

## Assignment 1

By: Jason Silva, ID: 801064375

Dated Completion on June 8th, 12024 HE (2024 AD)

The first assignment for ECGR 4105 about Linear Regression models and how to use them to predict future values based on a set of arbitrary training data.

my\_functions.py contains the necessary cost and gradient descent formulas made from scratch. Hopefully in future labs we will use libraries to make it easier.

### Problem 1

Problem 1 contains the linear regression techniques for each independent variable, with the assumption that only one of the inputs determines the output. X1, according to the author of this assignment attempt, is the best fitting; X2 and X3 are far too scattered, with the outliers causing the regression line to fall short. As an assumption, normalization would be necessary for those two inputs to minimize them for more accurate results (although not necessary for this course, I consider it as an option along with standardization.)

In regards to the learning rate, several adjustments were made (and kept consistent across the assignment) to analyze the effect of varying learning rates (alpha) on the Cost vs. Iterations graphs. They all managed to converge between 0.1 and 0.01 alpha rates, some faster than others. My assumption for the future: less iterations to convergence, the better the learning model.

### Problem 2

Problem 2 contains the same linear regression idea, but with multiple variables instead of just one independent variable. Several problems were encountered, and circumvented, during this portion: a lot of time was spent digging deeper into the Cost function of these models, all the regression lines do not seem to line up very well with all the values (X3, in this case, was chosen as the approximation factor.)

With the linear regression model determined, predictions of the output are listed as follows, with  $h_{\theta}(x)$  being the hypothesis formula given from this class:  $(X1, X2, X3) = (1, 1, 1) \rightarrow h_{\theta}(x) \rightarrow 5.311 - 2.003(1) + 0.533(1) - 0.265(1) = 3.576$   $(2, 0, 4) \rightarrow h_{\theta}(x) \rightarrow 5.311 - 2.003(2) + 0.533(0) - 0.265(4) = 0.245$   $(3, 2, 1) \rightarrow h_{\theta}(x) \rightarrow 5.311 - 2.003(3) + 0.533(2) - 0.265(1) = 0.103$

Perhaps these values do not accurately represent the data, but this homework assignment has proven nonetheless to be a learning experience from a Novice Python programmer (this author)