LAB ACTIVITY 1: LINEAR REGRESSION

CARANDO, Paolo Creas
GOMENDOZA, Frank
LIM, Theo Paolo
LORENA, Neil John
ZABAT, Juan Paolo

I.INTRODUCTION

In this experiment, linear regression will be part of machine learning. It will determine on a given input, an output depending on the given Training Data. Gradient Descent will be used to predict a certain output. For example given a set of data, the machine will determine what output will be given to the user by calculating the values of theta0 and theta1.

Two equations will be needed on is the hypothesis (Equation 1) that will be inserted to the Gradient Descent equation (Equation 2)

$$h_\theta(x) = \theta^T x = \theta_0 + \theta_1 x_1$$

Equation 1: Hypothesis

Take note that x will be equal to 1 to obtain a line equation which is y = mx+b, where m is theta1 and b will be theta0. The goal of this is to minimize the sum of the squared errors to fit the given training data. Done correctly, the output plot will be within the training data.

II.PROCEDURE

PROCEDURE 1

1. In the first exercise, calculate the values of theta1 and theta 0.
2. In order to calculate the theta, load the training data and initialize all constants such as the theta, and alpha. See Code 1.

```
x = load('ex2x.dat');
y = load('ex2y.dat');
alpha = 0.07;
theta = [0 0];
```

Code 1: Loading of Training Data and Constants

3. Plot the training data using plot command (Code 2)

```
plot(x,y,'o')
xlabel('Age in Years');
ylabel('Height In Meters');
```

Code 2: Plotting of Training Data

4. After initializing, assign m as the total length of the 'x' training data as well as including an 'ones' column to the x parameter, seen at Code 2.

```
m = length(x);
x = [ones(m,1), x];
```

Code 3: Setting up total data and x parameter.

4. Obtain the first iteration of the training data using the Gradient Descent equation, see Equation 1.

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right) x_j^{(i)} \qquad \text{(for all } j\text{)}$$

Equation 2: Gradient Descent

5. Implement the Gradient Descent in Matlab/Octave to obtain theta0 and theta1.

6. Plot the first iteration with the original training data; see plot Code 4 for holding on the original data.

```
plot(x(:,2),y,'o')
hold on
plot(x(:,2),x*theta','-')

xlabel('Age in Years');
ylabel('Height In Meters');
```

Code 4: Plotting Training Data with First Iteration

PROCEDURE 2

1. After accomplishing the first iteration, implement a 'for' loop for the 1500 iterations.
2. Plot the 1500 iteration with the original Training Data, see code 4 for plotting.

PROCEDURE 3

1. After obtaining the linear regression, the theta can now calculate the height for ages 3.5 and 7.

PROCEDURE 4

1. Obtain the contour plot of Procedure 3, number 1.

III.RESULTS AND DISSCUSION

PROCEDURE 1

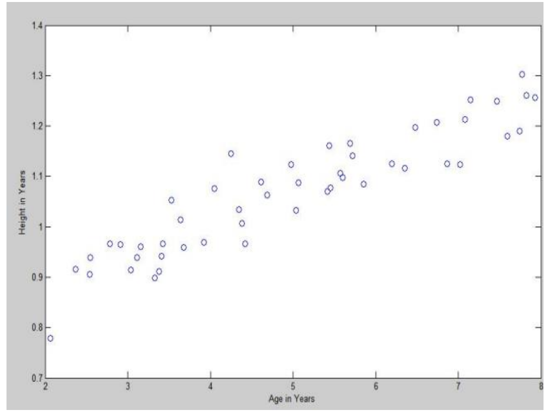

Fig. 1. Training Data Plot with Linear

PROCEDURE 2

Gradient Descent with linear regression (1st Iteration)

```
h(1) = (((x*theta')-y)'*x(:,1))*alpha/m;
h(2) = (((x*theta')-y)'*x(:,2))*alpha/m;
theta = theta - h;
```

Code 5: Octave code of 1st Iteration Gradient Descent with linear regression

1st Iteration

$\theta 1 =$ __0.0745__

$\theta 2 =$ ____0.38___

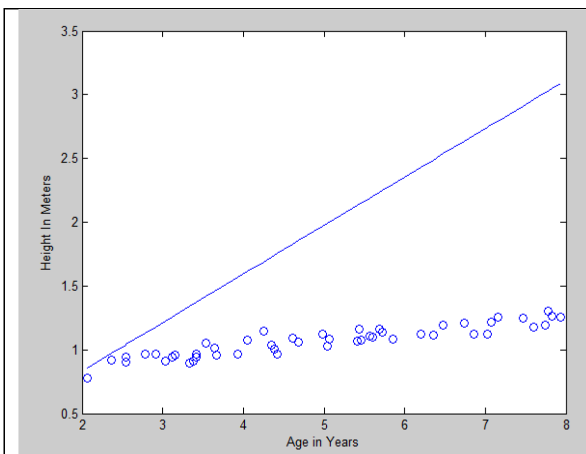

Fig. 2: Plot for 1st Iteration

PROCEDURE 2

```
For i=1:1500
h(1) = (((x*theta')-y)'*x(:,1))*alpha/m;
h(2) = (((x*theta')-y)'*x(:,2))*alpha/m;
theta = theta - h;
end
```

Code 6: Octave code of 1500 Iterations Gradient Descent with linear regression

1500th Iteration

$\theta 1$ = __0.750150__

$\theta 2$ = __0.063883__

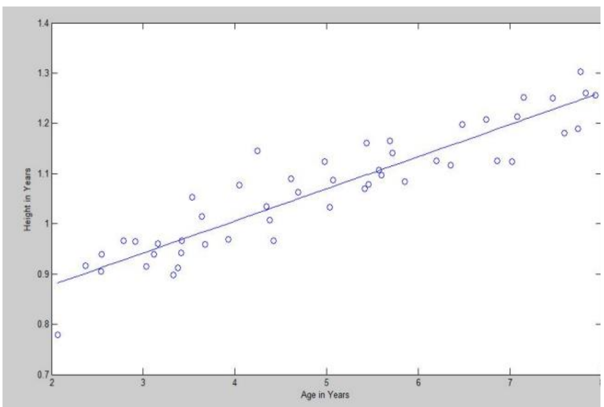

Fig. 3. Training Data Plot with Linear Regression Line

PROCEDURE 3

```
a1 = theta(2)*3.5 + theta(1)*1
a2 = theta(2)*7 + theta(1)*1
```

Code 6: Code for calculating the height.

Height Predictions

Age = __3.5__Height = _0.97374 m_

Age = __7___ Height = _1.1973 m__

PROCEDURE 4

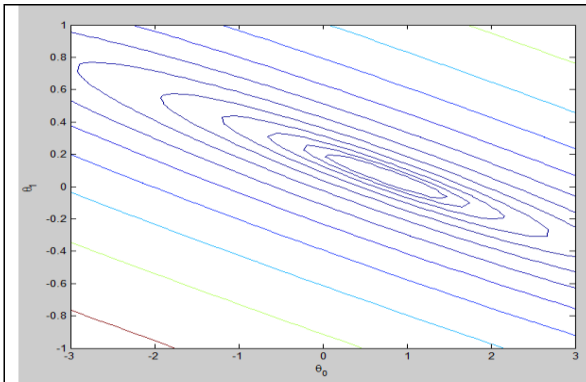

Fig. 4. Surface Plot of the Calculated Heights

```
mesh(theta0_vals,theta1_vals,J_vals);
```

Code 7: Plotting the contour plot

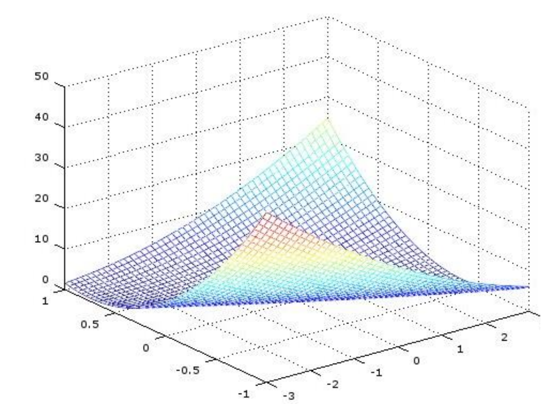

Fig. 5. Contour Plot

IV.CONCLUSION

The group concluded that by applying Linear Regression in teaching a machine with a given set of training data, the machine would be able to learn the system and generate a certain output given that enough iteration are performed. Evident in Fig. 2, the group just performed 1 iteration and resulted to a distant result. But in Fig. 3, 1500 iterations were performed and a more accurate result was obtained. Therefore, with an accurate machine that learned the system, the group used the equation of a line to predict the height in specific ages.