Assignment – 4

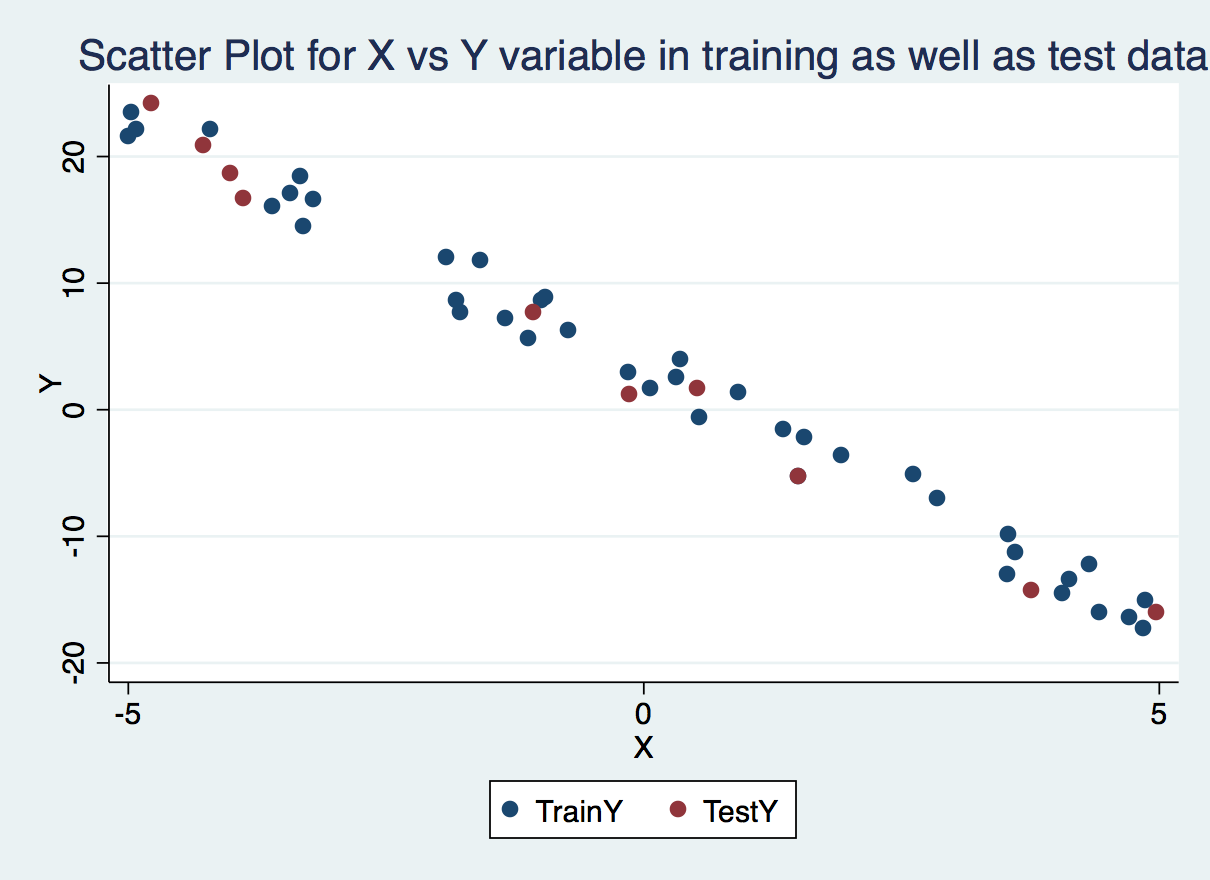
Foundation of Artificial Intelligence – CS2701

Name: Khushboo Thaker

Email: [kmt81@pitt.edu](mailto:kmt81@pitt.edu)

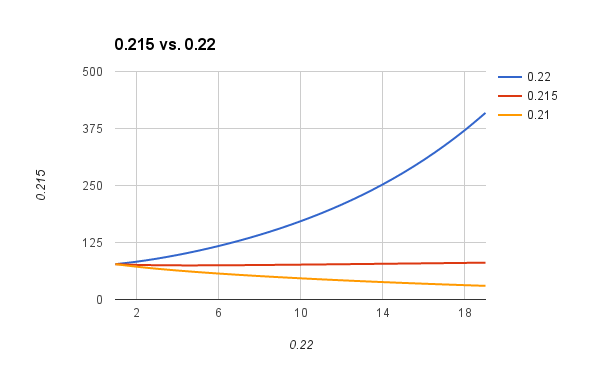
# Part1:

## Scattered Plot of test and train data:

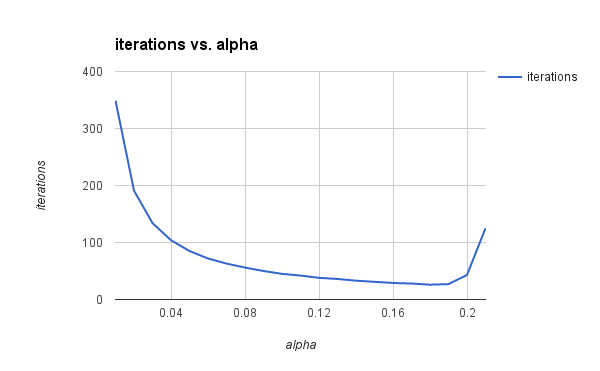


## Choice of alpha

The choice of alpha should be such that it does not cause deviation from optimum (minimum of the function). As can be seen from the figure below if the alpha is above 0.21 it causes alpha to deviate from the original objective function. The below graph has x-axis as number of iteration and y axis mean squared error(MSE) plotted at different alpha values. (0.22,0.215,0.21). I did not plotted values for all alpha values because then it becomes hard to differentiate due to large MSE scale and same is the case with too many iterations.



I have selected alpha with 0.05 as it neither converges too fast nor too slow. Also as we have small number of features and training data it is not a overhead. But yes in practical applications it might be problematic. Below is the graph showing the number of iterations required for convergence in with different alpha values. Not mentioned the iterations for greater than alpha value as it starts diverging.



## Convergence Criteria

The main task of gradient descent is to find out the global optimum. For which it moves in the direction where the objective function value is minimum.

My convergence criteria is “Difference between previous and current value of objective function”

If( abs(prev – curr) < mindiff) – that means the objective function is not changing much so the on the point of minimum. Also an upper limit to iteration can be added so that in case of divergence the loop doesn’t go to infinity but alpha takes care of this.

Also the choice of difference (mindiff) helps in stopping from overfitting of dataset where a little error is acceptable for generalization. But I am not very sure how this works in case of linear function.

Also I tried to keep number of iterations to be convergence criteria but then for different value of alpha the convergence iterations required are different. Thus selection of this criteria works for all alpha values

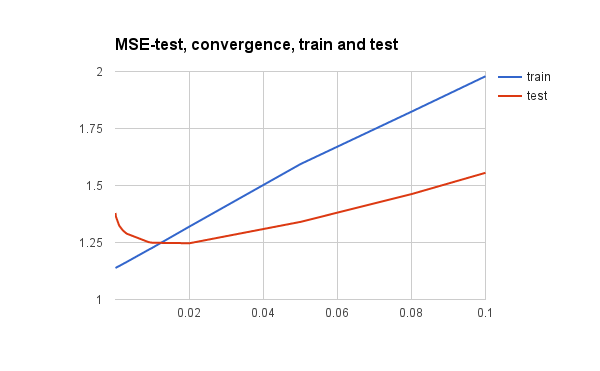
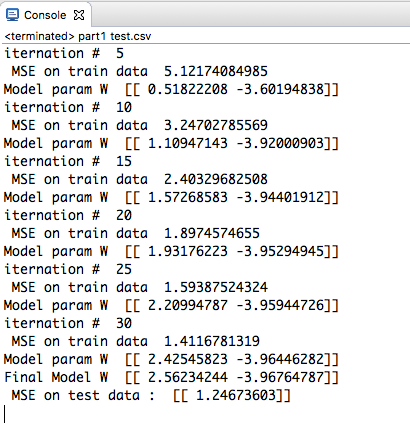


Figure Convergence rate and overfitting issue:

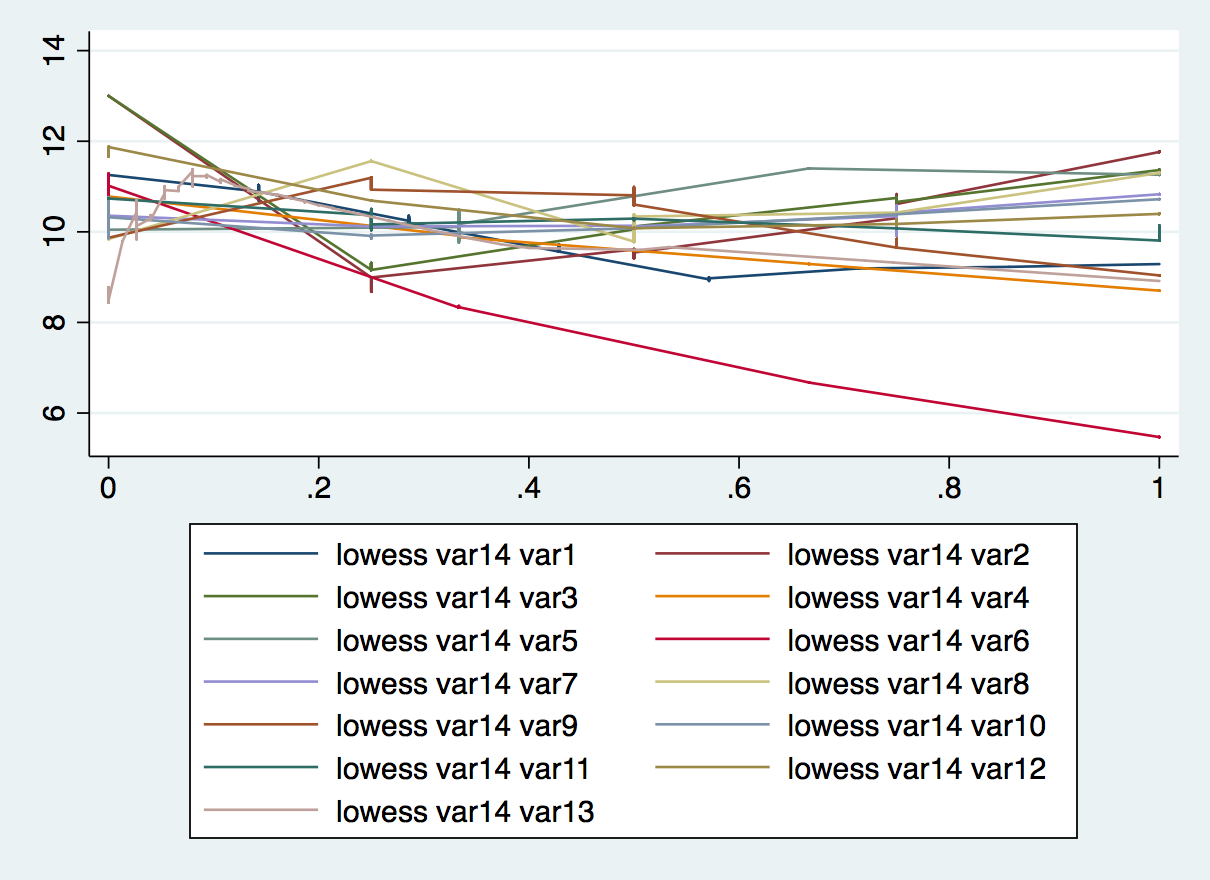
OUTPUT: alpha = 0.05 / convergence = abs(prevobj - currobj) < 0.02

Python lin\_reg.py part1 “part1 train.py” “part1 test.py”



# Part 2

Plotting the data with respect to the grade of student (var14)



The relations don’t seem to be linear.

## Briefly present your learning rate choice and your convergence criterion

I took the same learning rate of 0.02 and convergence criteria as in part 1.

## How did you decide to choose these features? What other options have  you experimented?

I did below experiments for feature selection.

Different things tried:

|  |  |
| --- | --- |
| Different types of linear model | Performance of my linear model on test data |
|  | MSE at alpha = 0.02 |
| Use of complete features without any transformation or pre-processing  Python multi part2.csv | 9.67 |
| Use of complete features with log transformation of each feature  Python log part2.csv | 9.61 |
| Manually created features\*  Python manual part2.csv | 9.47 |
| Automatic selection of features based on forward selection  Python subset part2.csv  Top features :  log(family relation)  sqr(fathers education)  sqr(workday alcohol)  sqr(going out with friends)  family relationship  log(free time after school)  python subset part2.csv | 9.00 |
| Correlation based feature selection. In this method at each step I select the feature which is maximally correlated with the target variable (y) but minimally correlated with the existing subset selected.  Python corr part2.csv | 9.7 |

**Final Feature:**

I found subset method to be more accurate although it is much more time and resource consuming.

My features are:

log(family relation)

sqr(fathers education)

sqr(workday alcohol)

sqr(going out with friends)

log(free time after school)

number of past class failures

This model gave me the MSE of 9.00

\*Manual features augmented by hand: [not in my final model]

1. maximum of parent education

2. number of previous failures

3. efficiency effecting

(( time with friends + travel time + free time after school) + number of school abscens/20)

4.health effecting – alcoholic + health issues