Linear Regression

AQUINO, Patricia Rose M. (11109734)

BASALLOTE, Lawrence Andrew G. (11123192)

CARAG, Stephanie L. (11130946)

JACINTO, Dan Emanuel C. (11115491)

LUNASCO, Jan,Osbert R. (10732837)

1 Introduction

Machine Learning is when computers can learn a pattern and come up with a conclusion or prediction given a set of data without these computers being initially programmed. The different notable elements for machine learning are the following: Experience, Task, Performance Measure. Experience is the set of data given to the computer from which it learns the patterns. The computer is gaining knowledge from these experiences geared towards performing a particular task. Performance measure, in other words, can be referred to as the accuracy rate of the performed tasks. Gradient descent is used for minimizing functions. It begins with an initial set of values and then moves towards a set of values minimizing the function. Calculus is used for the minimization technique. Linear regression is used to fit a line to a set of plotted data. The linear regression formula is seen in equation 1 and the gradient descent formula is seen in equation 2. The linear regression model is plugged in to the gradient descent formula.

$$h_\theta\left(x\right) = \theta^T x = \sum_{i=0}^{n} \theta i x_i \tag{1}$$

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=0}^{m} (h_\theta(x^i) - y^i) x_j^i \tag{2}$$

Let us understand that to be able to fit a line across a particular given data, we need to identify the best-fitted slope (recalling the line equation of $y = mx + b$) for a lower error rate. We need to identify the best m and b, which are the slope and y-intercept, respectively, for linear regression. More will be discussed in the following portions of this report.

2 Procedure

1. Implement gradient descent using a learning rate of $\alpha = 0.07$. Since Matlab/Octave and Octave index vectors starting from 1 rather than 0, youll probably use theta (1) and theta(2) in Matlab/Octave to represent θ_0 and θ_1. Initialize the parameters to θ_0 = 0 (i.e., θ_0 = θ_1 = 0), and run one iteration of gradient descent from this initial starting point. Record the value of of θ_0 and θ_1 that you get after this first iteration. (To verify that your implementation is correct, later well ask you to check your values of θ_0 and θ_1 against ours.)
2. Continue running gradient descent for more iterations until θ converges. (this will take a total of about 1500 iterations). After convergence, record the final values of θ_0 and θ_1 that you get. When you have found θ, plot the straight line fit from your algorithm on the same graph as your training data. The plotting commands will look something like this:

```
hold on
plot(x(:,2), x*theta, 1)
legend(Training data, Linear regression)
```

Note that for most machine learning problems, x is very high dimensional, so we dont be able to plot $h_\theta(x)$. But since in this example we have only one feature, being able to plot this gives a nice sanity-check on our result.
3. Finally, wed like to make some predictions using the learned hypothesis. Use your model to predict the height for a two boys of age 3.5 and age 7. Debugging If you are using Matlab/Octave and seeing many errors at runtime, try inspecting your matrix operations to check that you are multiplying and adding matrices in ways that their dimensions would allow. Remember that Matlab/Octave by default interprets an operation as a matrix operation. In cases where you dont intend to use the matrix definition of an operator but your expression is ambiguous to Matlab/Octave, you will have to use the dot operator to specify your command. Additionally, you can try printing x, y, and theta to make sure their dimensions are correct.

3 Results and Discussion

A. Draw Data Plot

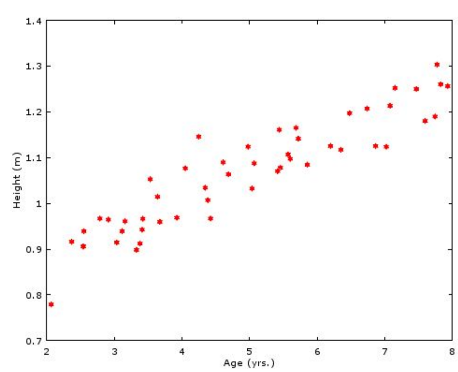

Fig. 1. Data plot of y and x data.

By affixing our signatures, we, the authors, pledge that: we have completed this course work on our own; we have not used any unauthorized material/assistance/help on this course work; and we have not given directly or indirectly to any other student/unauthorized person/means any access to any part of the specified course work. Coram Deo.

Lab Work Date: Sept. 07, 2015
Submission Date: Sept. 14, 2015

Figure 1 shows the data plot for age corresponding its height which is represented by x and y. Using the load command, $attr\_name = load('data\_name.dat')$, to input the training set.

B. Record the first iteration
$\theta_0 = 0.074528$
$\theta_1 = 0.380022$

C. Record the 1500 iteration
$\theta_0 = 0.750150$
$\theta_1 = 0.063883$
Using the formula of the gradient descent with a learning rate of $\alpha = 0.07$, the value for θ_0 and θ_1 for the first iteration, found in B, and its value for 1500 iteration, found in C, were recorded. θ_0 and θ_1 represents the slope and y-intercept.

D. Draw a line in Figure 1

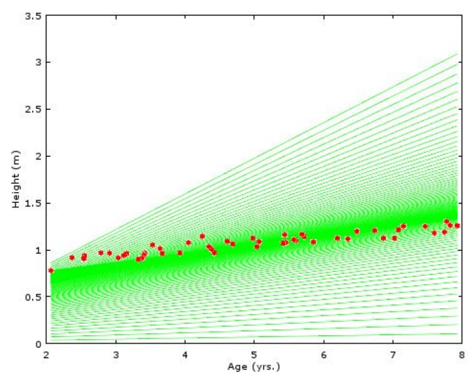

Fig. 2. Data plot of Fig. 1 with gradient descent.

The green lines shown in Figure 2 represents a linear regression model applied in a gradient descent formula. Gradient descent is important in minimizing the error which can be computed using the difference between the distance of the gradient line value and the point(data) value.

E. Prediction of height and final contour plot

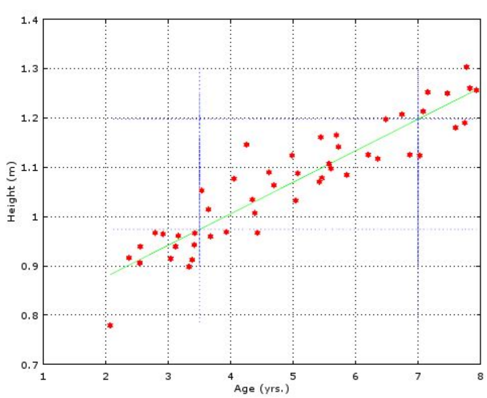

Fig. 3. Prediction of Height when age is equal to 3.5 and 7.

$age = 3.5\ height = 0.9737$
$age = 7\ height = 1.1975$

Figure 4 shows the effect when the value of θ_0 and θ_1 are changed. It shows the gradient descent closer to the point that would minimize the cost function.

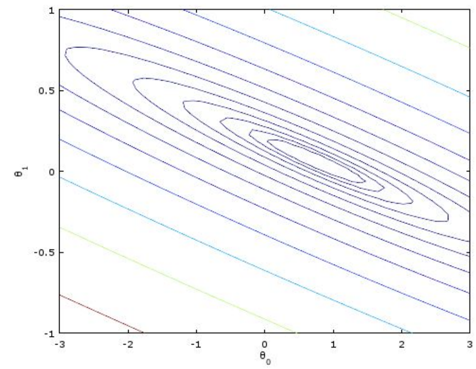

Fig. 4. Contour Plot

4 Conclusion

In this experiment, we learned about Linear Regression, a model used to represent the relationship between various elements. It can be solved using the basic equation, y = mx + b, which isthen iterated depending on the number of ”explanatory variables”. To implement the linear regression model in our experiment, we used Octave to program the formula for gradient descent. Then given a set of data, we plugged in the values into the formula to produce the graph of the gradient descent. Linear regression is a very important concept to machine intelligence since this model can be used to predictthe future values using the given data. This can also help a machine learn by finding the pattern between the past values.
In the experiment, we encountered several difficulties. For example, is the programming of the formula for the gradient descent. We had to come up with an algorithm that will make the program able to solve for the gradient descent with the given data. This helped us understand more how gradient descent works since we weren’t just plugging in values to the formula to arrive with the answer anymore. For the program to work, we had to visualize the specific steps to solving for the gradient descent, and then code it. One of the distinct differences between human and machine, is that a human can instantly visualize anything, whereas a machine has to break the steps down to the most basic of steps. If we will think like humans usually do, we can just plug in values repeatedly into the formula, then plot the graph. However, since we had to teach the program to do it, we had to break the steps down for it. This process helped us understand how the basic steps of plotting the gradient descent works.

5 References

http://openclassroom.stanford.edu/MainFolder/DocumentPage.php?course=MachineLearning&doc=exercises%2Fex2%2Fex2.html
http://spin.atomicobject.com/2014/06/24/gradient-descent-linear-regression/