sCS 596 Machine Learning

Homework Assignment 3

(10 Credits)

Due: 11:59 pm, March 6, 2018

Notice: we remove the part for unit price prediction since it may result in confusions. Thus, this homework assignment includes only the practices for linear regression.

The goal of this homework assignment is to master the programming of Linear Regression method. Sample codes are provided and you are required to complete missing lines, evaluate your codes and report your observations. Details instructions are as follows.

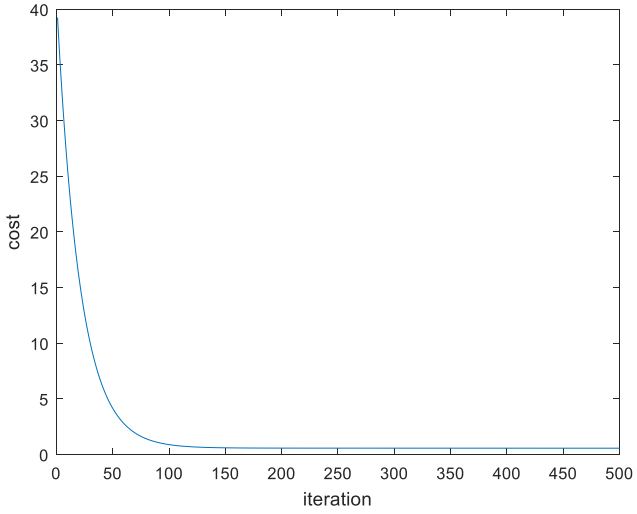
Overview. In this programing exercise, you will need to implement a linear regression algorithm for predicting student's university GPAs from two features, i.e., Math SAT and Verb SAT. You will be instructed to program the GD method to solve the linear regression model from a set of training samples. Then, you will apply the model to predict the university GPAs of testing samples.

Datasets. The student's data table is available in the attached 'sat.csv file, which includes a data matrix. Each row represents one of the 105 students and includes five properties. We will use the properties: Math SAT, Verb SAT as input features, and the property: University GPA as output labels. We split the whole dataset into two parts: the first 60 rows for training and the rest for testing.

Sample Codes. The file "main_ha3.py" provides the starting codes for 4 major steps: loading training and testing data, training the linear regression model, testing and evaluating the learned model.

The **first** step is to load student's data and split them into two sets, one for training and the other for testing. We use the function download_data() to load data from the 'sat.csv' file.

The **second** step will call the function gradientDescent() in GD.py, i.e., the implementation of gradient descent method, to obtain the optimal parameters and costs over iteration. The latter will be visualized as a convergence curve as the following:



The **third** step will apply the learned model (i.e., the optimal parameters) to predict the GPA for testing students. The **last** step will return the average error and standard deviation (STD) of the evaluation results.

There are four PLACEHOLDERs in the provided scripts: two in 'main_ha3.py" and the other two in "GD.py"

In "main_ha3.py",

PLACEHOLDER1: you will need to change the two variables: alpha and MAX_ITER and observe how the convergences curve and evaluation results change. Write down your observations in the report.

PLACEHOLDER2: You will also need to normalize the two input features and the output labels in the first step. We use a function rescaleMatrix() now. Please replace it with your own normalization codes. You can use the function rescaleMatrix() to verify if you codes are running properly.

There are a few different ways for normalizing these features, e.g., you can try scale every feature to be 0 and 1, or taking mean off every value, or others. Please test and evaluate at least one preprocessing way.

In "GD.py", you will need to implement the gradient descent function gradientDescent (). Replace the temporary code lines in PLACEHOLDER with your own codes.

PLACEHOLDER3: write your codes to update theta, i.e., the parameters to estimate, following the gradient direction.

PLACEHOLDER4: calculate the current cost with the updated model parameters, i.e. theta.

Write-up

In the report you will describe your algorithm and any decisions you made to write your algorithm a particular way. Then you will show and discuss the results of your codes following the above instructions. In the case of this project, show the convergence curves and quantitative results for each case of your implementations. Also, discuss anything extra you did. Feel free to add any other information you feel is relevant.

How to submit

- Submit your source codes and report using the SDSU Blackboard. The codes should be self-contained, and run without any error. Otherwise, penalty will be applied.
- Two re-submissions are allowed.
- A total of three late submissions are allowed in the whole semester.
- ***No hard copy required.***