Assignment 1

February 16, 2016

1 OPTIMIZATION AND LINEAR REGRESSION

Problem 1 Let $f(x) = x_1^2(x_1 - 2) + x_2 log(x_2)$.

- Find stationary points of f(x) (2pts).
- Determine if those stationary points are local maxima, local minimum, or a saddle point. Support your answers. (4pts)

Problem 2 Let $f(x_1, x_2) = \frac{1}{2}(x_1)^2 + \frac{3}{2}(x_2)^2$. Using step length $\alpha = 0.5$, with starting point $x^0 = [3, 1]$, write the first 2 iterations of the gradient descent method for minimizing function f. For each iteration, show your current function value, current point estimate, search direction, next point in the iteration. (8pts)
Problem 3 Show that in the case of simple linear regression, the least squares line always passess through the point $(\bar{x}, \bar{y})$ (4pts).
Problem 4 The R^2 statistics is defined as: $R^2 = 1 - \frac{RSS}{TSS} = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$. Prove that in the simple linear regression with Y is the response, X is the predictor , R^2 is actually equal to the square of the correlation between X and Y. (6pts)
Problem 5 Show that if A is a symmetric matrix, we have: $x^T Ay = y^T Ax$. (4pts)

2 MATLAB PROGRAMMING

Problem 1 Write your own function that implement simple linear regression, call it my_SLR.m.

- The function input is vector of predictor X and vector of response Y. (2pts)

- Function output a bector β: $[\beta_0; \beta_1]$ where β_0 is the coefficient of intercept and β_1 is the coefficient of predictor X. You can use the formula for β_0 and β_1 given in the lecture slides. (6pts)
- The function also outputs R^2 defined in problem 5 (and in the textbook). (2pts)

Included in the assigment there is a data file name: problem_1.mat. The data file has two input vectors X and Y that you can use to test your implementation. In order to load the data file into Matlab, you can use the syntax:load('problem_1.mat').

Problem 2 Write your own function that implement multiple linear regression, call it my_MLR.m.

- The function input is matrix of predictors X and vector of response Y. Matrix X has already been augmented with a column of 1's so you don't have to do that.(2pts)
- Function output a bector β which is the vector of least square coefficient estimates. You can use the formula for β given in the lecture slides. You can use the Matlab function inv() that returns the inverse of a matrix. (6pts)
- The function also outputs R^2 defined in problem 5 (and in the textbook). (2pts)

Included in the assigment there is a data file name: problem_2.mat. The data file has two input vectors X and Y that you can use to test your implementation. In order to load the data file into Matlab, you can use the syntax:load('problem_2.mat').

Problem 3 In this problem you will create some simulated data and fit simple linear regression models to it.

- Using the normrnd() function to create a vector x containing 100 observations drawn from a normal distribution N(0,1). You can use the syntax: $normrnd(\mu, \sigma, [m, n])$ where μ and σ is the mean and variance of the distribution.[m,n] is the size of the vector that you want to generate. (2pts)
- Use the same function to create a vector of noise, ϵ, containings 100 observations drawn from a N(0,0.25) distribution. (2pts)
- Use X and ϵ to create a vector y according to the model:(2pts)

$$Y = -1 + 2X + \epsilon.$$

- Create a scatterplot displaying the relationship between X and Y. Comment on what you see. You can use the command scatter() in Matlab. (2pts)
- Fit a least squares linear model to predict Y using X. You can use the command fitlm() in matlab. How do the estimated values $\hat{\beta}_0, \hat{\beta}_1$ compared to the β_0 and β_1. (2pts)
- Is there a relationship between the predictor X and the response Y? Support your answer. (2pts)