CS299 Machine Learning: Assignment #1

Due on Thursday, March 14th, 2018

Andrew Ng 6:30 am

Bryan Zhang

Contents

Problem 1

1(a)

$H = \nabla \cdot \nabla^T J(\theta)$
So $z^T Hz = z^T(\nabla \cdot \nabla^T \cdot J(\theta))z = (z^T \cdot \nabla)^2 J(\theta) = (z^T \nabla^T \cdot J(\theta))^2 \cdot \frac{1}{J(\theta)}$.
Since the first factor of $z^T Hz$ is a square term which is definitely bigger than 0, So we focus on the factor $\frac{1}{J(\theta)}$. Because $J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} \log(h_\theta(y^{(i)}x^{(i)})$ and the hypopthesis function is a signoid function ranging from (0,1),the log of the hypothesis funcition must be negative and thus the coost function must be postive. So does the $z^T Hz$.
q.e.d.

1(b)

The optimized θ is $[-2.62042271649454, 0.760346235045246, 1.17193037252339]$.

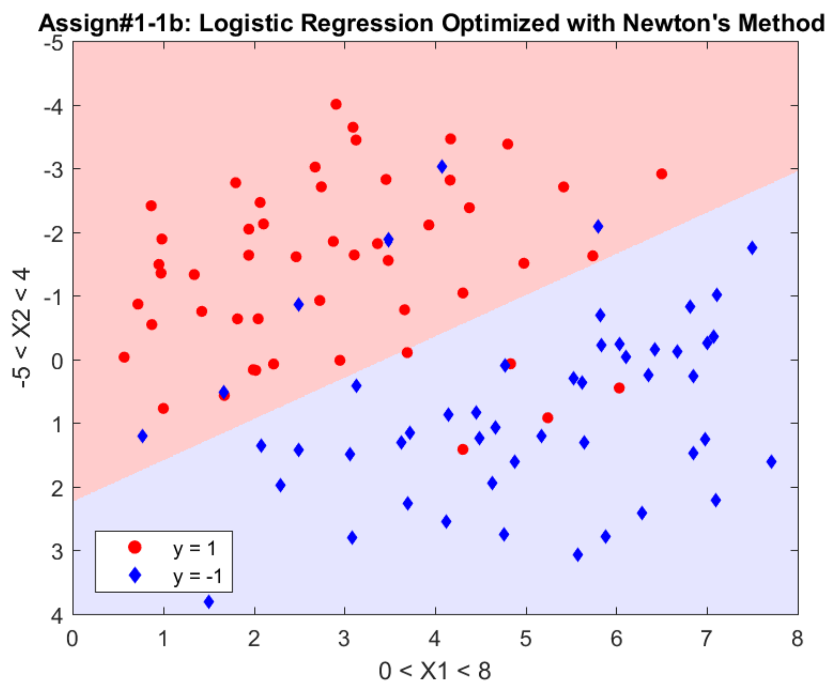
```
close all; clear; clc;

% the rows of X is input variables
% the rows of Y is the response variables
fileIDX = fopen('logistic_x.txt', 'r');
sizeX = [2 Inf];
formatSpec = '%f';
X = fscanf(fileIDX, formatSpec, sizeX).';
% append the intercept term
X = [ones(size(X, 1), 1) X];

fileIDY  = fopen('logistic_y.txt', 'r');
Y = fscanf(fileIDY, formatSpec);

%plot the x and y
% the sub dataset for respective y is 1
Xp = X(1:50 ,:);
%the other half of sub dataset
Xn = X(51:size(X, 1) ,:);
sz = 25;
x1range = [0 8];
x2range = [-5 4];



%the implementation of the Newton's Method for logistic regression
%serveral instance variables
THETA_INITIAL = zeros(1, size(X, 2));
ERROR_MARGINS = 0.00001;
%the size of sample space
m = size(Y, 1);




%the sigmoid function
sigmoid =  @(z)1./(1 + exp(-z));
%the cost function
```

```
J = @(theta) 1 / m * sum(log(sigmoid(Y.*( X * theta'))));

thetaOptimized = getTheta(J, THETA_INITIAL,ERROR_MARGINS);
%the hypthesis funciton
h = @(X) sigmoid(X * thetaOptimized');

%plot the decision boundary
% step size for the accuracy of the boundary curve
inc = 0.01;

% generate grid coordinates
[x1, x2] = meshgrid(x1range(1):inc:x1range(2), x2range(1):inc:x2range(2));
imageSize = size(x1);

x1x2 = [x1(:) x2(:)]; % make the (x1, x2) pairs as row vectors

hypothesis = zeros(length(x1x2), 1);
for i = 1:length(x1x2)
    Xhypo= [1 x1x2(i,:)];
    htemp = h(Xhypo);
    if htemp > 0.5
        hypothesis(i) = 1;
    else
        hypothesis(i) = 0;
    end
end
%reshap the hypothesis to be positioned on each grid point
decisionMap = reshape(hypothesis, imageSize);

% plot the decision boundary
figure
imagesc(x1range, x2range, decisionMap);
hold on;
cmap = [1 0.8 0.8; 0.9 0.9 1];
colormap(cmap);


scatter(Xp(:,2), Xp(:, 3), sz, 'red', 'filled');
hold on;
scatter(Xn(:,2), Xn(:, 3), sz, 'blue', 'filled', 'd');
hold on;
title("Assign#1-1b: Logistic Regression Optimized with Newton's Method ");
xlim(x1range); ylim(x2range);
xlabel('0 < X1 < 8');
ylabel('-5 < X2 < 4');
legend('y = 1','y = -1', 'Location', 'Southwest')
hold on;

%save the image
 saveas(gcf,'1b.png')

%disp tests
disp(sum(sigmoid([1 2 3])));
```

```
disp(J([0 0 0]));
disp(getGradient(J, [0 0 0]));
disp(getHessian(J, [0 0 0]));
disp(getTheta(J, [0 0 0], 0.00001));

% get the optimized theta
function theta = getTheta(costfunc, thetaIni, errorMargins)
%% costfunc is the cost funciton for the logistic regression
%% thetaIni is the start popint to search for the optimized theta
%% return the optimized theta which can make the gradient down to zero
%% thus the cost function down to the minimal
theta = thetaIni;
grad = getGradient(costfunc, theta);
while norm(grad) > errorMargins
    grad = getGradient(costfunc, theta);
    H = getHessian(costfunc, theta);
    disp('H:'); disp(H);
    disp('grad:'); disp(grad);
    theta = theta - grad / H;
    disp('theta'); disp(theta);
end
end

%get the hessian
function H = getHessian(f, x)
%% f is a function
%% x is a input varibale
%% return the hessian for the function at x
gx = getGradient(f, x);
H = zeros(size(x, 2));
h = 0.00001;

%iterate over al indexes in x
for i = 1: size(x, 2)
    oldValues = x(i);
    x(i) = oldValues + h;
    gxh = getGradient(f, x); %get the grad f(x + h)
    x(i) = oldValues; % restore to previous value

    %compute the second partial derative
    H(:,i) = (gxh - gx)./ h;
    %iterate over to the next variable
end
end


% get the gradient
function grad = getGradient(f, x)
%% f is a function
%% x is a input varibale
%% return the gradient for the function at x

fx = f(x);
```

```
grad = zeros(size(x));
h = 0.00001;

%iterate  over all indexes in x
for i = 1:size(x, 2)
oldValues = x(i);
x(i) = oldValues + h; %increment by h
fxh  = f(x); % evaluate f(x + h)
x(i) = oldValues;%restore to the previous value for x(i)

%compute the partial derative
grad(i) = (fxh - fx) / h; %the slop
%iterate to the next index
end
end
```

1(c)



Problem 2

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Problem 2 continued on next page. . .

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.