```
> library(tidyverse)
>
> Data <- read_csv("UTD_DataSet_Final.csv")
Parsed with column specification:
cols(
  ID = col_double(),
  Feature_1 = col_double(),
  Feature_2 = col_double(),
  Feature_3 = col_double(),
  Feature_4 = col_double(),
  Feature_5 = col_double(),
  Feature_6 = col_double(),
  Feature_7 = col_double(),
  Feature_8 = col_double(),
  Feature_9 = col_double(),
  Feature_10 = col_double()
)
> glimpse(Data)
Observations: 3,782
Variables: 11
$ ID         <dbl> 1, 2, 4, 5, 7, 8, 9, 10, 11, 12, 14, 15, 16, 17, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28,
29,…
$ Feature_1  <dbl> 5200.062, 9520.000, 13685.000, 6426.000, 10353.000, 8330.000, 7259.000,
11250.000, 8500.051,…
$ Feature_2  <dbl> 7161.0, 5217.0, 6827.0, 2769.0, 10912.0, 4920.0, 7065.9, 21321.0, 6053.0,
3431.6, 9978.0, 97…
$ Feature_3  <dbl> 169744, 116888, 118415, 95160, 82558, 156921, 141358, 102768, 162684,
115758, 123502, 104458…
$ Feature_4  <dbl> 999, 999, 999, 999, 1390, 1896, 1997, 999, 1598, 1598, 999, 999, 999, 999,
999, 999, 999, 99…
$ Feature_5  <dbl> 17, 17, 14, 17, 7, 14, 17, 7, 7, 7, 7, 21, 7, 17, 18, 17, 7, 17, 17, 14, 7, 17, 14, 0,
14, 1…
$ Feature_6  <dbl> 25, 2, 21, 24, 32, 32, 25, 21, 32, 26, 26, 9, 27, 32, 11, 32, 11, 20, 32, 21, 32,
32, 24, 24…
$ Feature_7  <dbl> 26, 0, 134, 76, 183, 347, 37, 129, 183, 248, 241, 164, 203, 183, 387, 282,
181, 154, 347, 13…
$ Feature_8  <dbl> 6, 6, 6, 6, 6, 3, 3, 3, 3, 6, 6, 6, 6, 6, 6, 6, 6, 3, 3, 6, 3, 6, 3, 3, 6, 6, 3, 3, 6, 6, 6,…
$ Feature_9  <dbl> 11, 8, 9, 8, 7, 9, 10, 5, 7, 11, 5, 7, 7, 9, 4, 6, 12, 8, 13, 6, 8, 9, 12, 5, 10, 8, 4, 4,
9…
$ Feature_10 <dbl> 1850, 4400, 4850, 3100, 3400, 3500, 1700, 4050, 4100, 1250, 3200, 1950,
3150, 3050, 4650, 43…
> summary(Data)
      ID         Feature_1         Feature_2       Feature_3         Feature_4      Feature_5
 Min.   :   1   Min.   :    1.19   Min.   :   9   Min.   :     5   Min.   : 698   Min.   : 0.00
```

```
 1st Qu.:1144   1st Qu.: 4199.99   1st Qu.: 5301   1st Qu.:  24274   1st Qu.:1242   1st Qu.: 7.00
 Median :2282   Median : 6247.50   Median : 7782   Median :  81990   Median :1498   Median :
7.00
 Mean   :2260   Mean   : 7897.25   Mean   : 9448   Mean   :  89241   Mean   :1559   Mean   :10.53
 3rd Qu.:3392   3rd Qu.: 9758.00   3rd Qu.:11395   3rd Qu.: 135166   3rd Qu.:1896   3rd
Qu.:17.00
 Max.   :4450   Max.   :90756.30   Max.   :98335   Max.   :1058105   Max.   :5462   Max.   :21.00

  Feature_6      Feature_7      Feature_8      Feature_9       Feature_10
 Min.   : 0.0   Min.   :  0.0   Min.   :0.000   Min.   : 0.000   Min.   :  125
 1st Qu.: 9.0   1st Qu.: 87.0   1st Qu.:3.000   1st Qu.: 2.000   1st Qu.: 1800
 Median :24.0   Median :169.0   Median :6.000   Median : 7.000   Median : 3050
 Mean   :20.1   Mean   :181.2   Mean   :5.025   Mean   : 6.502   Mean   : 4227
 3rd Qu.:28.0   3rd Qu.:275.0   3rd Qu.:6.000   3rd Qu.:10.000   3rd Qu.: 5400
 Max.   :33.0   Max.   :388.0   Max.   :7.000   Max.   :25.000   Max.   :51500
                                                                 NA's   :382
> names(Data)
 [1] "ID"         "Feature_1"  "Feature_2"  "Feature_3"  "Feature_4"  "Feature_5"  "Feature_6"
"Feature_7"
 [9] "Feature_8"  "Feature_9"  "Feature_10"
> head(Data)
# A tibble: 6 x 11
     ID Feature_1 Feature_2 Feature_3 Feature_4 Feature_5 Feature_6 Feature_7 Feature_8
Feature_9 Feature_10
  <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>     <dbl>
1     1    5200.     7161   169744      999       17       25       26        6       11      1850
2     2    9520      5217   116888      999       17        2        0        6        8      4400
3     4   13685      6827   118415      999       14       21      134        6        9      4850
4     5    6426      2769    95160      999       17       24       76        6        8      3100
5     7   10353     10912    82558     1390        7       32      183        6        7      3400
6     8    8330      4920   156921     1896       14       32      347        3        9      3500
>
> Data$ID <- NULL
>
> Data$Feature_10[is.na(Data$Feature_10)] <- round(mean(Data$Feature_10, na.rm = TRUE))
>
> summary(Data)
  Feature_1        Feature_2       Feature_3        Feature_4      Feature_5      Feature_6
 Min.   :   1.19   Min.   :   9   Min.   :    5   Min.   : 698   Min.   : 0.00   Min.   : 0.0
 1st Qu.: 4199.99   1st Qu.: 5301   1st Qu.:  24274   1st Qu.:1242   1st Qu.: 7.00   1st Qu.: 9.0
 Median : 6247.50   Median : 7782   Median :  81990   Median :1498   Median : 7.00   Median
:24.0
 Mean   : 7897.25   Mean   : 9448   Mean   :  89241   Mean   :1559   Mean   :10.53   Mean   :20.1
 3rd Qu.: 9758.00   3rd Qu.:11395   3rd Qu.: 135166   3rd Qu.:1896   3rd Qu.:17.00   3rd Qu.:28.0
```

```
 Max.   :90756.30   Max.   :98335   Max.   :1058105   Max.   :5462   Max.   :21.00   Max.   :33.0
  Feature_7      Feature_8      Feature_9       Feature_10
 Min.   :  0.0   Min.   :0.000   Min.   : 0.000   Min.   :  125
 1st Qu.: 87.0   1st Qu.:3.000   1st Qu.: 2.000   1st Qu.: 1900
 Median :169.0   Median :6.000   Median : 7.000   Median : 3450
 Mean   :181.2   Mean   :5.025   Mean   : 6.502   Mean   : 4227
 3rd Qu.:275.0   3rd Qu.:6.000   3rd Qu.:10.000   3rd Qu.: 4900
 Max.   :388.0   Max.   :7.000   Max.   :25.000   Max.   :51500
>
> library(corrplot)
> M<-cor(Data)
> corrplot(M, method="circle")
>
> #Individual correlations
> cor(Data_train$Feature_10, Data_train$Feature_1, method = "pearson")
Error in is.data.frame(y) : object 'Data_train' not found
>
> #Converting the variables to factors
> Data$Feature_5 <- as.factor(Data$Feature_5)
> Data$Feature_6 <- as.factor(Data$Feature_6)
> Data$Feature_7 <- as.factor(Data$Feature_7)
> Data$Feature_8 <- as.factor(Data$Feature_8)
>
> #Splitting the training Data
> Data_train <- Data[1:3400,]
> summary(Data_train)
  Feature_1        Feature_2      Feature_3        Feature_4     Feature_5     Feature_6
Feature_7
 Min.   :   1.19   Min.   :   9   Min.   :    5   Min.   : 698   7    :1711   32    : 645   183   : 184
 1st Qu.: 4199.99   1st Qu.: 5270   1st Qu.:  23956   1st Qu.:1242   17    : 874   24    : 478   76    :
135
 Median : 6181.60   Median : 7729   Median :  81056   Median :1498   14    : 458   20    : 245
138   : 102
 Mean   : 7778.87   Mean   : 9386   Mean   :  88538   Mean   :1556   0     : 136   9     : 227   54    :
86
 3rd Qu.: 9599.97   3rd Qu.:11270   3rd Qu.: 134143   3rd Qu.:1896   8     :  81   1     : 206   164
:  83
 Max.   :90756.30   Max.   :98335   Max.   :1058105   Max.   :5462   18    :  29   28    : 175   282   :
82
                                                      (Other): 111   (Other):1424   (Other):2728
  Feature_8      Feature_9       Feature_10
 6     :2284   Min.   : 0.000   Min.   :  125
 3     :1091   1st Qu.: 2.000   1st Qu.: 1800
 0     :   9   Median : 7.000   Median : 3050
```

```
 5     :  6   Mean   : 6.444   Mean   : 4227
 2     :  3   3rd Qu.:10.000   3rd Qu.: 5400
 7     :  3   Max.   :25.000   Max.   :51500
 (Other):  4
> glimpse(Data_train)
Observations: 3,400
Variables: 10
$ Feature_1  <dbl> 5200.062, 9520.000, 13685.000, 6426.000, 10353.000, 8330.000, 7259.000,
11250.000, 8500.051,…
$ Feature_2  <dbl> 7161.0, 5217.0, 6827.0, 2769.0, 10912.0, 4920.0, 7065.9, 21321.0, 6053.0,
3431.6, 9978.0, 97…
$ Feature_3  <dbl> 169744, 116888, 118415, 95160, 82558, 156921, 141358, 102768, 162684,
115758, 123502, 104458…
$ Feature_4  <dbl> 999, 999, 999, 999, 1390, 1896, 1997, 999, 1598, 1598, 999, 999, 999, 999,
999, 999, 999, 99…
$ Feature_5  <fct> 17, 17, 14, 17, 7, 14, 17, 7, 7, 7, 7, 21, 7, 17, 18, 17, 7, 17, 17, 14, 7, 17, 14, 0,
14, 1…
$ Feature_6  <fct> 25, 2, 21, 24, 32, 32, 25, 21, 32, 26, 26, 9, 27, 32, 11, 32, 11, 20, 32, 21, 32,
32, 24, 24…
$ Feature_7  <fct> 26, 0, 134, 76, 183, 347, 37, 129, 183, 248, 241, 164, 203, 183, 387, 282, 181,
154, 347, 13…
$ Feature_8  <fct> 6, 6, 6, 6, 6, 3, 3, 3, 3, 6, 6, 6, 6, 6, 6, 6, 6, 3, 3, 6, 3, 6, 3, 3, 6, 6, 3, 3, 6, 6, 6,…
$ Feature_9  <dbl> 11, 8, 9, 8, 7, 9, 10, 5, 7, 11, 5, 7, 7, 9, 4, 6, 12, 8, 13, 6, 8, 9, 12, 5, 10, 8, 4, 4,
9…
$ Feature_10 <dbl> 1850, 4400, 4850, 3100, 3400, 3500, 1700, 4050, 4100, 1250, 3200, 1950,
3150, 3050, 4650, 43…
> head(Data_train)
# A tibble: 6 x 10
  Feature_1 Feature_2 Feature_3 Feature_4 Feature_5 Feature_6 Feature_7 Feature_8
Feature_9 Feature_10
      <dbl>     <dbl>     <dbl>     <dbl> <fct>     <fct>     <fct>     <fct>
<dbl>      <dbl>
1    5200.      7161    169744       999 17        25        26        6
11       1850
2    9520       5217    116888       999 17        2         0         6
8       4400
3   13685       6827    118415       999 14        21        134       6
9       4850
4    6426       2769     95160       999 17        24        76        6
8       3100
5   10353      10912     82558      1390 7         32        183       6
7       3400
6    8330       4920    156921      1896 14        32        347       3
9       3500
> Data_test <- Data[3401:3782,]
>
> #Feels Outlier
> Data_train[(Data_train$Feature_10==51500),]
# A tibble: 1 x 10
  Feature_1 Feature_2 Feature_3 Feature_4 Feature_5 Feature_6 Feature_7 Feature_8
Feature_9 Feature_10
```

```
    <dbl>    <dbl>    <dbl>    <dbl> <fct>    <fct>    <fct>    <fct>       <dbl>     <dbl>
1     7250    26100.     2514     2967 7        1        62       3            0     51500
> #plot independent var
> hist(Data_train$Feature_10)
> boxplot(Data_train$Feature_10)
> #Since data is left skewed, we can take log to have a normal distribution
> hist(log(Data_train$Feature_10))
> boxplot(log(Data_train$Feature_10))
> #XGBoost ----
> library(xgboost)
> library(Matrix)
>
> #creating sparse matrix for applying XGBoost
> sparse <- sparse.model.matrix(Feature_10 ~ ., data = Data)[,-1]
> nrow(sparse)
[1] 3782
> head(sparse)
6 x 454 sparse Matrix of class "dgCMatrix"
  [[ suppressing 51 column names ‘Feature_1’, ‘Feature_2’, ‘Feature_3’ … ]]

1  5200.062  7161 169744  999 . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . . . . .
2  9520.000  5217 116888  999 . . . . . . . . . . . . . . . 1 . . . . . 1 . . . . . . . . . . . . . . . . . .
3 13685.000  6827 118415  999 . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . . . . . . . 1
4  6426.000  2769  95160  999 . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . . . . .
5 10353.000 10912  82558 1390 . . . . . . 1 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
6  8330.000  4920 156921 1896 . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . . . . . . . .

1 . . . 1 . ......
2 . . . . . ......
3 . . . . . ......
4 . . 1 . . ......
5 . . . . . ......
6 . . . . . ......

 .....suppressing 403 columns in show(); maybe adjust 'options(max.print= *, width = *)'
 ..............................
> sparse_matrix <- sparse[1:3400,]
> sparse_test <- sparse[3401:3782,]
> nrow(sparse_test)
[1] 382
>
>
> bst <- xgboost(data = sparse_matrix, label = (Data_train$Feature_10), max_depth = 40,
+               eta = 0.05, nthread = 2, nrounds = 300)
```

[1] train-rmse:5366.545410
[2] train-rmse:5128.309570
[3] train-rmse:4902.312988
[4] train-rmse:4687.156738
[5] train-rmse:4483.240723
[6] train-rmse:4289.467773
[7] train-rmse:4105.696777
[8] train-rmse:3930.093018
[9] train-rmse:3763.229004
[10] train-rmse:3604.751953
[11] train-rmse:3454.135498
[12] train-rmse:3310.912109
[13] train-rmse:3174.426270
[14] train-rmse:3043.857666
[15] train-rmse:2920.006836
[16] train-rmse:2802.018799
[17] train-rmse:2689.365234
[18] train-rmse:2581.636963
[19] train-rmse:2478.532227
[20] train-rmse:2380.761475
[21] train-rmse:2287.588867
[22] train-rmse:2198.641602
[23] train-rmse:2114.208008
[24] train-rmse:2033.161011
[25] train-rmse:1955.290894
[26] train-rmse:1881.385620
[27] train-rmse:1810.744263
[28] train-rmse:1743.063477
[29] train-rmse:1678.609985
[30] train-rmse:1616.596924
[31] train-rmse:1557.188477
[32] train-rmse:1500.396484
[33] train-rmse:1446.346191
[34] train-rmse:1394.002930
[35] train-rmse:1344.101807
[36] train-rmse:1296.270142
[37] train-rmse:1250.587402
[38] train-rmse:1206.455566
[39] train-rmse:1164.688477
[40] train-rmse:1123.476074
[41] train-rmse:1083.965698
[42] train-rmse:1045.906494
[43] train-rmse:1009.592590
[44] train-rmse:974.954712

[45] train-rmse:942.128235
[46] train-rmse:910.753479
[47] train-rmse:880.640381
[48] train-rmse:851.664062
[49] train-rmse:823.814697
[50] train-rmse:797.038330
[51] train-rmse:771.182556
[52] train-rmse:746.357178
[53] train-rmse:722.511597
[54] train-rmse:699.677979
[55] train-rmse:677.473450
[56] train-rmse:656.229065
[57] train-rmse:635.138977
[58] train-rmse:615.093506
[59] train-rmse:595.885071
[60] train-rmse:577.367065
[61] train-rmse:559.658875
[62] train-rmse:542.511414
[63] train-rmse:526.077881
[64] train-rmse:510.184570
[65] train-rmse:494.945251
[66] train-rmse:480.283569
[67] train-rmse:466.154541
[68] train-rmse:452.518341
[69] train-rmse:439.370300
[70] train-rmse:426.804413
[71] train-rmse:414.662994
[72] train-rmse:402.923584
[73] train-rmse:391.732758
[74] train-rmse:380.930817
[75] train-rmse:370.442749
[76] train-rmse:360.307312
[77] train-rmse:350.534729
[78] train-rmse:341.056427
[79] train-rmse:331.866272
[80] train-rmse:323.045776
[81] train-rmse:314.562347
[82] train-rmse:306.451538
[83] train-rmse:298.634033
[84] train-rmse:291.097778
[85] train-rmse:283.849182
[86] train-rmse:276.863434
[87] train-rmse:270.117950
[88] train-rmse:263.615570

[89] train-rmse:257.348022
[90] train-rmse:251.330109
[91] train-rmse:245.522842
[92] train-rmse:239.964783
[93] train-rmse:234.587723
[94] train-rmse:229.386917
[95] train-rmse:224.364395
[96] train-rmse:219.526306
[97] train-rmse:214.919968
[98] train-rmse:210.480377
[99] train-rmse:206.207825
[100] train-rmse:202.036575
[101] train-rmse:198.067535
[102] train-rmse:194.226318
[103] train-rmse:190.555206
[104] train-rmse:187.016876
[105] train-rmse:183.650467
[106] train-rmse:180.387848
[107] train-rmse:177.257950
[108] train-rmse:174.255066
[109] train-rmse:171.388916
[110] train-rmse:168.609192
[111] train-rmse:165.949432
[112] train-rmse:163.436783
[113] train-rmse:161.008575
[114] train-rmse:158.697357
[115] train-rmse:156.462830
[116] train-rmse:154.351471
[117] train-rmse:152.322525
[118] train-rmse:150.382858
[119] train-rmse:148.543961
[120] train-rmse:146.739624
[121] train-rmse:145.026489
[122] train-rmse:143.391632
[123] train-rmse:141.823120
[124] train-rmse:140.318909
[125] train-rmse:138.886780
[126] train-rmse:137.514053
[127] train-rmse:136.198212
[128] train-rmse:134.956314
[129] train-rmse:133.762909
[130] train-rmse:132.627563
[131] train-rmse:131.542709
[132] train-rmse:130.508072

[133] train-rmse:129.520416
[134] train-rmse:128.574356
[135] train-rmse:127.674957
[136] train-rmse:126.818405
[137] train-rmse:125.999046
[138] train-rmse:125.222443
[139] train-rmse:124.483620
[140] train-rmse:123.774773
[141] train-rmse:123.092056
[142] train-rmse:122.444962
[143] train-rmse:121.829140
[144] train-rmse:121.243202
[145] train-rmse:120.687759
[146] train-rmse:120.159904
[147] train-rmse:119.654175
[148] train-rmse:119.176491
[149] train-rmse:118.722176
[150] train-rmse:118.290108
[151] train-rmse:117.878380
[152] train-rmse:117.486610
[153] train-rmse:117.113213
[154] train-rmse:116.758125
[155] train-rmse:116.420799
[156] train-rmse:116.099335
[157] train-rmse:115.793793
[158] train-rmse:115.502716
[159] train-rmse:115.226654
[160] train-rmse:114.963287
[161] train-rmse:114.713318
[162] train-rmse:114.475090
[163] train-rmse:114.244270
[164] train-rmse:114.026947
[165] train-rmse:113.822678
[166] train-rmse:113.625244
[167] train-rmse:113.440407
[168] train-rmse:113.261848
[169] train-rmse:113.092117
[170] train-rmse:112.931656
[171] train-rmse:112.779617
[172] train-rmse:112.635330
[173] train-rmse:112.497849
[174] train-rmse:112.368111
[175] train-rmse:112.245430
[176] train-rmse:112.128380

[177] train-rmse:112.018257
[178] train-rmse:111.910904
[179] train-rmse:111.809570
[180] train-rmse:111.715279
[181] train-rmse:111.624062
[182] train-rmse:111.539307
[183] train-rmse:111.456711
[184] train-rmse:111.379089
[185] train-rmse:111.304787
[186] train-rmse:111.234169
[187] train-rmse:111.166855
[188] train-rmse:111.104393
[189] train-rmse:111.043472
[190] train-rmse:110.985985
[191] train-rmse:110.931526
[192] train-rmse:110.879929
[193] train-rmse:110.830925
[194] train-rmse:110.784492
[195] train-rmse:110.740501
[196] train-rmse:110.698959
[197] train-rmse:110.659676
[198] train-rmse:110.622177
[199] train-rmse:110.586571
[200] train-rmse:110.552666
[201] train-rmse:110.520538
[202] train-rmse:110.490005
[203] train-rmse:110.460770
[204] train-rmse:110.433197
[205] train-rmse:110.407089
[206] train-rmse:110.382156
[207] train-rmse:110.358665
[208] train-rmse:110.336288
[209] train-rmse:110.315170
[210] train-rmse:110.295166
[211] train-rmse:110.276131
[212] train-rmse:110.258141
[213] train-rmse:110.240913
[214] train-rmse:110.224594
[215] train-rmse:110.209084
[216] train-rmse:110.194336
[217] train-rmse:110.180412
[218] train-rmse:110.167152
[219] train-rmse:110.154495
[220] train-rmse:110.142418

[221] train-rmse:110.131058
[222] train-rmse:110.120186
[223] train-rmse:110.109825
[224] train-rmse:110.100037
[225] train-rmse:110.090630
[226] train-rmse:110.081902
[227] train-rmse:110.073387
[228] train-rmse:110.065453
[229] train-rmse:110.057755
[230] train-rmse:110.050278
[231] train-rmse:110.043144
[232] train-rmse:110.036392
[233] train-rmse:110.029831
[234] train-rmse:110.023804
[235] train-rmse:110.017990
[236] train-rmse:110.012360
[237] train-rmse:110.007126
[238] train-rmse:110.002174
[239] train-rmse:109.997490
[240] train-rmse:109.993019
[241] train-rmse:109.988892
[242] train-rmse:109.984718
[243] train-rmse:109.980843
[244] train-rmse:109.977043
[245] train-rmse:109.973564
[246] train-rmse:109.970276
[247] train-rmse:109.967133
[248] train-rmse:109.964233
[249] train-rmse:109.961418
[250] train-rmse:109.958687
[251] train-rmse:109.956123
[252] train-rmse:109.953644
[253] train-rmse:109.951340
[254] train-rmse:109.949081
[255] train-rmse:109.946976
[256] train-rmse:109.944992
[257] train-rmse:109.942986
[258] train-rmse:109.941101
[259] train-rmse:109.939354
[260] train-rmse:109.937675
[261] train-rmse:109.936119
[262] train-rmse:109.934669
[263] train-rmse:109.933281
[264] train-rmse:109.931900

```
[265]   train-rmse:109.930611
[266]   train-rmse:109.929306
[267]   train-rmse:109.928169
[268]   train-rmse:109.927071
[269]   train-rmse:109.926048
[270]   train-rmse:109.925056
[271]   train-rmse:109.924118
[272]   train-rmse:109.923195
[273]   train-rmse:109.922348
[274]   train-rmse:109.921494
[275]   train-rmse:109.920670
[276]   train-rmse:109.919945
[277]   train-rmse:109.919197
[278]   train-rmse:109.918518
[279]   train-rmse:109.917862
[280]   train-rmse:109.917236
[281]   train-rmse:109.916649
[282]   train-rmse:109.916092
[283]   train-rmse:109.915588
[284]   train-rmse:109.915039
[285]   train-rmse:109.914566
[286]   train-rmse:109.914116
[287]   train-rmse:109.913635
[288]   train-rmse:109.913208
[289]   train-rmse:109.912796
[290]   train-rmse:109.912460
[291]   train-rmse:109.912064
[292]   train-rmse:109.911728
[293]   train-rmse:109.911400
[294]   train-rmse:109.911087
[295]   train-rmse:109.910774
[296]   train-rmse:109.910522
[297]   train-rmse:109.910233
[298]   train-rmse:109.909935
[299]   train-rmse:109.909660
[300]   train-rmse:109.909439
>
> importance <- xgb.importance(feature_names = colnames(sparse_matrix), model = bst)
> head(importance)
     Feature       Gain      Cover   Frequency
1:  Feature_9 0.49895893 0.074665179 0.036572550
2:  Feature_4 0.25068276 0.045381302 0.064269533
3:  Feature_1 0.13002043 0.118131719 0.273054709
4:  Feature_2 0.03279903 0.072697349 0.208543947
```

```
5:   Feature_3 0.01733145 0.077879896 0.153288618
6: Feature_620 0.01365999 0.008527617 0.003912348
>
> #Validating predictions
> pred_Y <- predict(bst, newdata=sparse_matrix)
> pred_Z <- round(exp(1)^pred_Y, digits=2)
> mean(abs(pred_Z - Data_train$Feature_10))
[1] Inf
>
> #Prediction using XGBoost
> pred <- predict(bst, newdata = sparse_test)
> Data_test$Feature_10 <- round(exp(1)^pred, digits=2)
>
>
> #creating the submission file ----
> Data_ID <- read_csv("UTD_DataSet_Final.csv")
Parsed with column specification:
cols(
  ID = col_double(),
  Feature_1 = col_double(),
  Feature_2 = col_double(),
  Feature_3 = col_double(),
  Feature_4 = col_double(),
  Feature_5 = col_double(),
  Feature_6 = col_double(),
  Feature_7 = col_double(),
  Feature_8 = col_double(),
  Feature_9 = col_double(),
  Feature_10 = col_double()
)
> Data_ID <- Data_ID[3401:3782,]
> Data_test$ID <- Data_ID[,"ID"]
>
> Submission <- data.frame(ID = Data_test$ID, Feature_10 = Data_test$Feature_10)
> glimpse(Submission)
Observations: 382
Variables: 2
$ ID        <dbl> 4024, 4025, 4026, 4027, 4028, 4029, 4030, 4031, 4032, 4033, 4034, 4035, 4036,
4037, 4038, 40…
$ Feature_10 <dbl> Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, Inf, In…
> write_csv(Submission, "submission.csv")
```