**FODS ASSIGNMENT 1 REPORT**

Group Members: -

2017A7PS0186H – Yash Jain

2017A7PS0180H – Siddharth Jain

2017A1PS1100H – Dharesh Vatsa

***Part A - Gradient Descent Method:***

1. Imported the values from the Given Dataset in an array named a.
2. Latitude, Longitude and Altitude value were normalised and normalised values were stored in another array named b.
3. Values of Threshold = 10 e -10 and step size n = 4 e -6 were defined.
4. Training Data consisted of first 70% of the given dataset.
5. Error Function (Cost Function) error was defined, which finds deviation of the obtained values from the given values using half of sum of squares.
6. Transition Function was error0 was defined, which finds partial derivatives of error function with respect to all the three w values.
7. Initialised the w values to 0, and a while loop was run until the difference of errors of two iterations was less than the threshold value.
8. Final W values were obtained.
9. To obtain a comparative measure we found R\*\*2 and RMSE Values, note that R\*\*2 and RMSE values were found against the test part of the original dataset.

***Part B – Stochastic Gradient Descent Method:***

1. Imported the values from the Given Dataset in an array named a.
2. Latitude, Longitude and Altitude value were normalised and normalised values were stored in another array named b.
3. Value of step size n = 1 e -8 was defined.
4. A new variable j was defined which was number of times the transition function was used.
5. Training Data consisted of first 70% of the given dataset.
6. Error Function (Cost Function) error was defined, which finds deviation of the obtained values from the given values using half of sum of squares.
7. Transition Function was error0 was defined, which finds partial derivatives of error function with respect to all the three w values.
8. Initialised the w values to constants, and a for loop was used to iterate j times on the transition function.
9. Final W values were obtained.
10. To obtain a comparative measure we found R\*\*2 and RMSE Values, note that R\*\*2 and RMSE values were found against the test part of the original dataset.

***Part C - Gradient Descent Method along with regularization:***

1. Imported the values from the Given Dataset in an array named a.
2. Latitude, Longitude and Altitude value were normalised and normalised values were stored in another array named b.
3. Value of step size n = 1 e -8 was defined.
4. A new variable t was defined which was changed until it is the best value for the regularization coefficient.
5. Training Data consisted of first 70% of the given dataset.
6. Error Function (Cost Function) error for L1 Regularisation was defined, which finds deviation of the obtained values from the given values using half of sum of squares and adding lamda times summation of square of all three current w values.
7. Error Function (Cost Function) error for L2 Regularisation was defined, which finds deviation of the obtained values from the given values using half of sum of squares and adding lamda times summation of absolute values of all three current w values.
8. Transition Function was error0 was defined, which finds partial derivatives of error function with respect to all the three w values.
9. Initialised the w values to constants, and a for loop was used to iterate j times on the transition function.
10. Final W values were obtained.
11. To obtain a comparative measure we found R\*\*2 and RMSE Values, note that R\*\*2 and RMSE values were found against the test part of the original dataset.

***Part D - Normal Equations Method:***

1. Imported the values from the Given Dataset in an array named a.
2. Latitude, Longitude and Altitude value were normalised and normalised values were stored in another array named b.
3. Training Data consisted of first 70% of the given dataset.
4. A new Array c was made which had its first element 1(for the constant W0), second element latitude and the third was longitude.
5. Another array f was made for altitude values.
6. Final array e was defined which consists of the final w values, which were found using the following mathematical implementation:-

e=(c.transpose())  
e=np.matmul(e,c)  
e=np.linalg.inv(e)  
e=np.matmul(e,c.transpose())  
e=np.matmul(e,f)  
e=e.transpose()

1. Final W values were obtained.
2. To obtain a comparative measure we found R\*\*2 and RMSE Values, note that R\*\*2 and RMSE values were found against the test part of the original dataset.

**Comparative study of all methods**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Method name | W0 | W1 | W2 | RMSE | R2 |
| 1.Gradient Descent Method | 0.206892163934 | 0.095921198430719 | -0.09443981314843836 | 20.010540639612955 | 0.033385247131083595 |
| 2.Stochastic Gradient Descent Method | 0.09485098462568686 | 0.1416463634844515 | 0.2416044451747444 | 28.474031812183068 | -0.41974710933240256 |
| 3.L1 regularization | 0.20689209910911835 | 0.09592145483596351 | -0.09443997748469533 | 20.010540350744595 | 0.033385251140640726 |
| 4.L2 regularization | 0.20689209976662531 | 0.09592144504906809 | -0.09443996753767069 | 20.01054035336305 | 0.03338525044724239 |
| 5.Normal equations method | 0.20689191 | 0.09592221 | -0.09444047 | 20.010539525810355 | 0.033385263854323166 |

**Plot of loss over train set at an interval of 20 iterations-**

