Weimin Gao

CSc 301 HW#3

Q1

```
% Script File: ShowSpline
close all
xvals = linspace(0,4*pi,100);
yvals = 2.*sin(xvals) + cos(xvals);
for n = 6
   x = linspace(0,4*pi,n)';
   y = 2.*sin(x) + cos(x);
   [a,b,c,d] = CubicSpline(x,y,1,1,1);
   svals = pwCEval(a,b,c,d,x,xvals);
   figure
   plot(x,y,'o')
   hold on
   plot(xvals,yvals,xvals,svals,'--')
   title(sprintf('Complete spline interpolant'))
end
for n = 6
   x = linspace(0,4*pi,n)';
   y = 2.*sin(x) + cos(x);
   [a,b,c,d] = CubicSpline(x,y,2,0,0);
   svals = pwCEval(a,b,c,d,x,xvals);
   figure
   plot(x,y,'o')
   hold on
   plot(xvals,yvals,xvals,svals,'--')
   title(sprintf('Natural spline interpolant'))
end
for n = 6
   x = linspace(0,4*pi,n)';
   y = 2.*sin(x) + cos(x);
   [a,b,c,d] = CubicSpline(x,y);
   svals = pwCEval(a,b,c,d,x,xvals);
   figure
   plot(x,y,'o')
   hold on
   plot(xvals,yvals,xvals,svals,'--')
   title(sprintf('Not-a-Knot spline interpolant'))
end
```

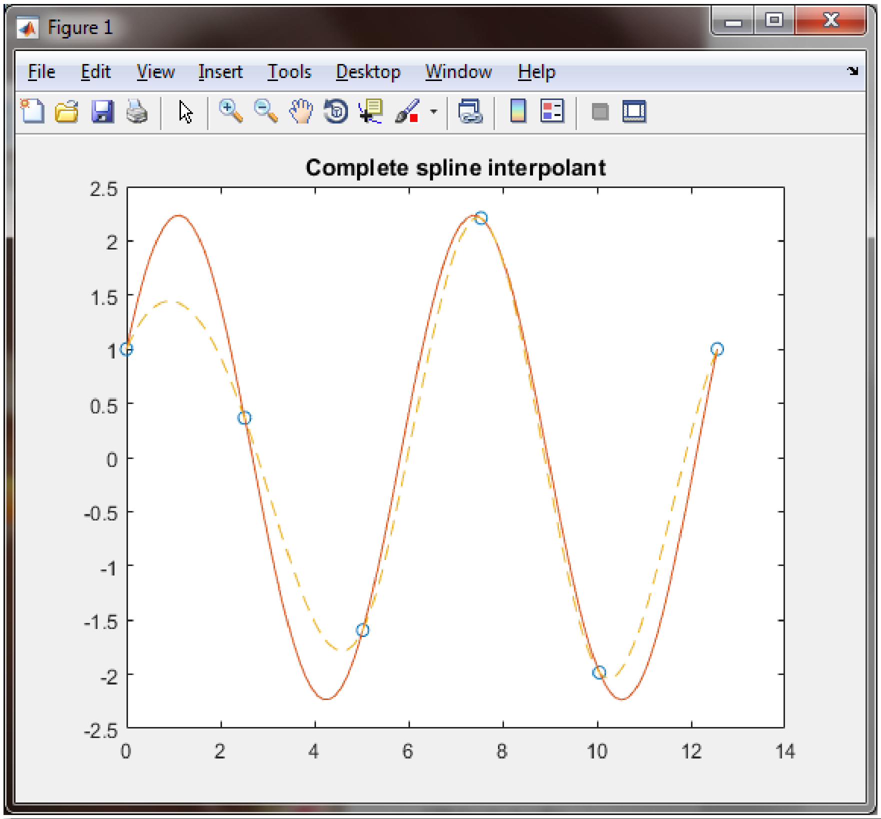


Figure 1
File Edit View Insert Tools Desktop Window Help
Complete spline interpolant
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0 2 4 6 8 10 12 14

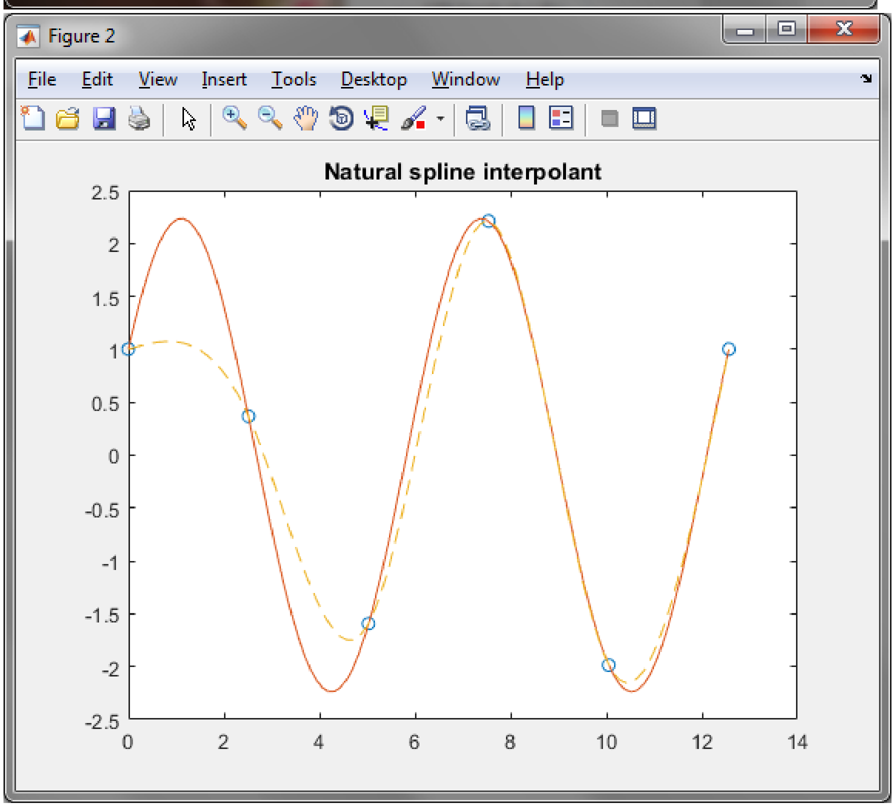
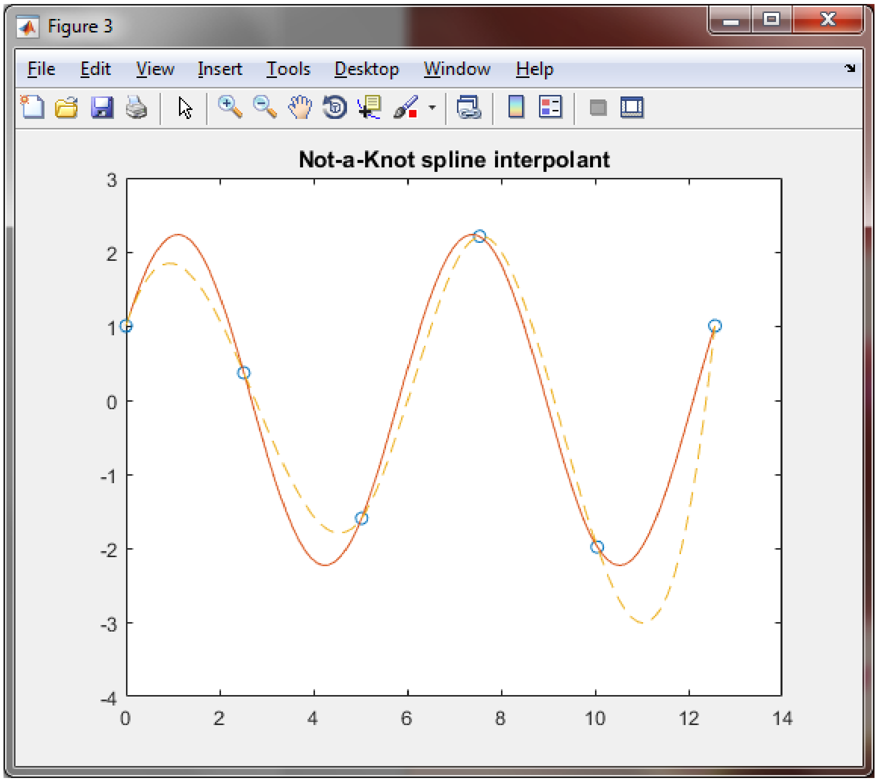


Figure 2
File Edit View Insert Tools Desktop Window Help
Natural spline interpolant
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0 2 4 6 8 10 12 14



Depended on my choice, when I compared function and spline interpolants, all of them have big gap or error. But the nature spline interpolant is more nice and smooth than other two in the end.

Q2

First, I used function **InterpN2** to find c_1, c_2, …..,c_n.

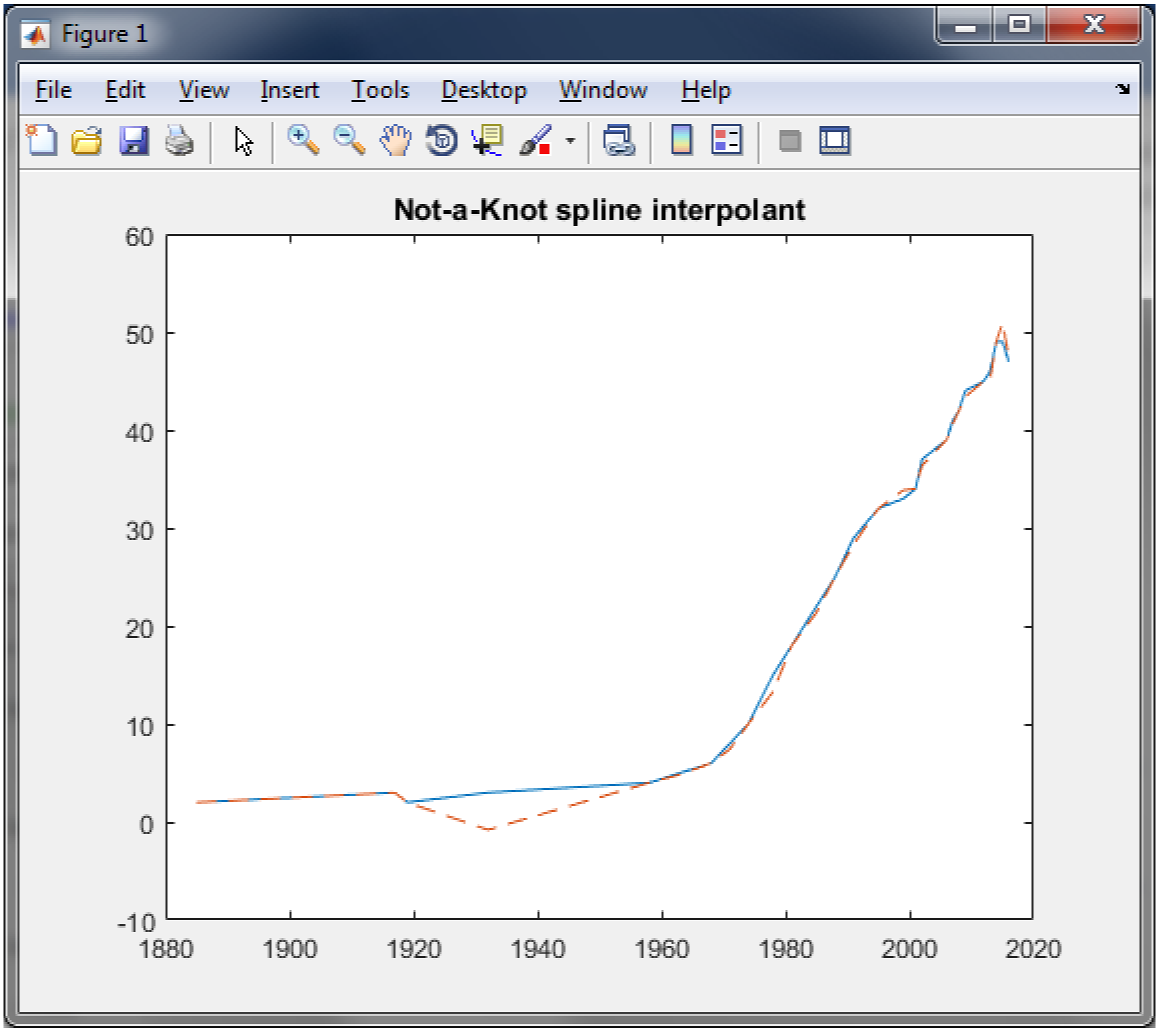
```
function c = InterpN2(x,y)
n = length(x);
for k = 1:n-1
   y(k+1:n) = (y(k+1:n)-y(k:n-1)) ./ (x(k+1:n) - x(1:n-k));
end
c = y;

%Script File
x = [1885 1917 1919 1932 1958 1963 1968 1971 1974 1978 1981 1985 1988
1991 1995 1999 2001 2002 2006 2007 2008 2009 2012 2013 2014 2015 2016];
y = [2 3 2 3 4 5 6 8 10 15 18 22 25 29 32 33 34 37 39 41 42 44 45 46 49
49 47];
c = InterpN2(x,y);
>> c

c =

  Columns 1 through 13

    2.0000    0.0313   -0.0156    0.0012   -0.0000    0.0000   -0.0000    0.0000   -0.0000    0.0000   -0.0000    0.0000   -0.0000

  Columns 14 through 26

    0.0000   -0.0000    0.0000   -0.0000    0.0000   -0.0000    0.0000   -0.0000    0.0000   -0.0000    0.0000   -0.0000    0.0000

  Column 27

   -0.0000
```

a) The Newton interpolation polynomial:

$$P(x) = 2+0.0313(x-1885)-0.0156(x-1885)(x-1917)+0.0012(x-1885)(x-1917)(x-1919)$$

b) The not-a-knot cubic spline:

```
%Script File
xvals = [1885 1917 1919 1932 1958 1963 1968 1971 1974 1978 1981 1985
1988 1991 1995 1999 2001 2002 2006 2007 2008 2009 2012 2013 2014 2015
2016];
yvals = [2 3 2 3 4 5 6 8 10 15 18 22 25 29 32 33 34 37 39 41 42 44 45
46 49 49 47];
for n = 10
   [a,b,c,d] = CubicSpline(x,y);
   svals = pwCEval(a,b,c,d,x,xvals);
   figure
   plot(xvals,yvals,xvals,svals,'--')
   title(sprintf('Not-a-Knot spline interpolant'))
end
```



The blue line shows each point of the date

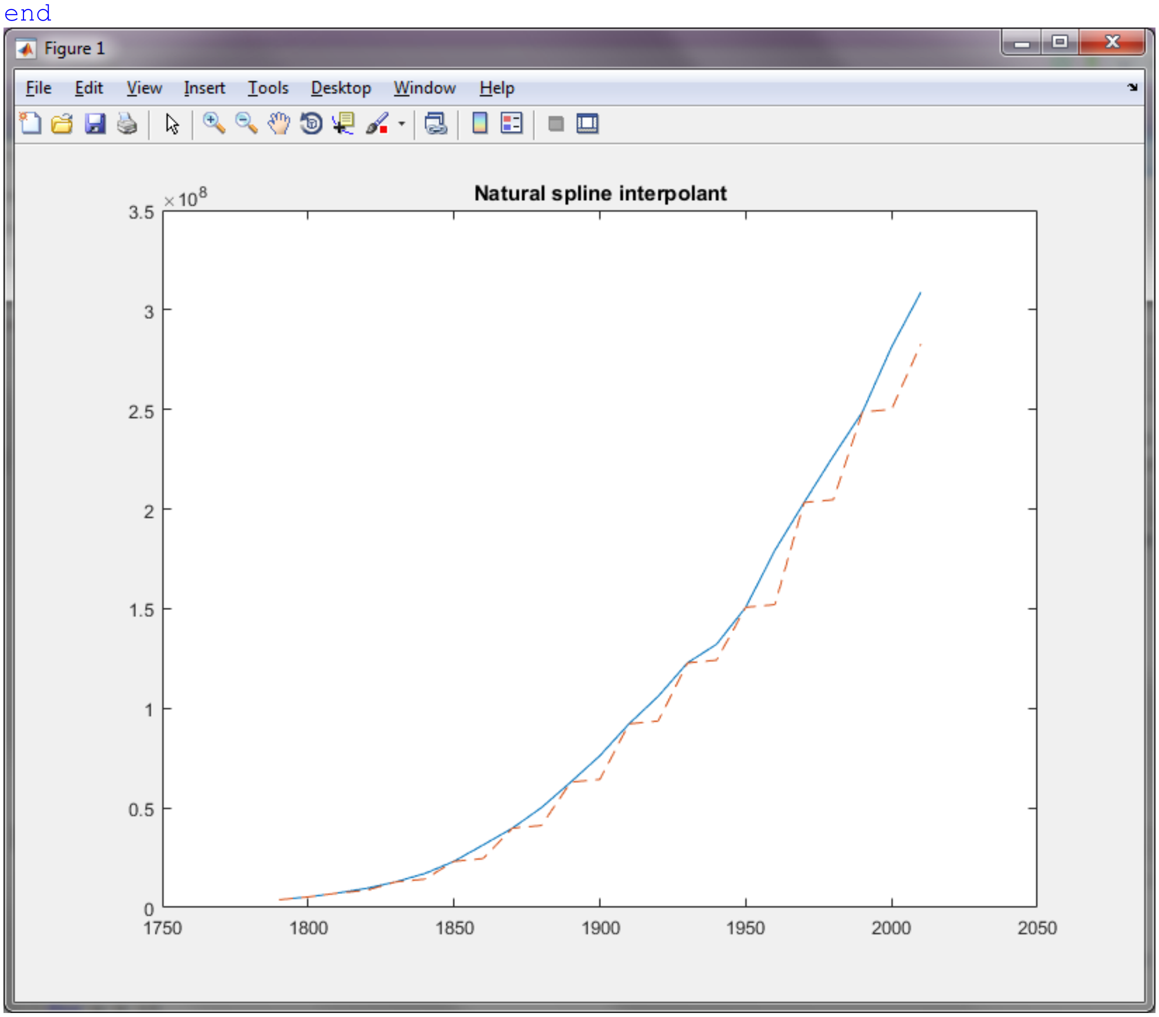
The red dash line shows the not-a-knot cubic spline for these data

c) If we put y=50 cent in the **a),** 50 = 2+0.0313(x-1885)-0.0156(x-1885)(x-1917)+0.0012(x-1885)(x-1917)(x-1919), I get **x=1952**.

But if we look the results from the **b),** we can predict that **in 2017** the cost will raise to 48 or return to 49, then **in 2018** the cost upper to 50 cents.

Q3. **a)**

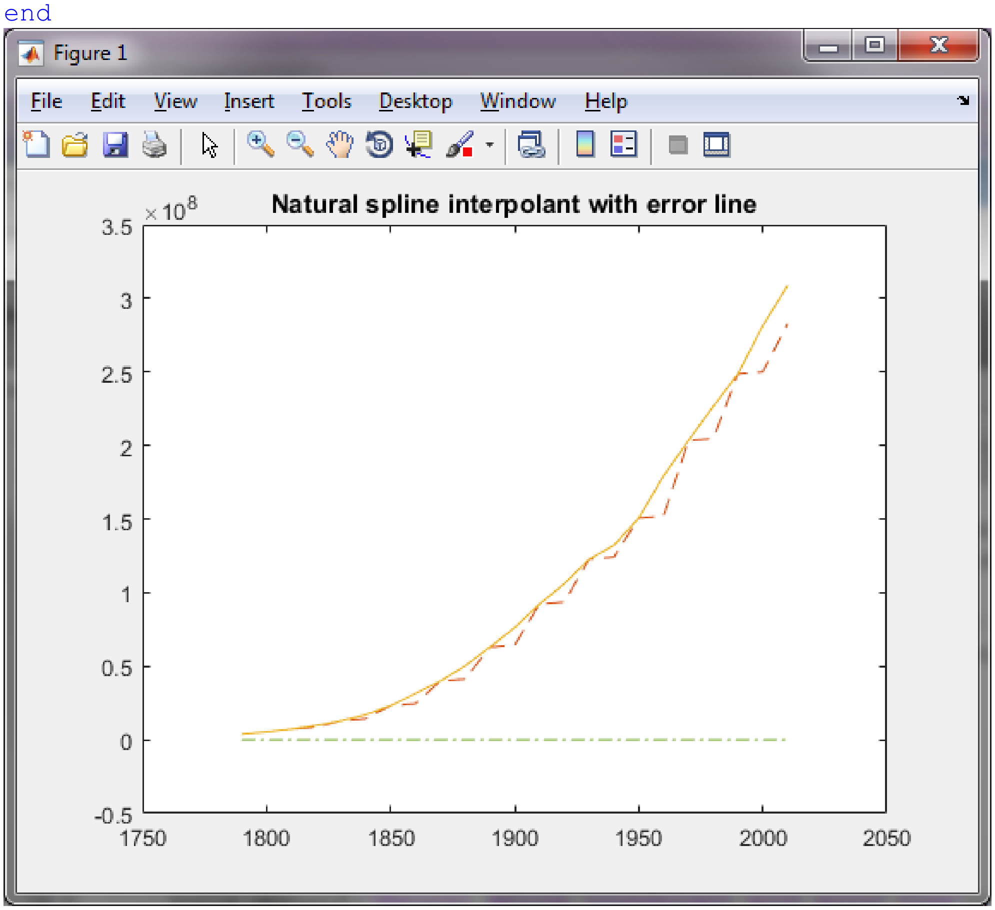
```
%Script File for Q3 a part with error
xvals = [1790 1800 1810 1820 1830 1840 1850 1860 1870 1880 1890 1900
1910 1920 1930 1940 1950 1960 1970 1980 1990 2000 2010];
yvals = [3929326 5308483 7239881 9638453 12866020 17069453 23191876
31443321 39818449 50189209 62947714 76212168 92228496 106021537
122775046 132164569 150697361 179323175 203302031 226545805 248709873
281421906 308745538];
for n = 6
   x = [1790 1800 1810 1820 1830 1840 1850 1860 1870 1880 1890 1900
1910 1920 1930 1940 1950 1960 1970 1980 1990 2000 2010];
   y = [3929326 5308483 7239881 9638453 12866020 17069453 23191876
31443321 39818449 50189209 62947714 76212168 92228496 106021537
122775046 132164569 150697361 179323175 203302031 226545805 248709873
281421906 308745538];
   [a,b,c,d] = CubicSpline(x,y,2,0,0);
   svals = pwCEval(a,b,c,d,x,xvals);
   figure
   plot(xvals,yvals,xvals,svals,'--')
   title(sprintf('Natural spline interpolant'))
end
```



The red dash line shows the the natural cubic spline.

b)

```
%Script File for Q3 b part with error
xvals = [1790 1800 1810 1820 1830 1840 1850 1860 1870 1880 1890 1900
1910 1920 1930 1940 1950 1960 1970 1980 1990 2000 2010];
yvals = [3929326 5308483 7239881 9638453 12866020 17069453 23191876
31443321 39818449 50189209 62947714 76212168 92228496 106021537
122775046 132164569 150697361 179323175 203302031 226545805 248709873
281421906 308745538];
for n = 6
   x = [1790 1800 1810 1820 1830 1840 1850 1860 1870 1880 1890 1900
1910 1920 1930 1940 1950 1960 1970 1980 1990 2000 2010];
   y = [3929326 5308483 7239881 9638453 12866020 17069453 23191876
31443321 39818449 50189209 62947714 76212168 92228496 106021537
122775046 132164569 150697361 179323175 203302031 226545805 248709873
281421906 308745538];
   [a,b,c,d] = CubicSpline(x,y,2,0,0);
   svals = pwCEval(a,b,c,d,x,xvals);
   figure
   plot(xvals,yvals,xvals,svals,'--')
   hold on
   err = (svals-yvals)./yvals;
   plot(xvals,yvals,xvals,err,'-.')
   title(sprintf('Natural spline interpolant with error line'))
end
```



The green line shows the errors for each year.