Linear Regression

LBYCP29 – Laboratory 1

Gervin Guevarra, Allen Koizumi, Nicholas Moreno, Charleston Uy

Department of Electronics and Communications Engineering

Gokongwei College of Engineering, De La Salle University

Manila, Philippines

*Abstract*—This laboratory report presents the implementation of linear regression in a scattered data. Using statistical methods, a trend line can be generated to represent, as well as to predict, these values.

Keywords—linear regression, data,

# Introduction

Linear regression is a statistical technique wherein a scatter plot or observed data is attempted to be modeled as a linear relationship. A trend line of the form (1),

*y = mx+b* 

where *y* is the dependent variable, *x* is the explanatory variable, m is the slope of the line and b is the y-intercept.

Linear regression is useful in determining the simplest representation of the data trend and can also be used in predicting values beyond the initial range of the observed data. Because of its simplicity, linear regression is often used in simple computer prediction processes, which can be loosely considered as part of artificial intelligence. The linear regression model can be generated using the following equations,

(2)

(3)

# Objectives

The experiment aims to achieve the following objectives

* To import raw data to Octave and show the data in a figure;
* To implement a linear regression algorithm that would produce a desirable trend line based on the given data by using gradient descent; and
* To plot the cost function as a 3D surface plot.

# Data and Results

The plot shown in Figure 1, are the two sets of data imported to octave. The data is plotted against a Height vs Age graph; which are the y and x axis respectively.

By using gradient descent, a linear regression model is implemented. A desireable trendline is a line that is passing through the middle of the data sets. And from the plot shown in Figure 2, the group achieved this desirable trendline. In this case, the algorithm implemented by the group provided an output similar to the desirable output.

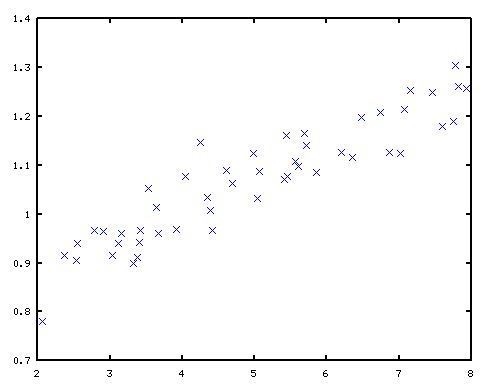


Figure 1. Training set plot.

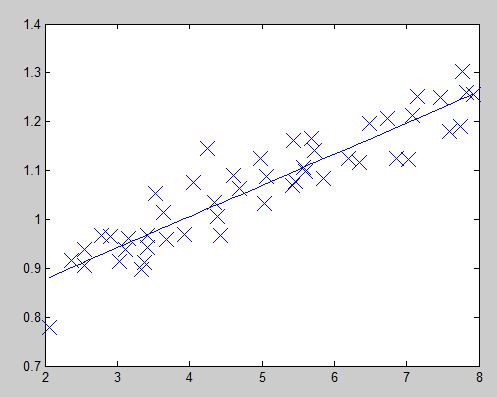
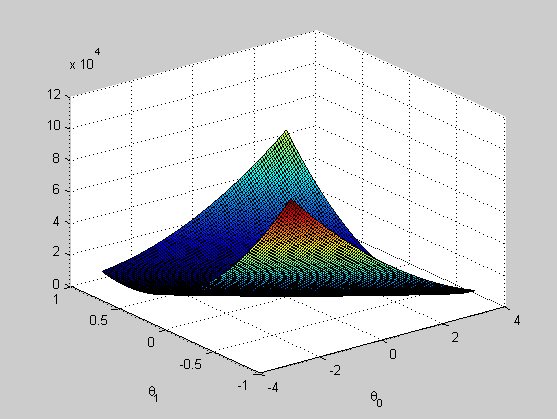


Figure 2. Generated trendline imposed on training set data.

Using the Gradient Descent Update Rule, the group was able to optimize the scattered data from the imported set of data. Since the objective of the gradient descent is to find the local minimum of a given set, the implemented an Ocaave/Matlab algorithm that would satisfy the set.

Figure 3. 3D Cost Function Plot



A cost function is a function that tries to estimate the minimum cost of producing a number of n results. In this case, cost function is plotted in a 3D figure. This is possible because the data the group is dealing with is only low dimensional in nature. The 3D plot would also allow the group to understand more about linear regression.

# Analysis and Conclusion

In machine learning, linear regression is used to calculate the output for a new data based on previous data set. After calculating the new data, a trendline is seen and the plot has been linearized. Through the use of octave, which is the free version of matlab, using gradient descent can produce a linear regression. Gradient descent starts with a preliminary set of parameter values and iteratively changes toward a set of parameter values that reduce the function. This iterative minimization is attained using calculus, moving along the negative direction of the function gradient.

# Questions

1. What is the relationship between this 3D surface and the value of θ0 and θ1 that your implementation of gradient descent had found?

The 3D plot shows the values of θ0 and θ1 before converging.

1. Provide your own data with at least 50 elements and apply the procedures performed in this experiment.

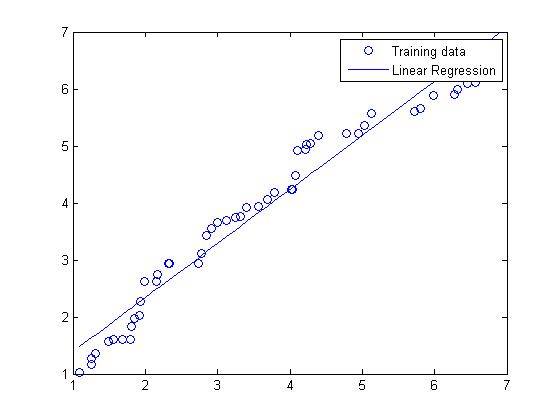


Figure 4. Linear regression applied to another data.

# Bibliography

|  |  |
| --- | --- |
| [1] | A. Ng, "Multivariate Linear Regression," 2012. [Online].  Available: http://openclassroom.stanford.edu/MainFolder/DocumentPage.php?course=  MachineLearning&doc=exercises%2Fex3%2Fex3.html. [Accessed 9 September 2015]. |
| [2] | M. Nedrich, "An Introduction to Gradient Descent and Linear Regression,"  2014. [Online]. Available: http://spin.atomicobject.com/2014/06/24/gradient-descent-  linear-regression/.  [Accessed 9 September 2015]. |
| [3] | C.-H. Chen, "Cost Functions," 2007. [Online]. Available: http://ocw.mit.edu  /courses/economics/14-01-principles-of-microeconomics-fall-2007/lecture-notes/14\_01\_lec13.pdf.  [Accessed 9 September 2015]. |

##### Appendix

Linear Regression.m

function [ theta0,theta1] = Linear\_Regression( x,y )

m=length(y);

theta=zeros(1,2);

theta2=zeros(1,2);

h1=zeros(1,2);

h=zeros(1,2);

x1=[ones(m,1),x];

for j=1:1500

h=0;

h1=0;

for i=1:m

h=h1+((x1(i,:)\*theta')-y(i))\*x1(i,:);

h1=h;

end

theta=theta2-(0.07/m)\*h;

theta2=theta;

end

plot(x,y,'o');

hold on;

plot(x1(:,2),x1\*(theta'),'-');

legend('Training data', 'Linear Regression');

theta0=theta(1);

theta1=theta(2);

end

Gradient Descent.m

function [ J] = Gradient\_Descent( x,y )

J\_vals=zeros(100,100);

m=length(y);

x1=[ones(m,1),x];

h1=zeros(1,2);

h=zeros(1,2);

theta0\_vals=linspace(-3,3,100);

theta1\_vals=linspace(-1,1,100);

for i=1:length(theta0\_vals)

for j=1:length(theta1\_vals)

t=[theta0\_vals(i);theta1\_vals(j)];

h1=0;

h=0;

for l=1:m

h=h1+((x1(l,:)\*t)-y(l))^2;

h1=h;

end

J\_vals(i,j)=(1/2\*m)\*h;

end

end

J\_vals=J\_vals';

figure;

surf(theta0\_vals,theta1\_vals, J\_vals)

xlabel('\theta\_0');ylabel('\theta\_1')

end