Comp 350-Assignment 5

Ian Benlolo 260744397
McGill University
November 15, 2017

1. **Vandermonde Form**
We start by building the Vandermonde matrix A and the equality $Ac = y$.

$$\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & 2 & 2^2 & 2^3\\ 1 & 3 & 3^2 & 3^3\\ 1 & 4 & 4^2 & 4^3\end{bmatrix}\begin{bmatrix}c_0\\ c_1\\ c_2\\ c_3\end{bmatrix} = \begin{bmatrix}y_0\\ y_1\\ y_2\\ y_3\end{bmatrix} = \begin{bmatrix}2\\ 0\\ -10\\ -34\end{bmatrix}$$

We solve this with GEPP.

$$\left[\begin{array}{cccc|c}1 & 1 & 1 & 1 & 2\\ 1 & 2 & 2^2 & 2^3 & 0\\ 1 & 3 & 3^2 & 3^3 & -10\\ 1 & 4 & 4^2 & 4^3 & -34\end{array}\right] \xrightarrow[R_{2,3,4}-R_1]{} \left[\begin{array}{cccc|c}1 & 1 & 1 & 1 & 2\\ 0 & 1 & 3 & 7 & -2\\ 0 & 2 & 8 & 26 & -12\\ 0 & 3 & 15 & 63 & -36\end{array}\right] \xrightarrow[R_3-2R_2]{R_4-3R_2} \left[\begin{array}{cccc|c}1 & 1 & 1 & 1 & 2\\ 0 & 1 & 3 & 7 & -2\\ 0 & 0 & 2 & 12 & -8\\ 0 & 0 & 6 & 42 & -30\end{array}\right]$$

$$\xrightarrow[R_4-3R_3]{R_3/2} \left[\begin{array}{cccc|c}1 & 1 & 1 & 1 & 2\\ 0 & 1 & 3 & 7 & -2\\ 0 & 0 & 1 & 6 & -4\\ 0 & 0 & 0 & 6 & -6\end{array}\right] \xrightarrow{R_4/6} \left[\begin{array}{cccc|c}1 & 1 & 1 & 1 & 2\\ 0 & 1 & 3 & 7 & -2\\ 0 & 0 & 1 & 6 & -4\\ 0 & 0 & 0 & 1 & -1\end{array}\right]$$

From this we get $c_3 = -1, c_2 = 2, c_1 = -1, c_0 = 2$.
$\therefore p_3(x) = c_0 + c_1x + c_2x^2 + c_3x^3 = 2 - x + 2x^2 - x^3$

Lagrange Form

$L_1(x) = \frac{(x-2)(x-3)(x-4)}{(1-2)(1-3)(1-4)} = -\frac{x^3}{6} + \frac{3x^2}{2} - \frac{13x}{3} + 4, L_2(x) = \frac{(x-1)(x-3)(x-4)}{(2-1)(2-3)(2-4)} = \frac{1}{2}(x^3 - 8x^2 + 19x - 12), L_3(x) = \frac{(x-1)(x-2)(x-4)}{(3-1)(3-2)(3-4)} = -\frac{x^3}{2} + \frac{7x^2}{2} - 7x + 4, L_4(x) = \frac{(x-1)(x-2)(x-3)}{(4-1)(4-2)(4-3)} = \frac{1}{6}(x^3 - 6x^2 + 11x - 6)$

Now,
$p_3(x) = 2 \times L_1(x) + 0 \times L_2(x) - 10 \times L_3(x) - 34 \times L_4(x) = \cdots = -x^3 + 2x^2 - x + 2$

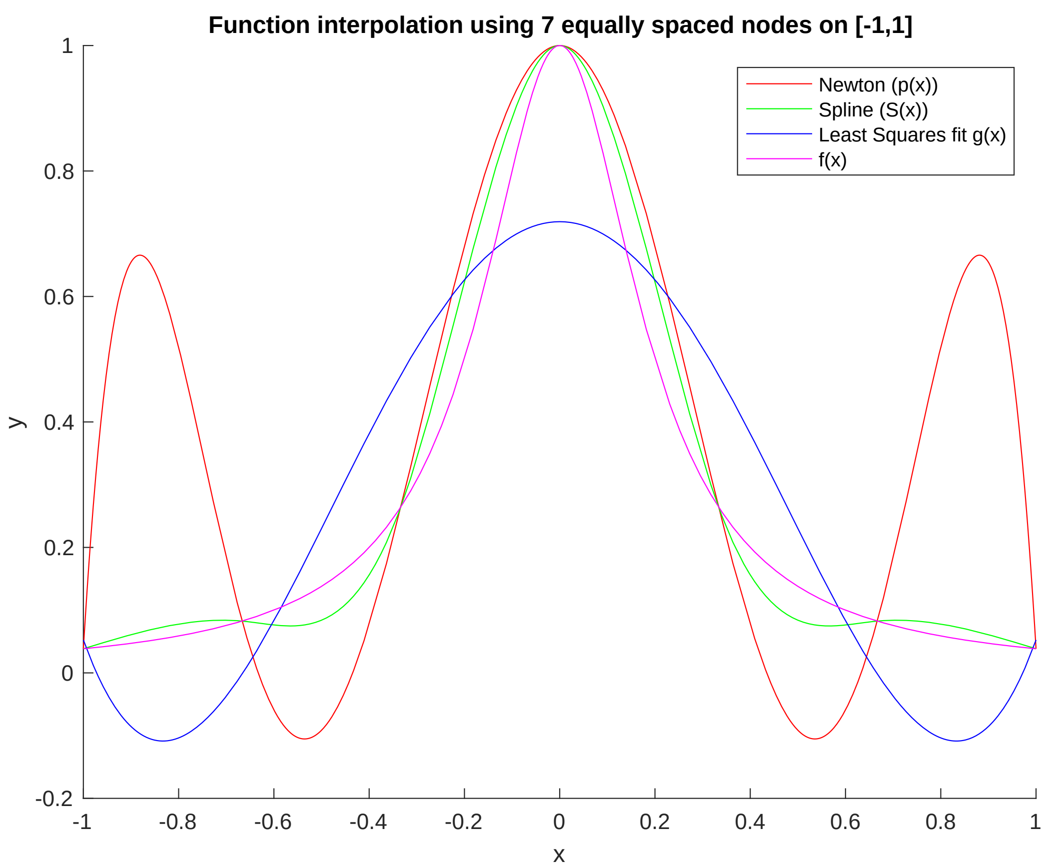
Newton Form We start by calculating $a_{k's}$:
$a_0 = 2$
$a_1 = f[x_0, x_1] = \frac{f[x_1]-f[x_0]}{x_1-x_0} = \frac{0-2}{2-1} = -2$
$f[x_1, x_2] = \frac{f[x_2]-f[x_1]}{x_2-x_1} = \frac{-10-0}{3-2} = -10$
$\therefore a_2 = f[x_0, x_1, x_2] = \frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0} = \frac{-10-(-2)}{3-1} = -4$
$f[x_2, x_3] = \frac{f[x_3]-f[x_2]}{x_3-x_2} = \frac{-34-(-10)}{4-3} = -24$
$f[x_1, x_2, x_3] = \frac{f[x_2,x_3]-f[x_1,x_2]}{x_3-x_1} = \frac{-24-(-10)}{4-2} = -7$
$\therefore a_3 = f[x_0, x_1, x_2, x_3] = \frac{f[x_1,x_2,x_3]-f[x_0,x_1,x_2]}{x_3-x_0} = \frac{-7-(-4)}{4-1} = -1$

Now,

$$\begin{aligned} p_3(x) &= a_0 + a_1(x-x_0) + a_2(x-x_0)(x-x_1) + a_3(x-x_0)(x-x_1)(x-x_2)\\ &= 2 - 2(x-1) - 4(x-1)(x-2) - 1(x-1)(x-2)(x-3) = \cdots = -x^3 + 2x^2 - x + 2\end{aligned}$$

2. (a)

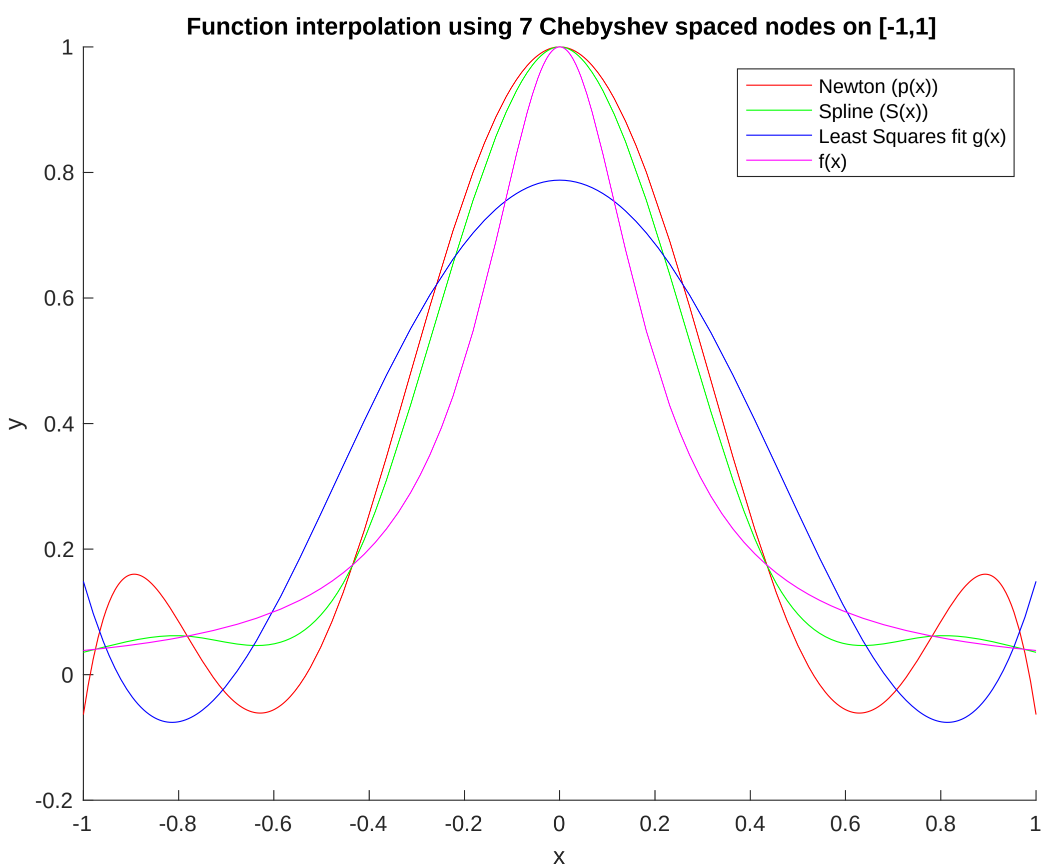
```
>> main2
Coefficients of p(x), x0...x6 =
    0.0385    0.1323    0.6211    1.8681   -8.2312   13.1349  -13.1349

Coefficients of S(x), x0...x6 =
         0   -1.8181   14.7262  -27.2160   14.7262   -1.8181         0

Least  square coefficients g(x) (a,b,c respectively)=
    0.7192
   -2.3862
    1.7195

f(x)=
  Columns 1 through 10

    0.0385    0.0545    0.0826    0.1379    0.2647    0.5902    1.0000    0.5902    0.2647↙
0.1379

  Columns 11 through 13

    0.0826    0.0545    0.0385

f(x)-p(x)=
  Columns 1 through 10

         0   -0.5534   -0.0000    0.2293         0   -0.1817         0   -0.1817    0.0000↙
0.2293

  Columns 11 through 13

   -0.0000   -0.5534    0.0000

f(x)-S(x) =
  Columns 1 through 10

         0   -0.0187   -0.0000    0.0539   -0.0000   -0.1289         0   -0.1289         0↙
0.0539

  Columns 11 through 13

   -0.0000   -0.0187    0.0000

f(x)-g(x)=
  Columns 1 through 10

   -0.0140    0.1631    0.0842   -0.0922   -0.2106   -0.0641    0.2808   -0.0641   -0.2106↙
-0.0922

  Columns 11 through 13

    0.0842    0.1631   -0.0140

>>
```



Function interpolation using 7 equally spaced nodes on [-1,1]
Newton (p(x))
Spline (S(x))
Least Squares fit g(x)
f(x)
y
x
1
0.8
0.6
0.4
0.2
0
-0.2
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

.

(b) .


Function interpolation using 7 Chebyshev spaced nodes on [-1,1]
Newton (p(x))
Spline (S(x))
Least Squares fit g(x)
f(x)
y
x
1
0.8
0.6
0.4
0.2
0
-0.2
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

```
>> main2
Coefficients of p(x), x0...x6 =
    0.0404   -0.1089    0.4033   -1.6510   -4.9051   -6.6214   -6.7917

Coefficients of S(x), x0...x6 =
         0   -2.6243   11.9239  -19.1052   11.9239   -2.6243         0

Least  square coefficients g(x) (a,b,c respectively)=
    0.7878
   -2.6087
    1.9696

f(x)=
  Columns 1 through 10

    0.0385    0.0545    0.0826    0.1379    0.2647    0.5902    1.0000    0.5902    0.2647↙
0.1379

  Columns 11 through 13

    0.0826    0.0545    0.0385

f(x)-p(x)=
  Columns 1 through 10

    0.1020   -0.0692    0.1348    0.0915   -0.1617   -0.2405         0   -0.2405   -0.1617↙
0.0915

  Columns 11 through 13

    0.1348   -0.0692    0.1020

f(x)-S(x) =
  Columns 1 through 10

    0.0029   -0.0069    0.0348    0.0416   -0.1153   -0.1997         0   -0.1997   -0.1153↙
0.0416

  Columns 11 through 13

    0.0348   -0.0069    0.0029

f(x)-g(x)=
  Columns 1 through 10

   -0.1102    0.1284    0.0652   -0.1208   -0.2575   -0.1267    0.2122   -0.1267   -0.2575↙
-0.1208

  Columns 11 through 13

    0.0652    0.1284   -0.1102

>>
```

(c) The plots that used the Chebyshev nodes were closer to $f(x)$ since these nodes are better for interpolating for all methods. This is especially visible at both ends of the functions, though at the center (x=0) the 7 equally spaced nodes seem to interpolate the functions much better than the later. Furthermore, all three interpolations using the Chebyshev nodes are much more similar to each other as well. In general, the Chebyshev nodes form quite a good set of nodes for polynomial interpolation.

Matlab code. Main functions included as .m file on mycourses

```
function y = newtonEval(x)
%coord=0for the 7 equally spaced, 1 for chebyshev
nodes= 0;
a = Newton(nodes);
n = length(a);

if nodes==0
    coord=linspace(-1,1,7);
elseif nodes==1 %for chebyshev
    for i=0:6
        coord(i+1)=cos((2*i+1)*pi/(2*6+2));
    end
end

y = a(n);

xi = coord(1,:);

for i=n-1: -1:1
    y = a(i) + y.*(x-xi(i));
end

----------------------------------
function a=Newton(nodes)
n=7;
if nodes==0
    xs=linspace(-1,1,7);
elseif nodes==1 %for chebyshev
    for i=0:6
        xs(i+1)=cos((2*i+1)*pi/(2*6+2));
    end
end
for i=1:n
    y(i)=f(xs(i));
end

for k=1:n-1
    a(k)=y(k);
    for i=k+1:n
        y(i)=(y(i)-y(k))/(xs(i)-xs(k));
    end
end
a(n)=y(n);
```

```
----------------------------------
function g=leastSquaresEval(x)
coord=0;
p=leastSquares(coord);
g=p(1)+p(2)*x.^2+p(3)*x.^4;
----------------------------------
function c=leastSquares(coord)
%coord=0 for 7 equally spaced nodes, 1 for chebyshev nodes
n=7;
if coord==0
    x=linspace(-1,1,7);
elseif coord==1 %for chebyshev
    for i=0:6
        x(i+1)=cos((2*i+1)*pi/(14));
    end
end
for i=1:7
    y(i)=f(x(i));
end
a=zeros(n,3);
for i=1:n
    a(i,1)=x(i)^0;
    a(i,2)=(x(i))^2;
    a(i,3)=(x(i))^4;
end

c=a\transpose(y);

----------------------------------
function z=calcZ(coords)

if coords==0
    x=linspace(-1,1,7);
elseif coords==1 %for chebyshev
    for i=0:6
        x(i+1)=cos((2*i+1)*pi/(14));
    end
    x=fliplr(x);
end
n=length(x);
for i=1:n
    y(i)=f(x(i));
end

for i=1:n-1
    h(i)=x(i+1)-x(i);
    b(i)=(y(i+1)-y(i))/h(i);
end

%forward elimination
u(2)=2*(h(1) + h(2));
```

```
v(2)=6*(b(2) - b(1));

for i=3:n-1
    u(i)=2*(h(i-1)+h(i)) - ((h(i-1))^2)/u(i-1);
    v(i)=6*(b(i)-b(i-1))-(h(i-1)*v(i-1))/u(i-1);
end

%back substitution
z(n)=0;
for i=n-1:-1:2
    z(i)=(v(i)-h(i)*z(i+1))/u(i);
end
z(1)=0;
----------------------------------

function S=cubicSpline(x)
coord=0;
if coord==0
    t=linspace(-1,1,7);
elseif coord==1 %for chebyshev
    for i=0:6
        t(i+1)=cos((2*i+1)*pi/(14));
    end
    t=fliplr(t);
end
n=length(t);
z=calcZ(coord);
for i=1:n
    y(i)=f(t(i));
end

for i=1:n-1
    if x<=t(i+1)
        break;
    end
end
h = t(i+1)-t(i);
B = -(h*z(i+1)/6)-(h*(z(i))/3) + ( y(i+1) - y(i))/h;
D = ( z(i+1)-z(i))/(6*h);
S = y(i) + (x-t(i))*(B + (x-t(i))*( (z(i)/2) + ((x - t(i))*D)));
----------------------------------
function y=f(x)
y=1/(1+25*(x .^ 2));
----------------------------------
```