**Facebook Comment Volume (Regression)**

Linear Regression Model:

Target Variable = -0.02519637 \* Page Popularity/likes -0.00197713 \* Page Checkins -0.01987703 \* Page Category + 0.55190598 \* CC2 -0.04211995 \* CC3 -0.01223099 \* CC4 + 0.12678152 \* Post Share Count +0.02057487 \* H Local -0.09385726 \* Base Time + 0.07958334 \* D5 - 0.04071064 \* D10

Experimented with various values of learning rate ∝ and reported the findings as how the error varies for train and test sets with varying ∝.

Threshold value is 0.00000001

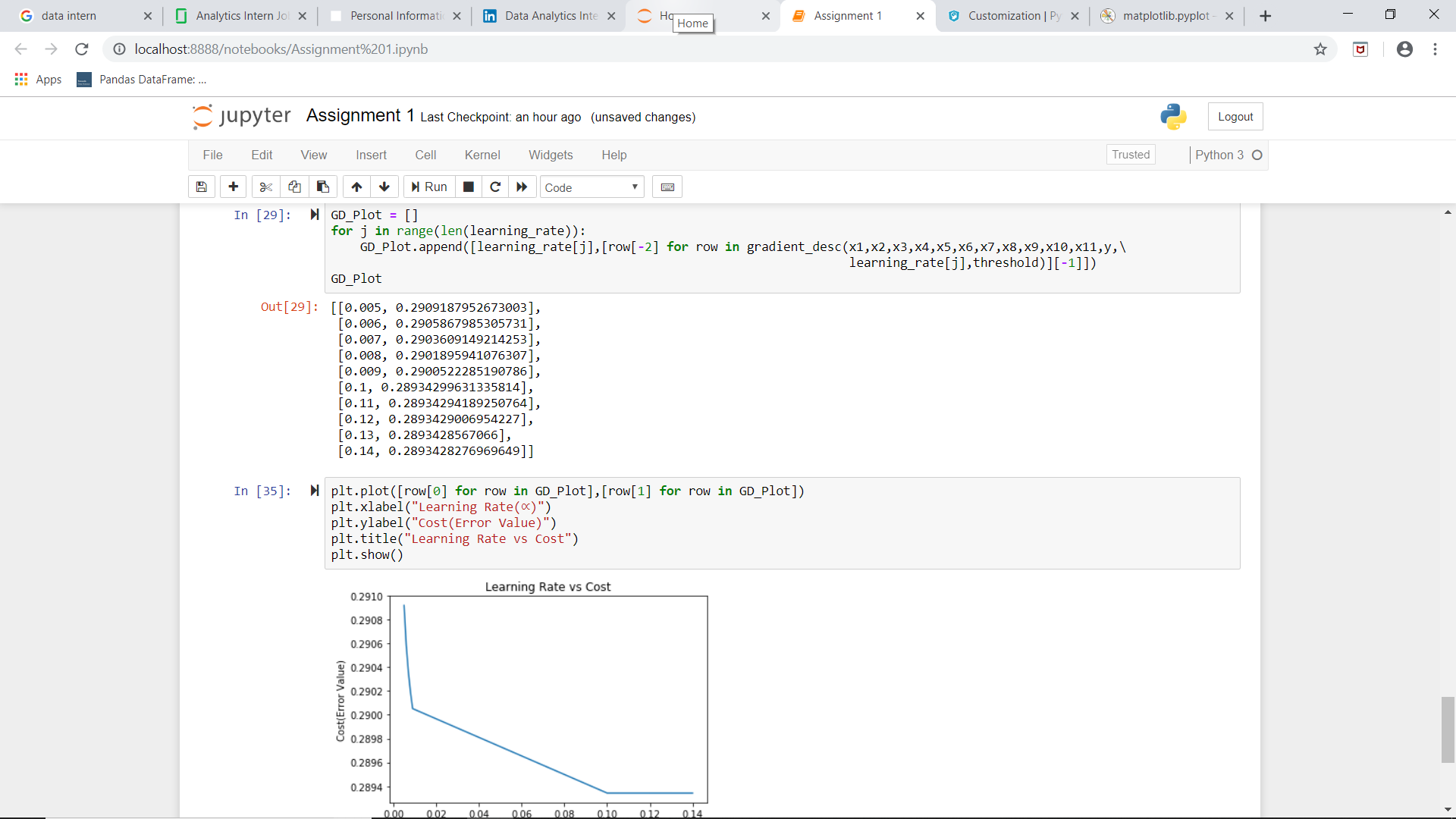
**Train Data**

For experiment purpose, I picked 10 ∝ (alpha) values which are as follows:

0.005 0.006 0.007 0.008 0.009 0.1 0.11 0.12 0.13 0.14

Corresponding cost for above alpha values are:

|  |  |
| --- | --- |
| Learning rate (∝) | Cost (Error Value) |
| 0.005 | 0.2909187952673003 |
| 0.006 | 0.2905867985305731 |
| 0.007 | 0.2903609149214253 |
| 0.008 | 0.2901895941076307 |
| 0.009 | 0.2900522285190786 |
| 0.1 | 0.28934299631335814 |
| 0.11 | 0.28934294189250764 |
| 0.12 | 0.2893429006954227 |
| 0.13 | 0.2893428567066 |
| 0.14 | 0.2893428276969649 |



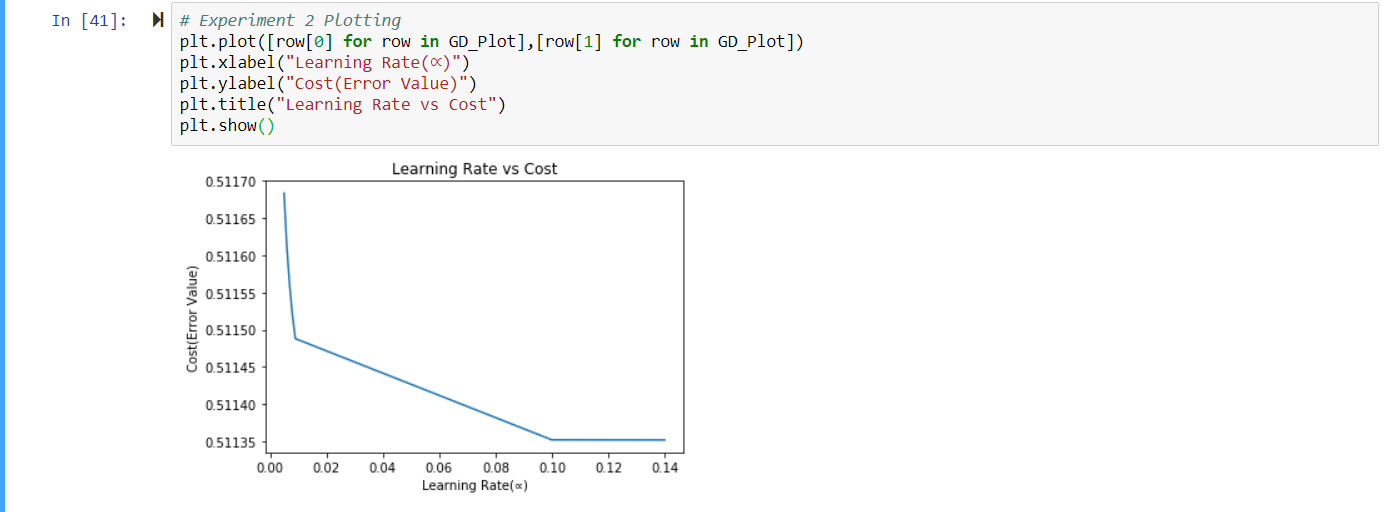
**Test Data**

For experiment purpose, I picked 10 ∝ (alpha) values which are as follows:

0.005 0.006 0.007 0.008 0.009 0.1 0.11 0.12 0.13 0.14

Corresponding cost for above alpha values is:

|  |  |
| --- | --- |
| Learning rate (∝) | Cost (Error Value) |
| 0.005 | 0.5116830433027897 |
| 0.006 | 0.5116110582365562 |
| 0.007 | 0.5115584815770934 |
| 0.008 | 0.5115187786573091 |
| 0.009 | 0.5114879953982685 |
| 0.1 | 0.5113521253321839 |
| 0.11 | 0.5113520777086473 |
| 0.12 | 0.5113520359042655 |
| 0.13 | 0.5113519976365848 |
| 0.14 | 0.5113519639824794 |



The best Learning rate (∝) is 0.10 for both test and train data because, as provided above in graph and matrix, the cost is almost same after 0.10. We can observe straight line stating that cost is not drastically and hence best learning rate

Experimented with various thresholds for convergence and Plotted error results for train and test sets as a function of threshold and describe how varying the threshold affects error.

Learning rate is 0.1

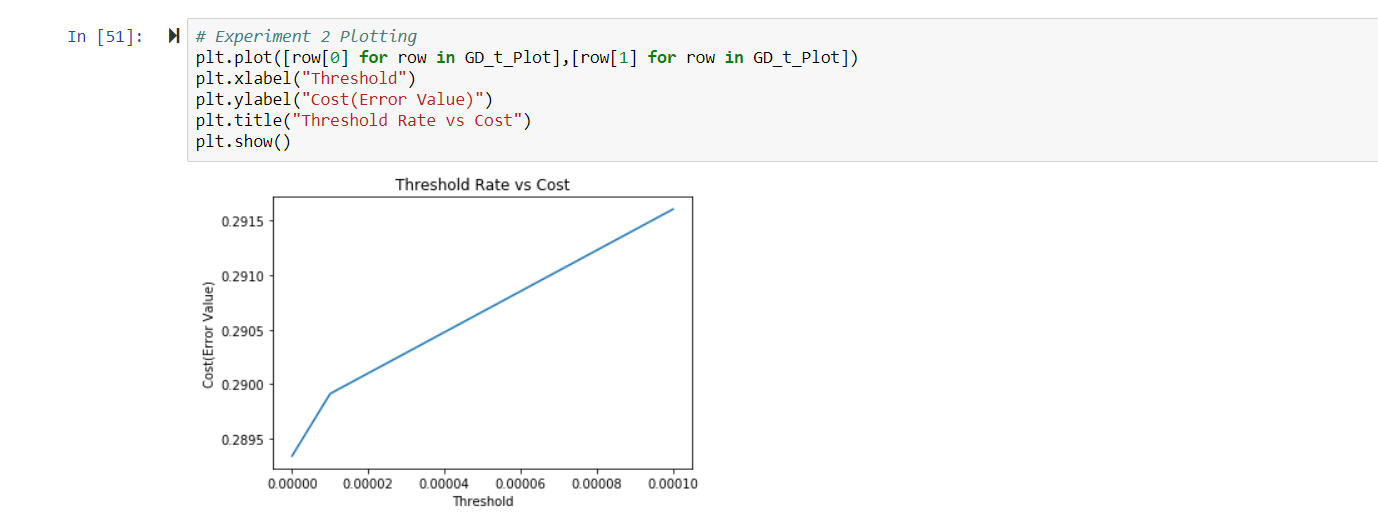
**Train Data**

For experiment purpose, I picked 5 threshold values which are as follows:

0.00001 0.000001 0.0000001 0.00000001

Corresponding cost for above threshold values are:

|  |  |
| --- | --- |
| Threshold | Cost (Error Value) |
| 0.0001 | 0.2916043788641238 |
| 1e-05 | 0.2899138485778441 |
| 1e-06 | 0.2894005303648134 |
| 1e-07 | 0.28934816162473065 |
| 1e-08 | 0.28934299631335814 |



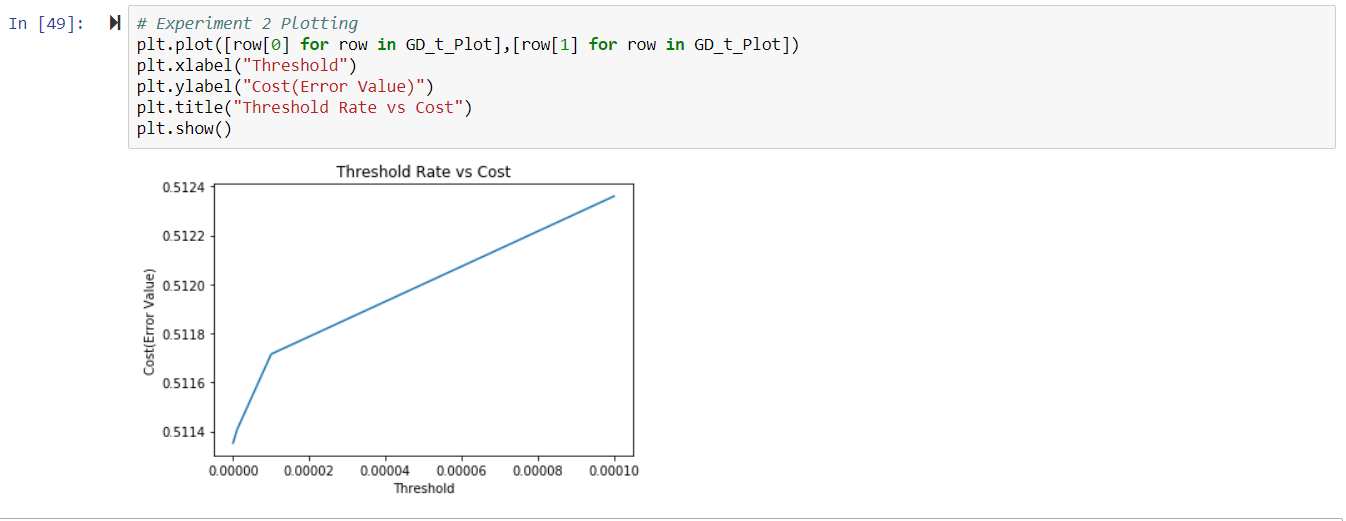
**Test Data**

For experiment purpose, I picked 5 threshold values which are as follows:

0.00001 0.000001 0.0000001 0.00000001

Corresponding cost for above threshold values are:

|  |  |
| --- | --- |
| Threshold | Cost (Error Value) |
| 0.0001 | 0.5123596305257354 |
| 1e-05 | 0.5117150847696308 |
| 1e-06 | 0.5114046571499172 |
| 1e-07 | 0.5113569593095023 |
| 1e-08 | 0.5113521253321839 |



The best threshold is 1e-08 because cost is minimum for this threshold. The plot for threshold vs iteration is: Blue line is for train data and Green line is for test data .

## Experimented with five random to predict the output

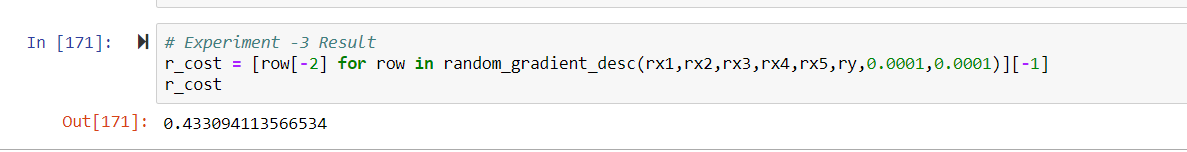
**Train Data**

For Learning rate = 0.0001 and threshold = 0.0001,

the cost is 0.433094113566534

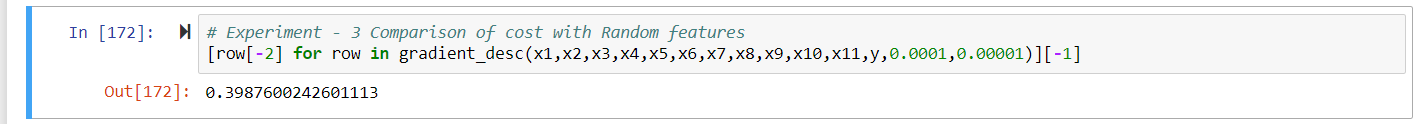
For 5 random variables which are as follows:

* Base DateTime weekday1
* H Local
* D15
* Base Time
* Post published weekday1



Whereas, original set of features, the cost is 0.3987600242601113 where original features are:

* Page Popularity/likes
* Page Checkins
* Page Category
* CC2
* CC3
* CC4
* Post Share Count
* H Local
* Base Time
* D5
* D10
* Target Variable



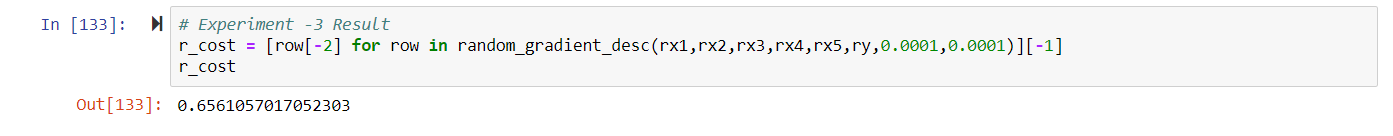
**Test Data**

For Learning rate = 0.0001 and threshold = 0.0001,

the cost is 0.6561057017052303

For 5 random variables which are as follows:

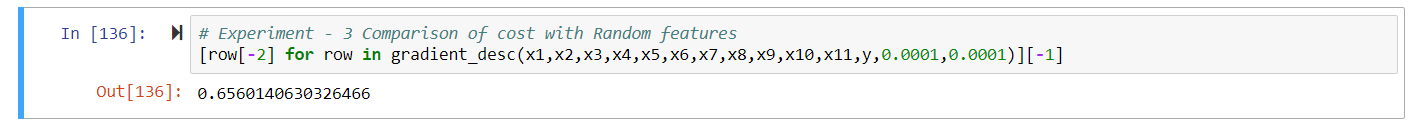
* Base DateTime weekday1
* H Local
* D15
* Base Time
* Post published weekday1



Whereas, original set of features, the cost is 0.6560140630326466

where original features are:

* Page Popularity/likes
* Page Checkins
* Page Category
* CC2
* CC3
* CC4
* Post Share Count
* H Local
* Base Time
* D5
* D10
* Target Variable



## Experimented with five features that seems to be best suited to predict the output

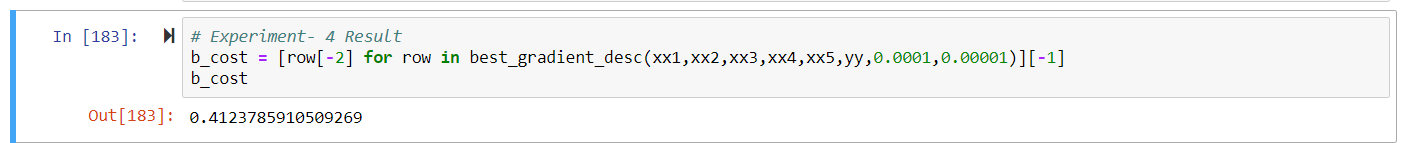
**Train Data**

For Learning rate = 0.0001 and threshold = 0.00001,

the cost is 0.4123785910509269

For 5 random variables which are as follows:

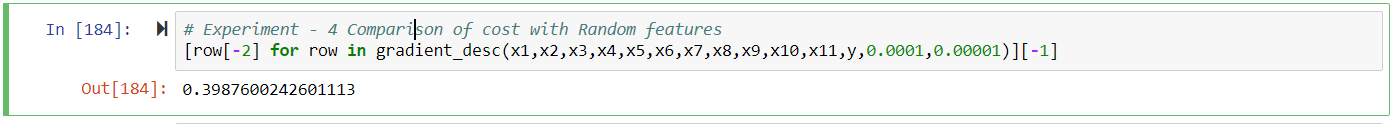
* D6
* D8
* D21
* CC2
* CC5



Whereas, original set of features, the cost is 0.3987600242601113 where

original features are:

* Page Popularity/likes
* Page Checkins
* Page Category
* CC2
* CC3
* CC4
* Post Share Count
* H Local
* Base Time
* D5
* D10



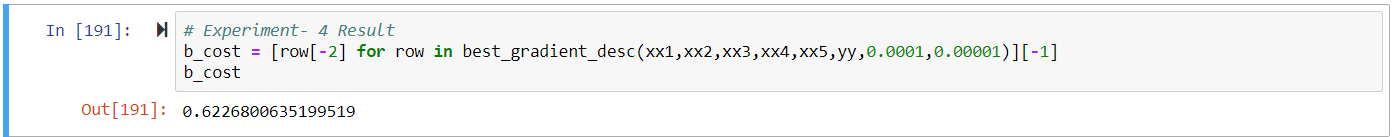
**Test Data**

For Learning rate = 0.0001 and threshold = 0.0001,

the cost is 0.6226800635199519

For 5 random variables which are as follows:

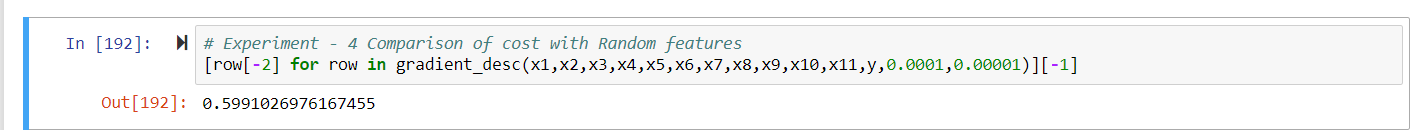
* D6
* D8
* D21
* CC2
* CC5



Whereas, original set of features, the cost is 0.5991026976167455

where original features are:

* Page Popularity/likes
* Page Checkins
* Page Category
* CC2
* CC3
* CC4
* Post Share Count
* H Local
* Base Time
* D5
* D10
* Target Variable



The best 5 features I selected doesn’t gives me better cost result as opposed to expectations, but it does give better result than the random features. I picked the best 5 features by observing how each feature is correlated to target variable using correlation matrix. We have in total 54 features and not all features may not fully explain the target features. Some feature explains better than rest so, the random 5 I picked does not explain the target feature trend, thus cost is high as compared to best 5 I picked.

As far as the comparison between original model and best model is concerned, the original model does explain better than best model keeping threshold and learning rate fixed which might be due to the fact, it has more and some relevant features which all together explains the target variable better.

Conclusion and Recommendations

The Gradient Descent algorithm forms a convex shape when plotted which means there will be a minimum point. If we observe the first experiment, we observed that lowest learning rate gave the minimum cost, so we can say lower the learning rate lower the cost value.

But there is a disadvantage to this issue, if we keep very low earning rate, it’ll take lot of time to reach minimum point and we should avoid it. Also, if our learning rate is too high it will skip minimum point and cost will rise.

So, we should keep our learning rate as suitable as possible to make sure we reach minimum cost in a smaller number of iterations.

Our Experiment 1 states 0.10 as the best alpha (learning rate), since it gives lowest cost of all at 0.00000001 threshold. Also, this threshold gives the minimum cost with lowest number of iterations which we can observe in experiment 2.

In experiment 3 and 4, we compared 5 random features model, best 5 features model and original model, to check which model best predict number of comments. Initial interpretation suggested original model predicts best then best feature model and last random feature model by comparing cost of all 3 models.

Also, if we observe our all model separately for test and train data, we’ll observe train data cost is very small as compared to test data which means there is lot of variance in our model which can be fixed by increasing our complexity or increasing datapoints.

I believe comments, page category, page likes, derived values matter most for predicting the number of comments. To improve our result, I would explore other types of models like decision tree or neural networks to predict number of comments. Also, I can try polynomial function for prediction.