Assignment 4

MATH 7502 - Semsester 2, 2018

Mathematics for Data Science 2

Created by Zhihao Qiao, Maria Kleshnina and Yoni Nazarathy

Question 1 - Carry over of question 10 from previous assignmet:

Solve Question 12.13 from [VMLS], page 241.

Question 2 - Carry over of question 11 from previous assignment:

You are given a channel impulse response, the n-vector c. Your job is to find an equalizer impulse response, the n-vector h that minimizes $||h * c - e_1||^2$. You can assume $c_1 \neq 0$.

(a) Explain how to find h, Apply your method to find the equalizer h for the channel $c = (1.0, 0.7, -0.3, -0.1, 0.05)$.

(b) Plot c, h, and $h * c$.

Question 3

Say that you observe data points $(y_1, x_1, m_1), \ldots, (y_n, x_n, m_1)$. Assume that y and x are real valued and that m is binary valued. Say you wish to use least squares to fit a function,

$$y(x) = \beta_0 + \beta_1 x + \beta_2 m_1$$

(i) Describe the A matrix for the problem $\min_\beta ||A\beta - y||$.

(ii) Consider the data values below. Plot the data values on the x-y plane using different colors for m=0 and m=1

(iii) Fit the model with for the data and plot the line(s) of best fit.

In [27]:

```
xVals = [1, 6, 11, 16, 21, 26, 31, 36, 41, 46, 51, 56, 61, 66, 71]
mVals = [1, 1, 0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1]
yVals = [215.132, 259.396, 99.2338, 324.469, 253.776, 450.759,
        305.793, 472.493, 258.894, 555.746, 335.42,
        636.734, 624.769, 435.191, 638.885];
```

Question 4

Carry out 12.14 from [VMLS], page 242 dealing with recursive least squares.

Question 5

Carry out 13.17 from [VMLS], pages 282-283.

Question 6

Carry out 13.19 from [VMLS], page 283.

Question 7

Carry out 14.17 from [VMLS], page 306.

Question 8

Carry out 15.4 from [VMLS], page 334.

Question 9

Carry out 15.11 from [VMLS], page 337.

Question 10

Carry out 16.5 from [VMLS], page 352.

Question 11

The code below considers a non-small least squares problem. With A of dimension $40,000 \times 1000$. It is constructed by selecting A and β randomly with a fixed seed. Plot the running time of this code for increasing n and p (e.g. keep $p/n = 1/8$ and increase (or decrease n).) Investigate the behaviour of the running time as the ratio of p and n changes.

In [24]:

```
using Distributions
srand(1988)
n = 40000
p = 500
A = round(50*rand(n,p))
beta = 5*rand(p)
y = A*beta + 2000*rand(Normal(),n);
betaHat = A \ y
maximum(abs(betaHat - beta))
```

Out[24]:

```
2.2697565816429908
```

Question 12

Implement a gradient descent algorithm for the the data of the previous question. Say that you wish to run it for a fixed number of iterations (with a specified fixed learning rate and fixed initial guess). Plot the maximum absolute value error rate as n grows.