HOMEWORK 3 - CS323

LIAM DUNGAN

Problem 1.

a.
$\|Tx\| \leq \|Tx\| \|x\|$
Since, $\left\|T^k\right\| = \|T * T * T * ...T_k\|$
and we know for some A and B, $\|A * B\| \leq \|A\| * \|B\|$
then $\left\|T^k\right\| = \|T * T * T * ...T_k\| \leq \|T\| * \|T\| * \|T\| * ... \|T_k\|$
$\left\|T^k x\right\| \leq \left\|T^k\right\| \|x\| \leq \|T\| * \|T\| * \|T\| * ... \|T_k\| = \|T\|^k \|x\|$
$\left\|T^k x\right\| \leq \|T\|^k \|x\|$
b.
$A = D - L - U$
so, $(D - L - U)x = b$
then, $Dx - Lx - Ux = b$
$Dx = (L + U)x + b$
$Dx^{k+1} = (L + U)x^k + b$
$x^{k+1} = D^{-1}(L + U)x^k + b$
with $T = D^{-1}(L + U)$, then $\Longrightarrow x^{k+1} = Tx^k + c$
c.
$x\star = Tx \star +c$
$e^k = x^k - x\star$
then, $e^{k+1} = (Tx^k + c) - (Tx \star +c)$
$e^{k+1} = Tx^k - Tx\star)$
$e^{k+1} = T(x^k - x\star)$
$\Longrightarrow e^{k+1} = Te^k$, where $e^k = x^k - x\star$

Problem 2.

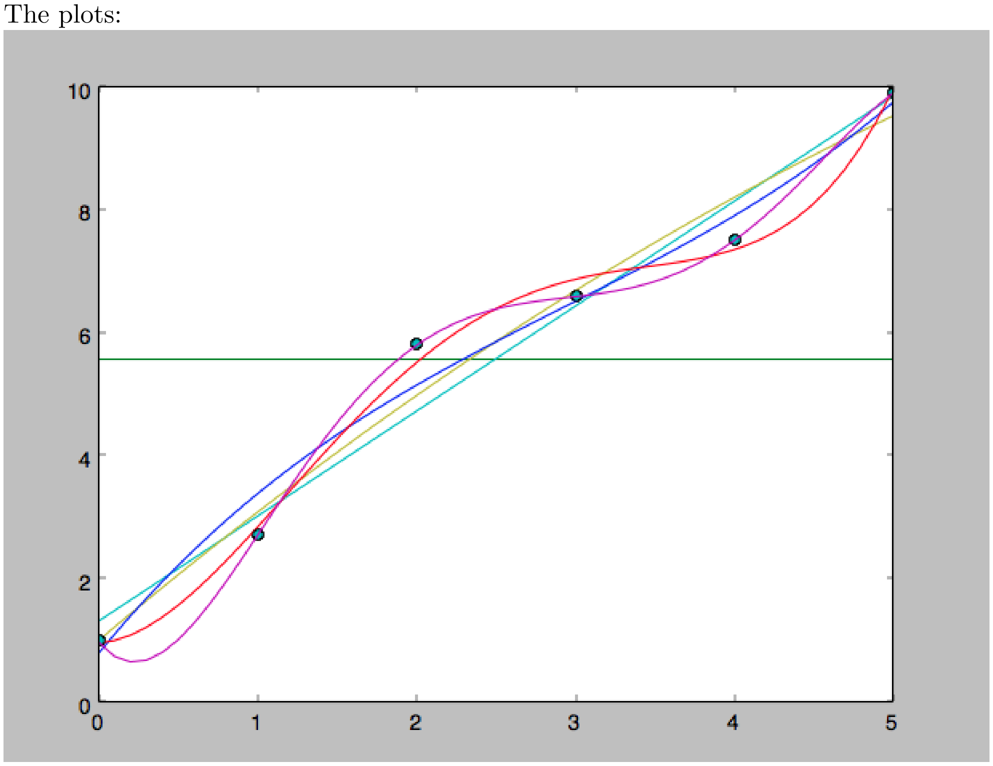
The error is reduced as the degree of each polynomial increases. The polynomial of degree n = 5 captures the trend of the data best.

The polynomials of degree n = 0,1,...,5 :

Date: Nov. 1, 2017.

```
5.583


1.706 x + 1.319

          2
-0.09464 x + 2.179 x + 1.004

        3          2
0.0713 x - 0.6294 x + 3.156 x + 0.7897

        4          3         2
0.1062 x - 0.9912 x + 2.634 x + 0.12 x + 0.9718

          5          4         3         2
-0.05917 x + 0.8458 x - 4.213 x + 8.304 x - 3.178 x + 1
```

The plots:



Code:

```python

import matplotlib.pyplot as plt
from matplotlib import pylab
import numpy as np

points = np.array([(0.0, 1.), (1., 2.7), (2., 5.8), (3, 6.6), (4.,7.5), (5.,9.9)])
x = points[:,0]
y = points[:,1]

# calculate polynomials
z0 = np.polyfit(x, y, 0)
z1 = np.polyfit(x, y, 1)
z2 = np.polyfit(x, y, 2)
z3 = np.polyfit(x, y, 3)
z4 = np.polyfit(x, y, 4)
z5 = np.polyfit(x, y, 5)
f0 = np.poly1d(z0)
f1 = np.poly1d(z1)
f2 = np.poly1d(z2)
f3 = np.poly1d(z3)
f4 = np.poly1d(z4)
f5 = np.poly1d(z5)

#print polynomials
print f0, "\n\n", f1, "\n\n", f2, "\n\n", f3, "\n\n", f4, "\n\n", f5, "\n"

#new x's and y's
x1_new = np.linspace(x[0], x[-1], 50)
y0_new = f0(x1_new)
plt.plot(x,y,'o', x1_new, y0_new)

y1_new = f1(x1_new)
plt.plot(x,y,'o', x1_new, y1_new)

y2_new = f2(x1_new)
plt.plot(x,y,'o', x1_new, y2_new)

y3_new = f3(x1_new)
plt.plot(x,y,'o', x1_new, y3_new)

y4_new = f4(x1_new)
plt.plot(x,y,'o', x1_new, y4_new)

y5_new = f5(x1_new)
plt.plot(x,y,'o', x1_new, y5_new)

plt.show()
```

Problem 3.

a.

x1 = 1,x2 =1

b.

x1 = 7.00888731, x2 = -8.39566299

c.

The results are very different given the slightly modified b vector. A possible reason for this is because the matrix is ill-conditioned, meaning the condition number is very large. The condition number and resulting x values can be seen in the output below.

```
L1 cond(A^T*A):  1631924.81894
L2 cond(A^T*A):  1204591.14638

For b =  [ 0.26  0.28  3.31]     x =  [ 1.  1.]
For b =  [ 0.27  0.25  3.33]     x =  [ 7.00888731 -8.39566299]
```

Code:

```
import numpy as np

A = np.array([[0.16,0.1],[0.17,0.11],[2.02,1.29]])
b = np.array([0.26,0.28,3.31])
b2 = np.array([0.27,0.25,3.33])

#newA = At*A
# ??? ValueError: operands could not be broadcast together with s

newA = np.dot(np.transpose(A) , A)
print "\nL1 cond(A^T*A): ",np.linalg.cond(newA,1)
print "L2 cond(A^T*A): ", np.linalg.cond(newA)

print ""
newb = np.dot(np.transpose(A) , b)
x = np.linalg.solve(newA,newb)
print "For b = ",b,"\t x = ", x

newb2 = np.dot(np.transpose(A) , b2)
x2 = np.linalg.solve(newA,newb2)
print "For b = ",b2,"\t x = ", x2
```

Problem 4.

a.

In order to show that the matrix is sigular, we will show that it's determinant is equal to 0.

$$\mathrm{A} = \begin{bmatrix} 0.1 & 0.2 & 0.3 \\ 0.4 & 0.5 & 0.6 \\ 0.7 & 0.8 & 0.9 \end{bmatrix}$$

$$\Longrightarrow \det(\mathrm{A}) = \begin{vmatrix} 0.1 & 0.2 & 0.3 \\ 0.4 & 0.5 & 0.6 \\ 0.7 & 0.8 & 0.9 \end{vmatrix} = 0.1 \begin{vmatrix} 0.5 & 0.6 \\ 0.8 & 0.9 \end{vmatrix} - 0.2 \begin{vmatrix} 0.4 & 0.6 \\ 0.7 & 0.9 \end{vmatrix} + 0.3 \begin{vmatrix} 0.4 & 0.5 \\ 0.7 & 0.8 \end{vmatrix}$$

$$= 0.1(0.45 - 0.48) - 0.2(0.36 - 0.42) + 0.3(0.32 - 0.35)$$

$$= 0$$

If $\mathrm{b} = \begin{bmatrix} 0.1 \\ 0.3 \\ 0.5 \end{bmatrix}$, the set of solutions to $\mathrm{Ax} = \mathrm{b}$ is $x = \left\{ x_1 - x_3 = \frac{1}{3}, \quad x_2 + 2x_3 = \frac{1}{3} \right.$

This system is dependent so there is infinitely many solutions.

b.

$$\begin{bmatrix} \frac{1}{10} & \frac{2}{10} & \frac{3}{10} \\ \frac{4}{10} & \frac{5}{10} & \frac{6}{10} \\ \frac{7}{10} & \frac{8}{10} & \frac{9}{10} \end{bmatrix} \Rightarrow \begin{bmatrix} \frac{7}{10} & \frac{8}{10} & \frac{9}{10} \\ \frac{4}{10} & \frac{5}{10} & \frac{6}{10} \\ \frac{1}{10} & \frac{2}{10} & \frac{3}{10} \end{bmatrix} \Rightarrow \begin{bmatrix} \frac{7}{10} & \frac{8}{10} & \frac{9}{10} \\ 0 & \frac{-3}{40} & \frac{-3}{20} \\ 0 & \frac{-3}{5} & \frac{-6}{5} \end{bmatrix} \Rightarrow \begin{bmatrix} \frac{7}{10} & \frac{8}{10} & \frac{9}{10} \\ 0 & \frac{-3}{5} & \frac{-6}{5} \\ 0 & \frac{-3}{40} & \frac{-3}{20} \end{bmatrix} \Rightarrow$$

$$\begin{bmatrix} \frac{7}{10} & \frac{8}{10} & \frac{9}{10} \\ 0 & \frac{-3}{5} & \frac{-6}{5} \\ 0 & 0 & 0 \end{bmatrix}$$

At this point in the process, when the entire row becomes zeros, the gaussian elimination fails.

c.

The computed solution becomes `[ 0.08333333  0.83333333 -0.25       ]`. Although this is a solution to the system, it is only one of an infinite amount that we found in part a.

The estimated condition number of this matrix is `8.64691128455e+16`