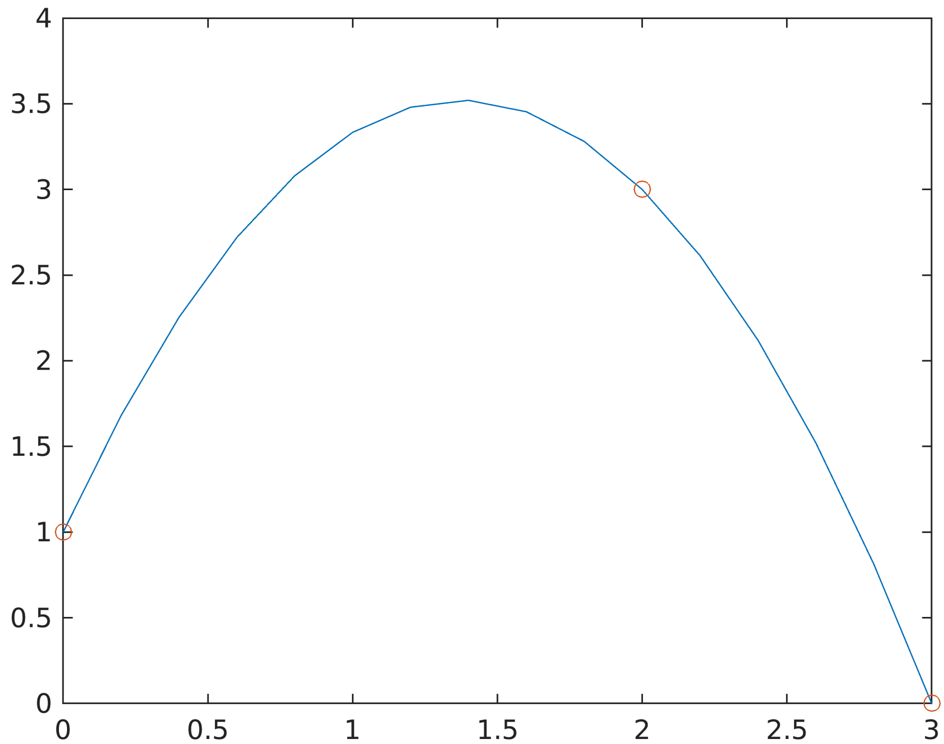
Might as well draw a graph too:

```
xaxis=(x(1):0.2:x(3))';
[n, ~] = size(xaxis);
for i=1:n
    yaxis(i) = polyinterp(x,y,xaxis(i));
end
hold off;
```




```
plot(xaxis, yaxis)
```

```
hold on;
plot(x,y, 'o')
```




```
function y0 = polyinterp(x,y,x0)
% POLYINTERP lalala
[n, ~] = size(x);
% Find interpolation coefficients
% using Newton's divided differences:
c = newtdd(x,y,n);
% Then evaluate the polynomial using nest:
y0 = nest(n-1,c,x0,x);
end

function c=newtdd(x,y,n)
% NEWTDD fra boka
% trekanten lagres slik i v-matrisen:
% f[x1] f[x1 x2] f[x1 x2 x3]
% f[x2] f[x2 x3]
% f[x3]
v = zeros(n,1);
for j=1:n
    v(j,1)=y(j);  % Fill in y column of Newton triangle
end
for i=2:n          % For column i,
    for j=1:n+1-i % fill in column from top to bottom
        % eksempel:
        % v(1,2) = (f[x2] - f[x1]) / (x2 - x1)
```

```
        v(j,i) = (v(j+1,i-1) - v(j,i-1)) / (x(j+i-1) - x(j));
    end
end
c = zeros(n,1);
for i=1:n
    c(i)=v(1,i); % Read along top of triangle
end              % for output coefficients
end
```

3.4 Cubic Splines

CP1a

Find the equations and plot the natural cubic spline that interpolates the data points

(a) (0,3),(1,5),(2,4),(3,1)

Comparing against example code from the book.

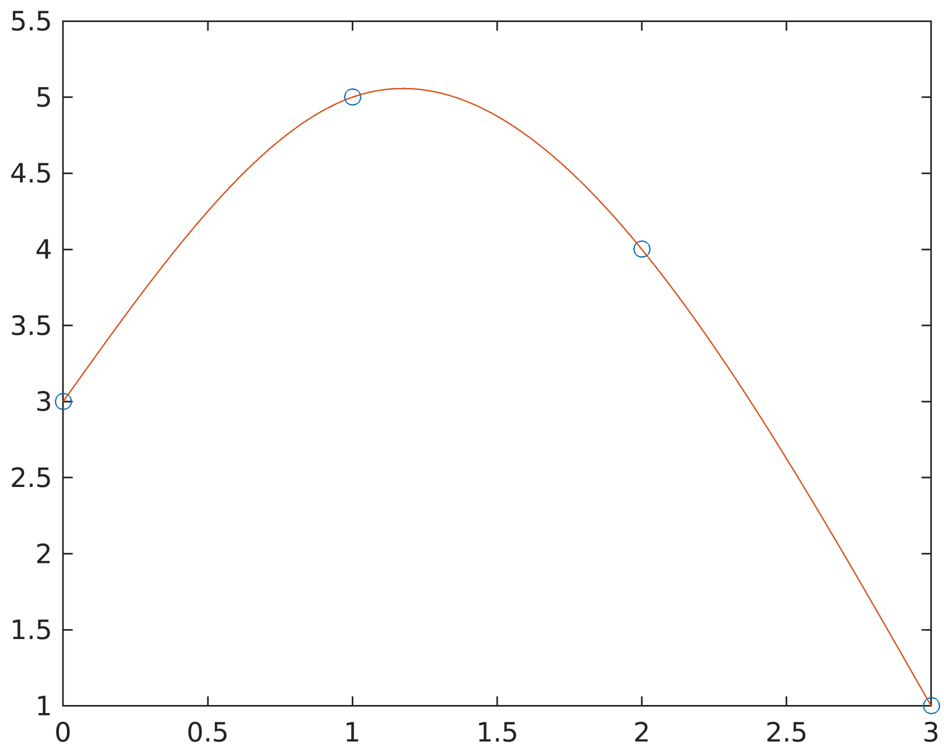
```
x = [0 1 2 3]';
y = [3 5 4 1]';
coeff = mysplinecoeff(x,y,"Natural")
```

```
coeff = 3×3
    2.6667         0   -0.6667
    0.6667   -2.0000    0.3333
   -2.3333   -1.0000    0.3333
```

```
bookcoeff = naturalsplinecoeff(x,y)
```

```
bookcoeff = 3×3
    2.6667         0   -0.6667
    0.6667   -2.0000    0.3333
   -2.3333   -1.0000    0.3333
```

```
mysplineplot(x,y,coeff,50)
```



CP2a

Same data points as above.

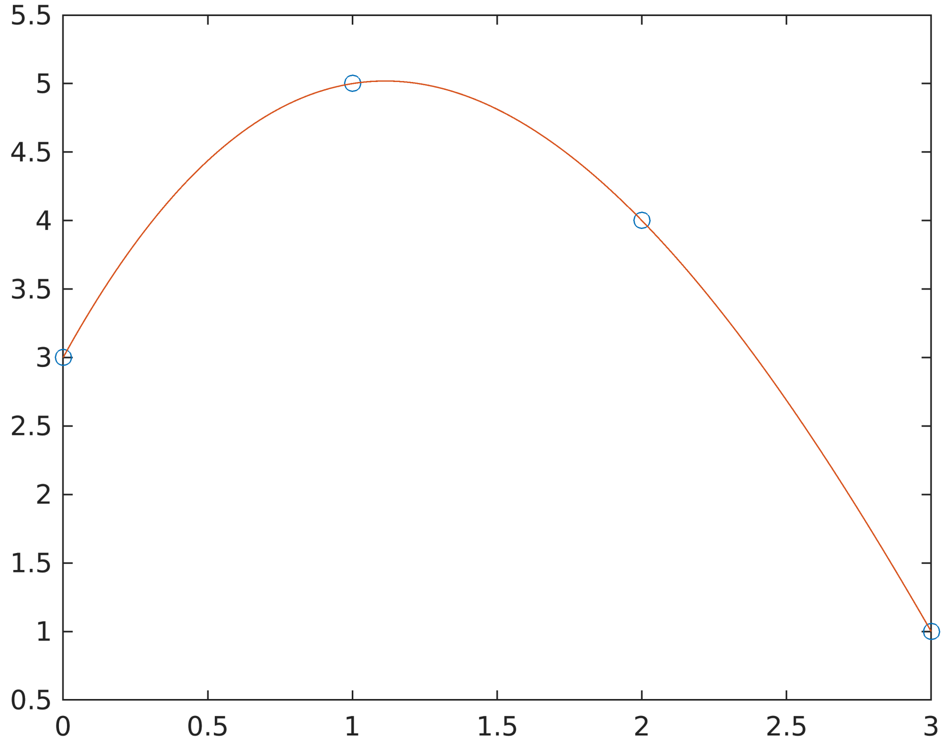
```
coeff = mysplinecoeff(x, y, "Not-a-knot")
```

```
coeff = 3×3
    3.8333   -2.0000    0.1667
    0.3333   -1.5000    0.1667
   -2.1667   -1.0000    0.1667
```

```
bookcoeff = notaknotsplinecoeff(x,y)
```

```
bookcoeff = 3×3
    3.8333   -2.0000    0.1667
    0.3333   -1.5000    0.1667
   -2.1667   -1.0000    0.1667
```

```
mysplineplot(x,y,coeff,50)
```



Yeah, I don't see *that* much of a difference between those two graphs.

The not-a-knot version does look a bit fatter, though.

```
function coeff=mysplinecoeff(x,y,mode)
% MYSPLINECOEFF yes it's my code haha
%               made with a great deal of
%               matlab pain (TM)

% hmmm yeah that's a better way to get *one* dimension
% than the [n, ~] = size(x) bullshit I've done.
```

```
n=length(x);

% Create del and delta vectors
del   = zeros(n-1,1);
delta = zeros(n-1,1);
for i=1:n-1
    del(i)   = x(i+1) - x(i);
    delta(i) = y(i+1) - y(i);
end

% create modified del vectors for diagonals
% with zeros in top&bottom rows
del1 = [0; del(1:n-2); 0];
del2 = [0; del(2:n-1); 0];
mid = 2*del1 + 2*del2;
% modify del1 and del2 again
% to fix whatever problem spdiags had...
del1 = [del1(2:n); 0];
del2 = [0; del2(1:n-1)];

% Create sparse matrix A
A = spdiags([del1, mid, del2], [-1,0,1], n, n);

% Set specifics of A based on end condition
if mode == "Natural"
    A(1,1) = 1;
    A(n,n) = 1;
elseif mode == "Not-a-knot"
    A(1,1:3) = [del(2), -(del(1) + del(2)), del(1)];
    A(n,n-2:n) = [del(n-1), -(del(n-2) + del(n-1)), del(n-2)];
else
    disp('yeah ok idk what you want')
end

% Create right-hand-side-vector
r = [0; 3*((delta(2:n-1) ./ del(2:n-1)) - (delta(1:n-2) ./ del(1:n-2))); 0;];
% Solve 3.24 or its variants
c = A\r;

% find b and d
d = zeros(n-1,1);
b = zeros(n-1,1);
for i=1:n-1
    d(i) = (c(i+1) - c(i)) / (3*del(i));
    b(i) = (delta(i)/del(i)) - (del(i)/3) * (2*c(i) + c(i+1));
end
coeff = [b c(1:n-1) d];
end

function mysplineplot(x,y,coeff,m)
n=length(x);
rang = (x(1):1/m:x(4))';
for i=1:n-1 % coeff rows
    for j=(i-1)*m+1:i*m+1
```

```
        % prepend y(i) to coefficient row!
        graph(j) = nest(3,[y(i) coeff(i,:)]',rang(j),repelem(x(i),3));
    end
end
plot(x,y,'o',rang,graph)
end

% BOOOOOOK....

function coeff=naturalsplinecoeff(x,y)
% from book
n=length(x);v1=0;vn=0;
A=zeros(n,n); % matrix A is nxn
r=zeros(n,1);
for i=1:n-1 % define the deltas
    dx(i)= x(i+1)-x(i);
    dy(i)=y(i+1)-y(i);
end
for i=2:n-1 % load the A matrix
    A(i,i-1:i+1)=[dx(i-1) 2*(dx(i-1)+dx(i)) dx(i)];
    r(i)=3*(dy(i)/dx(i)-dy(i-1)/dx(i-1)); % right-hand side
end

% Set endpoint conditions
% Use only one of following 5 pairs:
A(1,1) = 1; % natural spline conditions
A(n,n) = 1;
%A(1,1)=2;r(1)=v1; % curvature-adj conditions
%A(n,n)=2;r(n)=vn;
%A(1,1:2)=[2*dx(1) dx(1)];r(1)=3*(dy(1)/dx(1)-v1); %clamped
%A(n,n-1:n)=[dx(n-1) 2*dx(n-1)];r(n)=3*(vn-dy(n-1)/dx(n-1));
%A(1,1:2)=[1 -1]; % parabol-term conditions, for n>=3
%A(n,n-1:n)=[1 -1];
%A(1,1:3)=[dx(2) -(dx(1)+dx(2)) dx(1)]; % not-a-knot, for n>=4
%A(n,n-2:n)=[dx(n-1) -(dx(n-2)+dx(n-1)) dx(n-2)];
coeff=zeros(n,3);
coeff(:,2)=A\r;
% solve for c coefficients
for i=1:n-1 % solve for b and d
coeff(i,3)=(coeff(i+1,2)-coeff(i,2))/(3*dx(i));
coeff(i,1)=dy(i)/dx(i)-dx(i)*(2*coeff(i,2)+coeff(i+1,2))/3;
end
coeff=coeff(1:n-1,1:3);
end

function coeff=notaknotsplinecoeff(x,y)
% from book
n=length(x);v1=0;vn=0;
A=zeros(n,n); % matrix A is nxn
r=zeros(n,1);
for i=1:n-1 % define the deltas
    dx(i)= x(i+1)-x(i);
    dy(i)=y(i+1)-y(i);
end
```

```
for i=2:n-1 % load the A matrix
    A(i,i-1:i+1)=[dx(i-1) 2*(dx(i-1)+dx(i)) dx(i)];
    r(i)=3*(dy(i)/dx(i)-dy(i-1)/dx(i-1)); % right-hand side
end

% Set endpoint conditions
A(1,1:3)=[dx(2) -(dx(1)+dx(2)) dx(1)]; % not-a-knot, for n>=4
A(n,n-2:n)=[dx(n-1) -(dx(n-2)+dx(n-1)) dx(n-2)];
coeff=zeros(n,3);
coeff(:,2)=A\r;
% solve for c coefficients
for i=1:n-1 % solve for b and d
coeff(i,3)=(coeff(i+1,2)-coeff(i,2))/(3*dx(i));
coeff(i,1)=dy(i)/dx(i)-dx(i)*(2*coeff(i,2)+coeff(i+1,2))/3;
end
coeff=coeff(1:n-1,1:3);
end

% plot thing from book too
% modified to take coeffs in!
% ...wait, does it *seriously* not use nest?
% yeah, I'm writing this myself, thank you very much.
function [x1,y1]=splineplot(x,y,coeff,k)
n=length(x);
x1=[]; y1=[];
for i=1:n-1
    xs=linspace(x(i),x(i+1),k+1);
    dx=xs-x(i);
    ys=coeff(i,3)*dx; % evaluate using nested multiplication
    ys=(ys+coeff(i,2)).*dx;
    ys=(ys+coeff(i,1)).*dx+y(i);
    x1=[x1; xs(1:k)'];
    y1=[y1;ys(1:k)'];
end
x1=[x1; x(end)];y1=[y1;y(end)];
plot(x,y,'o',x1,y1)
end
% (which means, this function is *not* used...)
```