Multivariate Linear Regression

LBYCP29 – Laboratory 2

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 multivariate linear regression in a scattered data consisting of three variables. Using statistical methods, a plot 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)

Where *Ɵ0*is the y-intercept, *Ɵ1* is the slope and α

Linear regression can also be applied to data involving 3 or more variables. Such condition is called *multivariate*, hence the *Multivariate Linear Regression*.

Most of the procedures used in a univariate (single variable) linear regression, such as Equations 1 and 2, can be applied to multivariate liner regression. One significant difference is that the learning rate α must be picked using the cost function *J(Ɵ)* which can be written in vectorized form as shown in Equation 5,

(4)

(5)

# Objectives

The experiment aims to achieve the following objectives

* To investigate multivariate linear regression using gradient descent and the normal equations;
* To examine the relationship between the cost function J(Ɵ), the convergence of gradient descent, and the learning rate α.

# Data and Results

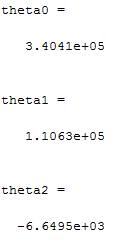
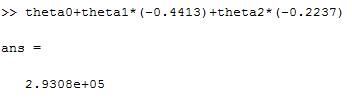


Figure 1. House price calculation and theta values using Gradient Descent

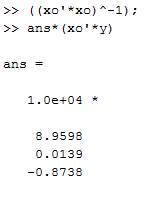
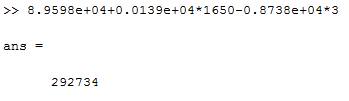


Figure 2. House price calculation and theta values using normal equation

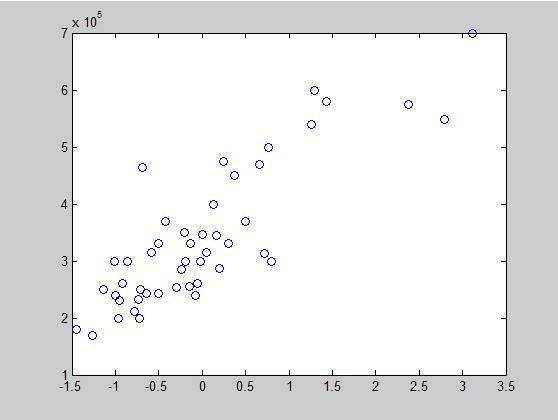


Figure 3. Price vs. Living Area – Preprocessed

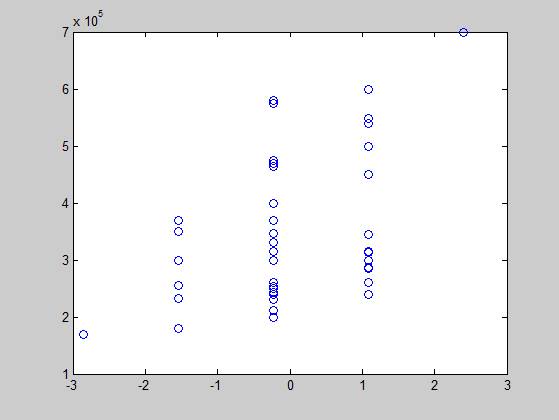


Figure 4. Price vs. Number of Rooms – Preprocessed

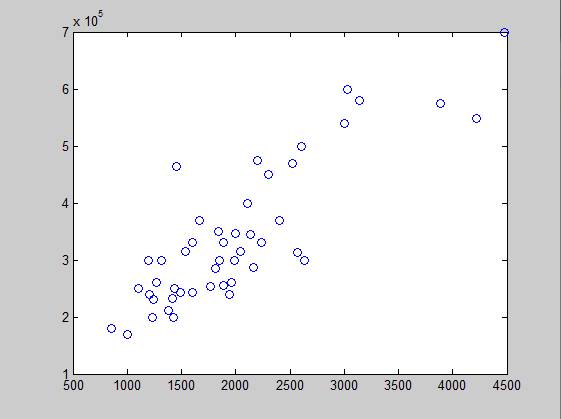


Figure 5. Price vs Living Area

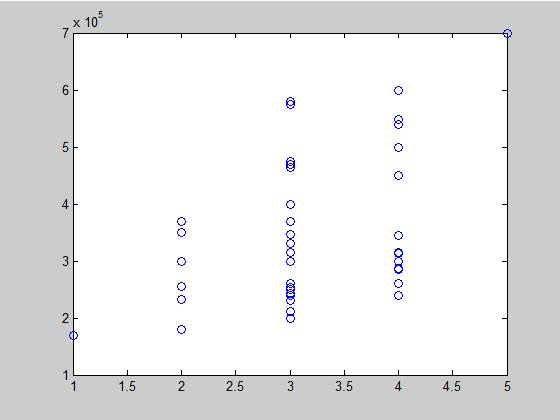


Figure 6. Price vs Number of Rooms

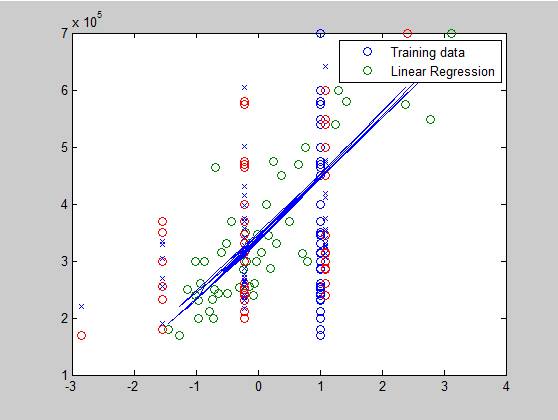


Figure 7. Theta plot

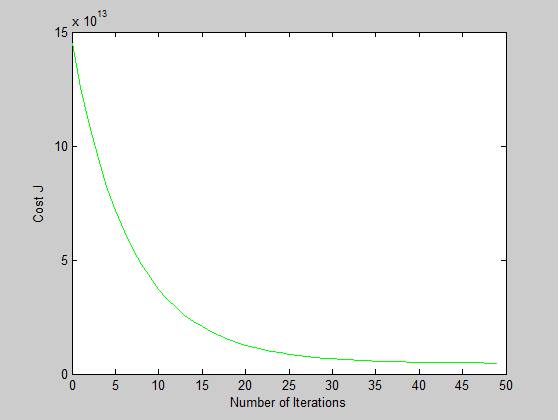


Figure 8. Cost function

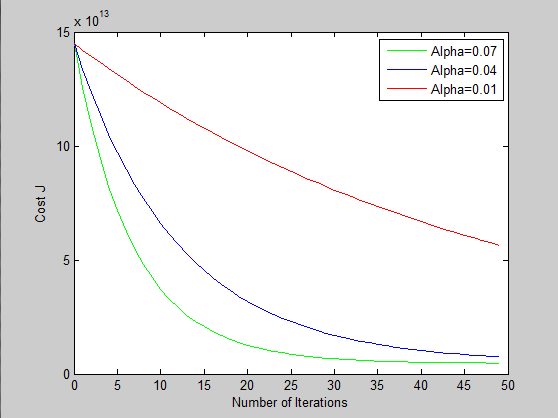


Figure 9. Comparison plot

For custom data

|  |  |  |  |
| --- | --- | --- | --- |
| Index | Degree of Mental Retardness | Degree of Distrust of Doctors | Degree of Illness |
| 1 | 2.8 | 6.1 | 44 |
| 2 | 3.1 | 5.1 | 25 |
| 3 | 2.59 | 6 | 10 |
| 4 | 3.36 | 6.9 | 28 |
| 5 | 2.8 | 7 | 25 |
| 6 | 3.35 | 5.6 | 72 |
| 7 | 2.99 | 6.3 | 45 |
| 8 | 2.99 | 7.2 | 25 |
| 9 | 2.92 | 6.9 | 12 |
| 10 | 3.23 | 6.5 | 24 |
| 11 | 3.37 | 6.8 | 46 |
| 12 | 2.72 | 6.6 | 8 |
| 13 | 3.47 | 8.4 | 15 |
| 14 | 2.7 | 5.9 | 28 |
| 15 | 3.24 | 6 | 26 |
| 16 | 2.65 | 6 | 27 |
| 17 | 3.41 | 7.6 | 4 |
| 18 | 2.58 | 6.2 | 14 |
| 19 | 2.81 | 6 | 21 |
| 20 | 2.8 | 6.4 | 22 |
| 21 | 3.62 | 6.8 | 60 |
| 22 | 2.74 | 8.4 | 10 |
| 23 | 3.27 | 6.7 | 60 |
| 24 | 3.78 | 8.3 | 12 |
| 25 | 2.9 | 5.6 | 28 |
| 26 | 3.7 | 7.3 | 39 |
| 27 | 3.4 | 7 | 14 |
| 28 | 2.63 | 6.9 | 8 |
| 29 | 2.65 | 5.8 | 11 |
| 30 | 3.26 | 7.2 | 7 |
| 31 | 3.15 | 6.5 | 23 |
| 32 | 2.6 | 6.3 | 16 |
| 33 | 2.74 | 6.8 | 26 |
| 34 | 2.72 | 5.9 | 8 |
| 35 | 3.11 | 6.8 | 11 |
| 36 | 2.79 | 6.7 | 12 |
| 37 | 2.9 | 6.7 | 50 |
| 38 | 2.74 | 5.5 | 9 |
| 39 | 2.7 | 6.9 | 13 |
| 40 | 3.08 | 6.3 | 22 |
| 41 | 2.18 | 6.1 | 23 |
| 42 | 2.88 | 5.8 | 31 |
| 43 | 3.04 | 6.8 | 20 |
| 44 | 3.32 | 7.3 | 66 |
| 45 | 2.8 | 5.9 | 9 |
| 46 | 3.29 | 6.8 | 12 |
| 47 | 3.56 | 8.8 | 21 |
| 48 | 2.74 | 7.1 | 13 |
| 49 | 3.06 | 6.9 | 10 |
| 50 | 2.54 | 6.7 | 4 |

Table 1. “Custom” data used for 2nd trial

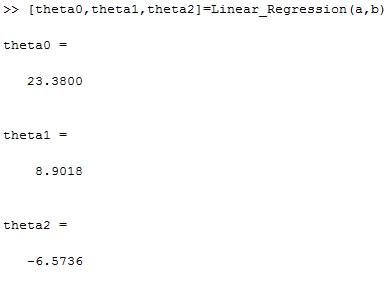
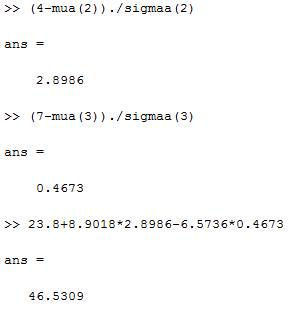


Figure 10. Illness calculation and theta values using gradient descent

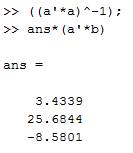
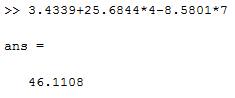


Figure 11. Illness calculation and theta values using normal equation

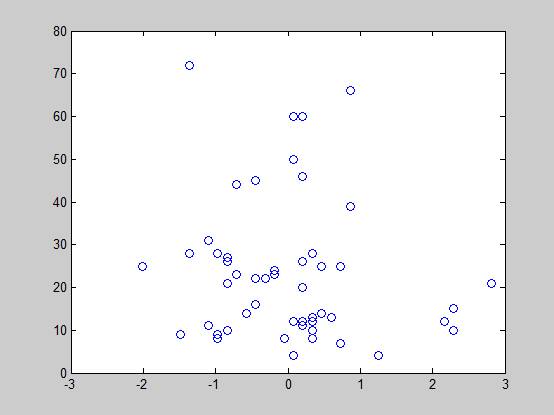


Figure 12. Illness vs Distrust - preprocessed

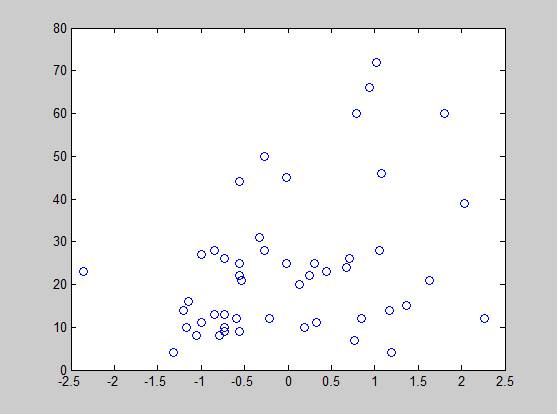


Figure 13. Illness vs Retardedness – preprocessed

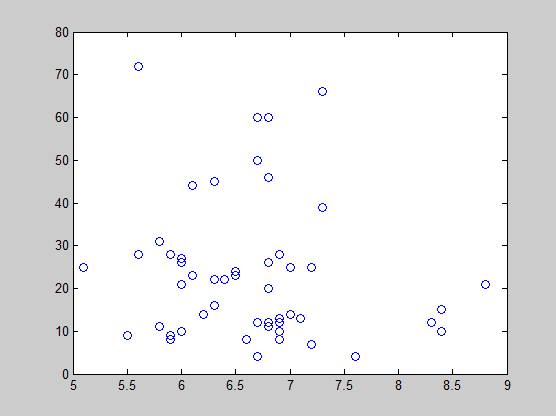


Figure 14. Illness vs Distrust

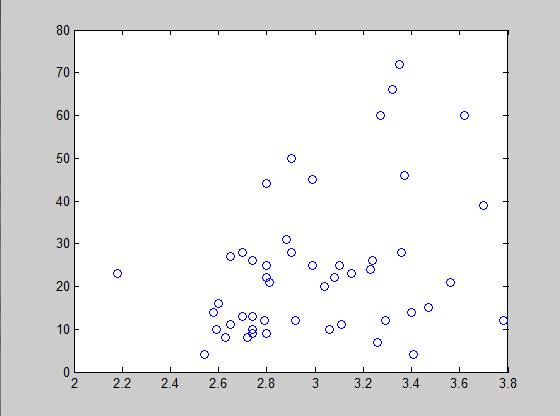


Figure 15. Illness vs Retardedness

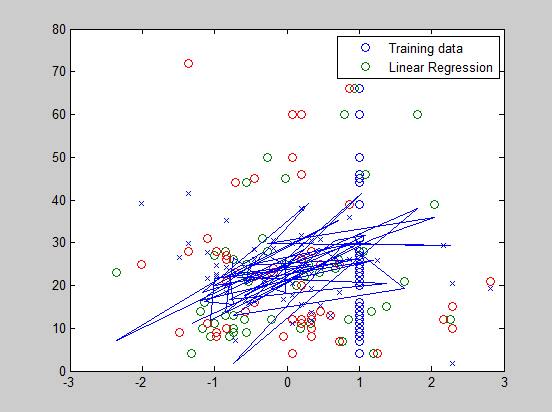


Figure 16. Theta plot

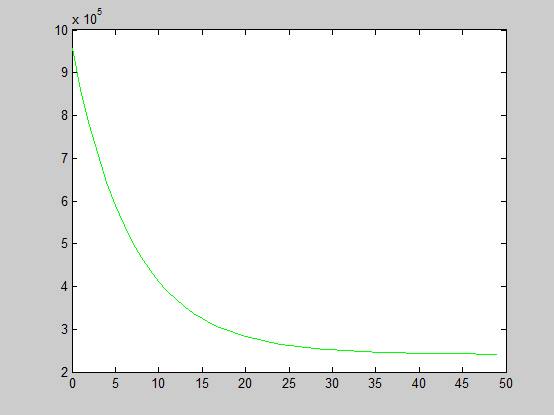


Figure 17. Cost Function

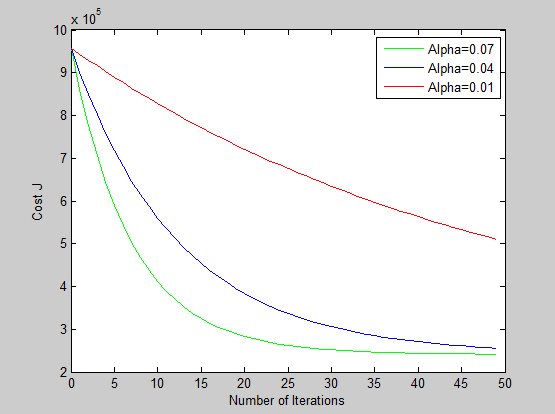


Figure 18. Comparison plot

# Analysis and Conclusion

If the learning rate is relatively small, the cost function decreases slowly, which means slow convergence during gradient descent. If the learning rate is relatively large, the cost function will drasticly increase instead of decaying and the gradient descent will not converge at all. In the experiment, at a certain point, increasing the learning rate past that point would not result in increasing the speed of convergence.

# 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

function multivariate(x,y)

theta2 = zeros(size(x(1,:)))';

theta = zeros(size(x(1,:)))';

h = zeros(size(x(1,:)))';

h1 = zeros(size(x(1,:)))';

alpha1 = .07; alpha2 = .04; alpha3 = .01;

j1 = zeros(50,1);

j2 = zeros(50,1);

j3 = zeros(50,1);

for num\_iterations = 1:50

j1(num\_iterations) = (1/2\*47)\*(x\*theta - y)'\*(x\*theta - y);

h=0;

h1=0;

for i=1:47

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

h1=h;

end

theta=theta2-((alpha1/47)\*h)';

theta2=theta;

end

theta2 = zeros(size(x(1,:)))';

theta = zeros(size(x(1,:)))';

for num\_iterations = 1:50

j2(num\_iterations) = (1/2\*47)\*(x\*theta - y)'\*(x\*theta - y);

h=0;

h1=0;

for i=1:47

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

h1=h;

end

theta=theta2-((alpha2/47)\*h)';

theta2=theta;

end

theta2 = zeros(size(x(1,:)))';

theta = zeros(size(x(1,:)))';

for num\_iterations = 1:50

j3(num\_iterations) = (1/2\*47)\*(x\*theta - y)'\*(x\*theta - y);

h=0;

h1=0;

for i=1:47

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

h1=h;

end

theta=theta2-((alpha3/47)\*h)';

theta2=theta;

end

figure;

plot(0:49, j1(1:50), 'g')

hold on;

plot(0:49, j2(1:50), 'b')

plot(0:49, j3(1:50), 'r')

xlabel('Number of Iterations')

ylabel('Cost J')

legend('Alpha=0.07', 'Alpha=0.04','Alpha=0.01');

end