AMSC/CMSC460 Section 2.

MATLAB Exercise 3. 2011.04.14

Least square estimation based on QR factorization **r= b** -**Ax** can be written in 4 steps

1. compute QR factorization $\mathbf{A} = \widehat{\mathbf{Q}}\widehat{\mathbf{R}}$
2. obtain $\mathbf{c} = \widehat{\mathbf{Q}}^{\boldsymbol{T}}\mathbf{b}$
3. solve $\widehat{\mathbf{R}}\mathrm{x} = \mathbf{c}$ for x
4. compute residual r

where **b** & **r** is in R^L, **x** is in R^N & **A** is in R^{LxN}

1. Write a MATLAB function code for the least square estimation

[x,r]=LSE_QR(A,b)

You can make use of the MATLAB function codes you have developed for this class so far.

2. To check whether your code is properly working or not, generate a synthetic data set by

b(l)=a0+a1*t(l)+epsilon(l)

for l=1,..,L=100, where

```
a0=3;
a1=2;
e=0.3;
t=linspace(0,1,L)';
epsilon=e*randn(L,1);
```

Save t and b in a file.

a) What is randn?
b) Why save the data

3. Write matlab code that

a) read your saved data;

b) use your MATLAB code to compute the coefficient c0 and c1 of the linear function

q(t)=c0+c1*t

that fits your data in the least square sense.

c) Compare (a0,a1) vs (c0,c1) as well as r vs expected residual (do you know how to get the expectation?)

d) plot in the same figure

- t vs q(t) as a blue line
- t(l) vs b(l) as red dots for l=1,…,L
- add x and y label
- add text to show (a0,a1,e) and (c0, c1)