EEC 289Q Data Analytics for Computer Engineers Homework 2

Ahmed Mahmoud

April, 29th 2018

Logistic Regression: The following code show the implementation of the logistic regression function

```
function [f,g] = logistic_regression(theta, X,y)
    m=size(X,2);
    n = size(X,1);
    f = 0;
    g = zeros(size(theta));

%%% YOUR CODE HERE %%%
    h = sigmoid(theta'*X);
    for i=1:m
         f = f - (y(i)*log(h(i)) + (1- y(i))*log(1-h(i)));
    end
    for i=1:m
        g = g + X(:,i)*(h(i)- y(i));
    end
end
```

Using this code, we were able to achieve training accuracy of $100\%$and test accuracy of 100% while the optimization took 5.258526 seconds.

The following shows the vectorized version of the same implementation which decreased the optimization time to 2.519300 seconds.

```
function [f,g] = logistic_regression_vec(theta, X,y)
    m=size(X,2);.
    f = 0;
    g = zeros(size(theta));

    %%% YOUR CODE HERE %%%
    h = sigmoid(theta'*X);
    f = -sum(y.*log(h) + (1.- y).*(log(1.-h)));
    g= X*(h - y)';
end
```

Linear Regression: The following code show the initial implementation of the linear regression method with which the optimization took 0.017464 seconds

```
function [f,g] = linear_regression(theta, X,y)
    m=size(X,2);
    n=size(X,1);
    f=0;
    g=zeros(size(theta));
%%% YOUR CODE HERE %%%
    err=theta'*X-y;
    for i=1:m
        f = f + 0.5*err(i)*err(i);
    end
    for i=1:n
        g(i) = sum(X(i,:).*err);
    end
end
```

The following is the vectorized version of the linear regression. With this code the optimization took 0.014477 seconds. The different is small between the vectorized and initial implementation since the number of parameters is small i.e., 14.

```
function [f,g] = linear_regression_vec(theta, X,y)
    m=size(X,2);
    f = 0;
    g = zeros(size(theta));
    %%% YOUR CODE HERE %%%
    err=theta'*X-y;
    f=1/2*err*err';
    g=X*err';
end
```

	linear regression	linear regression vec	logistic regression	logistic regression vec
Test#1	3.97861e	1.39289e-10		
Test#1	4.40389e	4.15336e-10		
Test#3	1.76783e	8.01214e-10		
Test#4	3.97861e	4.88868e-10		
Test#5	2.07973e	8.01214e-10		
Test#6	3.97861e	4.15336e-10		
Test#7	1.15902e	4.15336e-10		
Test#8	2.07973e	2.30699e-10		
Test#9	1.50102e	7.99005e-11		
Test#10	2.83986e	7.99005e-11		

Figure 1: Metric table for the four processors

Gradient Testing