Kulkarni, Anushree Milind

AXK180027

FACEBOOK COMMENT VOLUME PREDICTION

ASSIGNMENT 1

**Goal:** To explore the gradient descent algorithm and its various parameters and predict the expected number of comments from Facebook Comment Volume Dataset.

**Tasks:**

Part 1: The dataset was downloaded, Variant No. 1 and Variant No. 5 of Training Sets and all variants from Test Sets were combined and split into 70/30 ratio by a random number generator.

Part2: As required, a model containing 17 parameters from the dataset was chosen. These parameters were picked based on domain knowledge and personal experience with the platform.

The chosen parameters are:

* Page Popularity/likes(1)
* Page Checkins(2)
* The total number of comments before selected base date/time.(3)
* The number of comments in last 24 hours, relative to base date/time(4)
* The number of comments in last 48 to last 24 hours relative to base date/time.(5)
* The number of comments in the first 24 hours after the publication of post but before base date/time.(6)
* Base time(7)
* Post length(8)
* Post Share Count(9)
* H Local(10)
* Post Promotion Status(11)
* Post Published Weekday (Sunday as base variable)(12-17)

Part 3:

No. of comments= b0+b1\*(1)+b2\*(2)+b3\*(3)+b4\*(4)+b5\*(5)+b6\*(6)+b7\*(7)+b8\*(8)+b9\*(9)+b10\*(10)+b11\*(11)+b12\*(12)+b13\*(13)+b14\*(14)+b15\*(15)+b16\*(16)+b17\*(17)

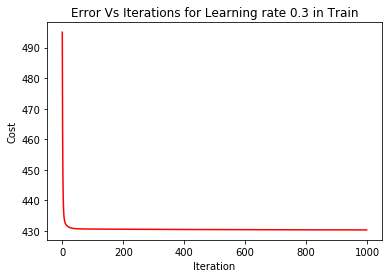
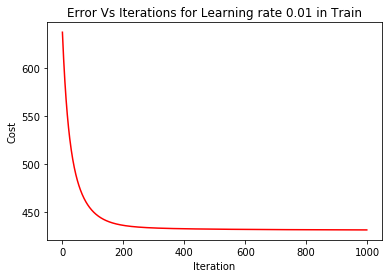
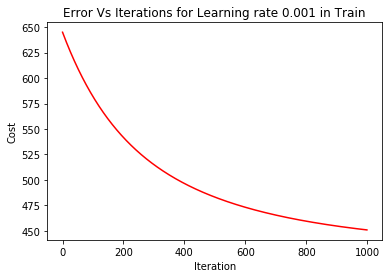
The gradient descent algorithm was implemented with batch update rule.

Initial parameter values, i.e. the values of Betas before running gradient descent are assumed to be 0.

For training data, alpha=0.001 ,0.01, 0.3:

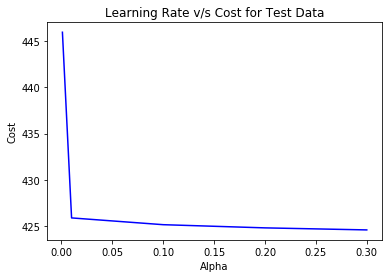
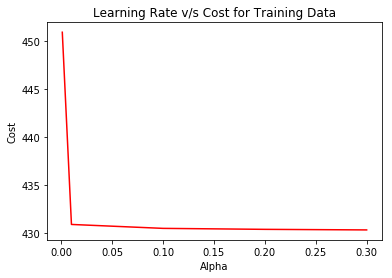
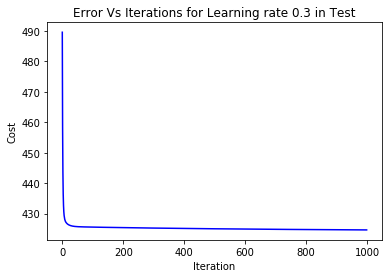
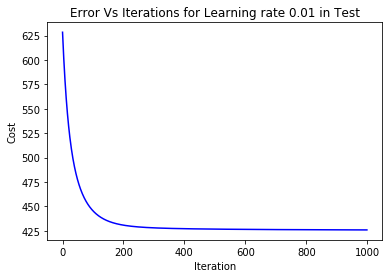
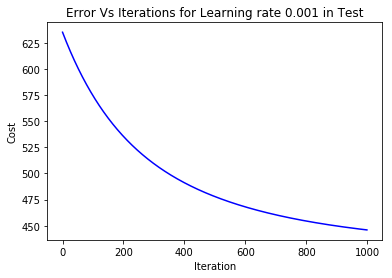
**Observations from Experimentation:**

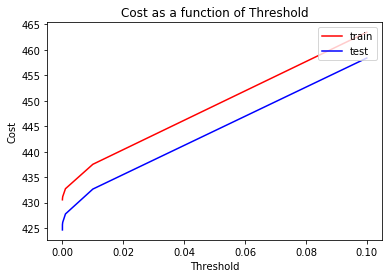
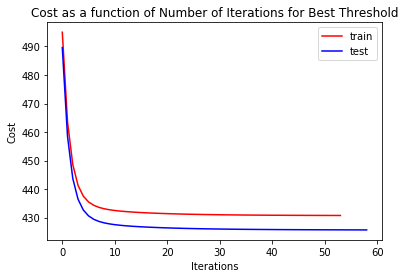
**Experiment 1:** As described in the experiment, on running the gradient descent function for train and test sets, for different learning rates, the observed variation is as shown in plots below



For test data, alpha=0.001, 0.01, 0.3:

The comparison of alpha against the learning rate for test and train data sets is as follows:





Here we can observe that as the value of threshold is reduced, the cost also reduces. As sensitivity to cost is a desirable feature in the model, we have picked a threshold value of 0.001%. The cost is 425.655 which is lower than previous experiment.

On plotting cost as a function of number of iterations for our chosen threshold value, the train and test data show comparable plots.

Thus, we can now save computing costs by setting a threshold acceptable to the desired accuracy of our predictions and minimum cost.

Experiment 2: As described in the experiment, on running the gradient descent function for train and test sets, for different thresholds for convergence and learning rate=0.3, the observed variation is as shown in plots below:

As it is sufficiently evident from the trend of the above plots, for a given alpha, the error tends to reduce with every iteration.

Also, as the learning rate increases, the, point of convergence is met sooner and is more defined.

We have thus chosen, a learning rate of 0.3 as it offers a low cost of 430.36 and the size of our data is well within our computing power that is associated with this learning rate.

Thus, there is a choice to be made here, between the computing cost and the accuracy of our predictions, leading us to the second experiment.

**Discussion:**

The results that we observe from the above experiments suggest that in order to run an efficient gradient descent, we must consider the learning rate, computing power and most importantly a low cost value. The accuracy desired in our predictions dictate the threshold value.

The betas observed in the final model indicate that for every increase in 10 likes for the page, the comments increase by 9.2 comments, longer the post fewer the comments received as not many tend to read it, with an increase in time of course the number of comments increase. The base time is the time of simulation of the scenario, thus, the farther away the base time is from the time of posting, it will have more comments.

In order to get better results, I would explore some interaction terms between the day of the post going live and the comments it received in the first 24 hours. Such interactions provide a lot more balanced insight into the conditional probabilities that occur in real world.

**Experiment 4:**

As prescribed in the experiment, on picking five variables, the cost computed on train data is 510.41 on the train data and 505.41 on test data. The picked variables are Likes, Check ins, base time, post length, and H local. These variables intuitively contribute the most to the comments a post might receive as they might help the post reach the feed of a relevant audience.

Our betas are

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Intercept | Likes | Checkins | Base Time | Post length | H Local |
| 7.2913296 | 0.9215202 | -0.24705152 | 12.40460703 | -8.31708318 | 0.72905507 |

This error is higher than the 10+ parameter model as we are missing out on some important variables like the day on which the post was posted and the how well it was received in terms of shares on the platform. It is less than model picked randomly.

**Experiment 3:**

As prescribed in the experiment, on picking five random variables, the cost computed is 516.97 on the train data and 511.82 on test data. Likes, check ins, talking about, comments within 48-24 hours and H local. The error is higher in this case as compared to the previous 430.36.