**Machine Learning – HW 2**

Atieh Kashani

**Part 1: Unit price estimation**

In this question an iterative method has been implemented to predict the unit prices of meal items (fish, chip, ketchup). It starts with the random number for each unit price and also random numbers for the portions of the meal.

Stop conditions are based on the maximum number of iteration and comparing the current iteration error with threshold (the iteration error is the difference between the estimated price and true price). If during the iteration the error is less than the specific threshold the iteration is terminated.

The unit price is adjusted based on the error that is the difference between the estimated price and true price and according to the following equation for each item:

(Est Price – True Price) \*

- (Est Price - True Price) \*

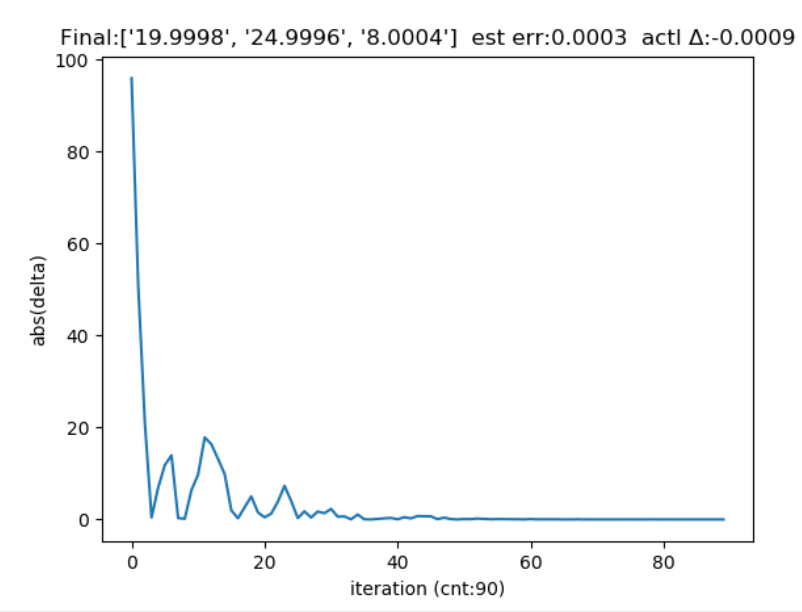
- (Est Price - True Price) \*

Also, for adjusting the unit price the contribution of each feature needs to be considered in this equation.

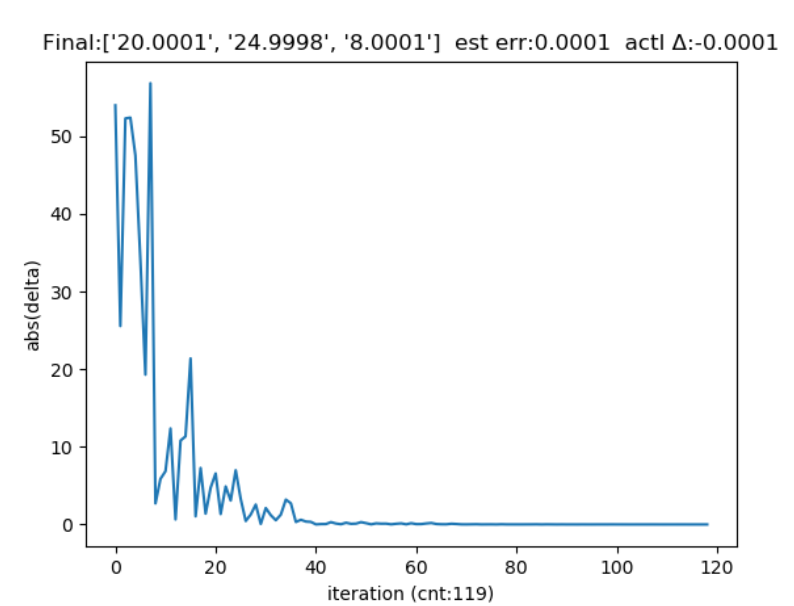
For learning rate, I chose the values 0.05, 0.01, 0.001 and tried different values for initial predictions and the stop conditions (maximum iterations and minimum delta).

The following figure shows the estimation error, final estimated unit price, the convergence curves for the best results:

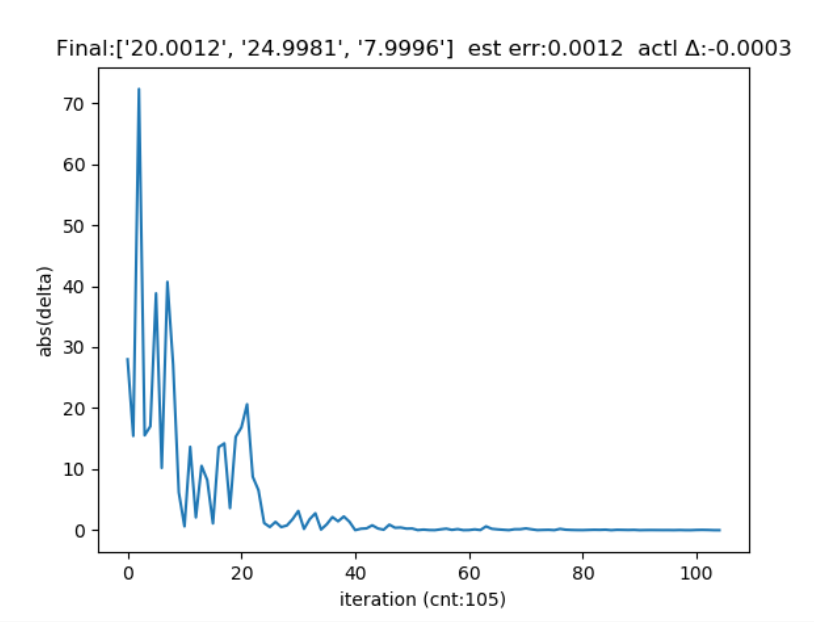
* Max\_Iteration: 150000 , Min\_Delta:0.001 , AlPHA= 0.01, initial unit prices= [20,20,20]



* Max\_Iteration: 100000 , Min\_Delta:0.0001 , AlPHA= 0.01, initial unit prices= [20,20,20]



* Max\_Iteration: 10000 , Min\_Delta:0.001 , AlPHA= 0.01, initial unit prices= [10,10,10]



**Part 2: Linear Regression**

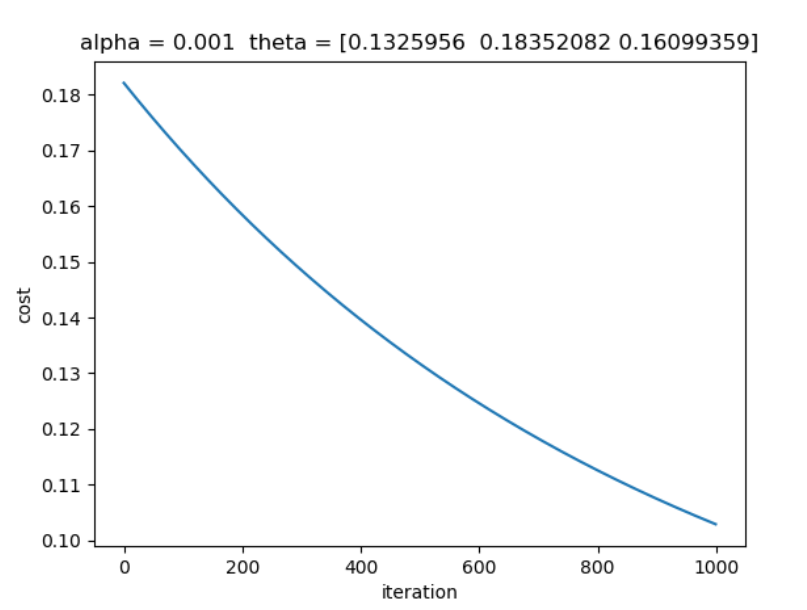
In this question a linear regression algorithm has been implemented to predict the student’s university GPAs from two features (Math Sat and Verb SAT). The provided dataset contains the required input features (Math Sat and Verb SAT) and output labels (university GPA). For data normalization I used normalized data between -1 and 1 using the given function.

After implementing the gradient descent to updated theta and cost function, I changed the learning rate and maximum iteration values and tried different configuration of these parameters. Following is the evaluation results and convergences curve:

* Alpha: 0.001 Max\_Iteration: 1000

The learning rate is low, so the convergence is slow

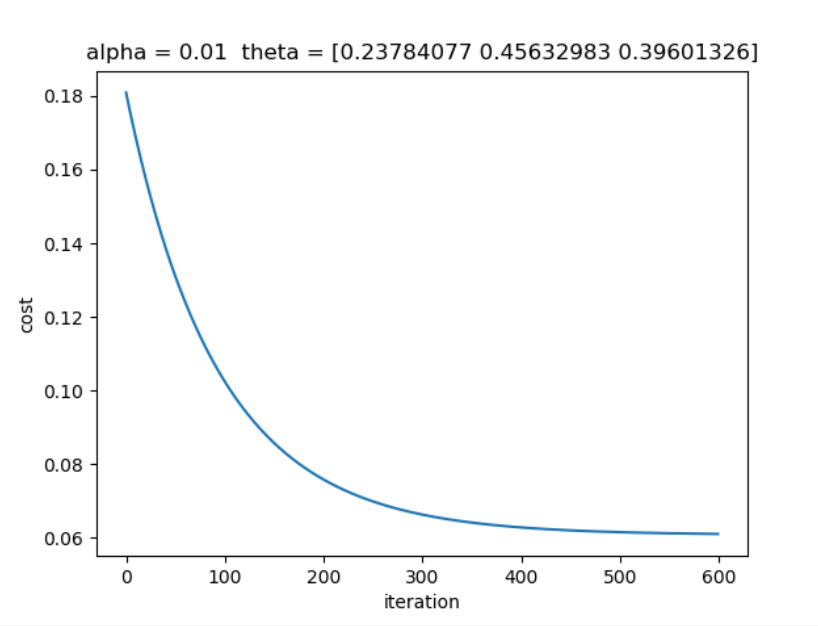
Average error: 0.34021990016789 Standard deviation: 0.2327348501



* Alpha: 0.01 Max\_Iteration: 600

The learning rate is higher, so the convergence is faster

Average error: 0.33786016111113304 Standard deviation: 0.24940733947168467



* Alpha: 0.05 Max\_Iteration: 1500

The learning rate is higher, so it affects the model and the convergence curve

Average error: 0.33786016111113304 Standard deviation: 0.24940733947168467

