Project Results

Group- P7

4/26/2018

First we load the data and create the features in the testing data.

## Here we load the test data along with their prediction  
library(lubridate)

##   
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':  
##   
## date

## Loading test file with predicted values   
test.pred = read.csv(file = 'test\_pred.csv', header = T, sep = ',')  
  
## Here we find the hour  
test.pred$click\_time = hour(test.pred$click\_time)  
  
## Here we find feature1 (combination of app and device)  
test.pred$feature1 = as.numeric(paste0(test.pred$app, test.pred$device))  
  
## Here we find feature2 (combination of app and os)  
test.pred$feature2 = as.numeric(paste0(test.pred$app, test.pred$os))  
  
## Here we find feature3 (combination of app and channel)  
test.pred$feature3 = as.numeric(paste0(test.pred$app, test.pred$channel))  
  
## Here we find feature4 (combination of device and os)  
test.pred$feature4 = as.numeric(paste0(test.pred$device, test.pred$os))  
  
## Here we find feature5 (combination of device and channel)  
test.pred$feature5 = as.numeric(paste0(test.pred$device, test.pred$channel))  
  
## Here we find feature6 (combination of os and channel)  
test.pred$feature6 = as.numeric(paste0(test.pred$os, test.pred$channel))

Since the predictions values are 0’s and 1’s, it seems logical to fit a logistic model to the data as shown below.

## logistic regression  
md = glm(pred ~ ip + app + device + os + channel + click\_time + feature1 + feature2 + feature3 + feature4 + feature5 + feature6, data = test.pred)  
summary(md)

##   
## Call:  
## glm(formula = pred ~ ip + app + device + os + channel + click\_time +   
## feature1 + feature2 + feature3 + feature4 + feature5 + feature6,   
## data = test.pred)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.8709 -0.0385 -0.0178 0.0067 1.5022   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.608e-02 1.382e-04 -478.294 < 2e-16 \*\*\*  
## ip -1.364e-10 8.787e-10 -0.155 0.877   
## app 1.030e-02 1.701e-05 605.951 < 2e-16 \*\*\*  
## device 1.038e-03 9.187e-05 11.297 < 2e-16 \*\*\*  
## os -6.088e-05 1.398e-05 -4.356 1.33e-05 \*\*\*  
## channel 6.960e-05 2.655e-07 262.169 < 2e-16 \*\*\*  
## click\_time 9.877e-04 8.643e-06 114.281 < 2e-16 \*\*\*  
## feature1 -1.071e-06 8.943e-09 -119.784 < 2e-16 \*\*\*  
## feature2 -2.281e-06 4.093e-08 -55.720 < 2e-16 \*\*\*  
## feature3 -5.916e-06 1.675e-08 -353.113 < 2e-16 \*\*\*  
## feature4 -7.167e-07 9.641e-09 -74.336 < 2e-16 \*\*\*  
## feature5 5.920e-08 9.178e-08 0.645 0.519   
## feature6 6.001e-07 1.396e-08 42.972 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 0.01973606)  
##   
## Null deviance: 434334 on 18790468 degrees of freedom  
## Residual deviance: 370850 on 18790456 degrees of freedom  
## AIC: -20433315  
##   
## Number of Fisher Scoring iterations: 2

From the results, we see ip and feature5 are not significant. Also note that the iteration between app, device, os and channel are important. We next only focus on prediction values equal to 1 with the goal of finding interesting interactions between app, device, os and channel.

## selecting only the observation that have predictions equal to 1  
test.pred.one = subset(test.pred, pred == 1)  
  
## Now lets see the Summary  
summary(test.pred.one)

## click\_id ip app device   
## Min. : 348 Min. : 0 Min. : 0.00 Min. : 0.00   
## 1st Qu.: 5819816 1st Qu.: 29072 1st Qu.: 19.00 1st Qu.: 0.00   
## Median :11767432 Median : 63228 Median : 19.00 Median : 0.00   
## Mean :10513353 Mean : 62373 Mean : 37.04 Mean : 29.18   
## 3rd Qu.:15270678 3rd Qu.: 95886 3rd Qu.: 35.00 3rd Qu.: 1.00   
## Max. :18790361 Max. :126413 Max. :521.00 Max. :3031.00   
## os channel click\_time pred   
## Min. : 0.00 Min. : 0.0 Min. : 4.00 Min. :1   
## 1st Qu.: 13.00 1st Qu.:213.0 1st Qu.: 5.00 1st Qu.:1   
## Median : 24.00 Median :343.0 Median :10.00 Median :1   
## Mean : 22.83 Mean :279.8 Mean :10.02 Mean :1   
## 3rd Qu.: 33.00 3rd Qu.:347.0 3rd Qu.:13.00 3rd Qu.:1   
## Max. :596.00 Max. :489.0 Max. :15.00 Max. :1   
## feature1 feature2 feature3 feature4   
## Min. : 0 Min. : 0 Min. : 101 Min. : 0   
## 1st Qu.: 190 1st Qu.: 1921 1st Qu.: 19213 1st Qu.: 24   
## Median : 190 Median : 1929 Median : 19347 Median : 38   
## Mean : 3325 Mean : 3352 Mean : 35435 Mean : 2571   
## 3rd Qu.: 573 3rd Qu.: 3225 3rd Qu.: 29347 3rd Qu.: 157   
## Max. :2523031 Max. :224153 Max. :521274 Max. :1848104   
## feature5 feature6   
## Min. : 0 Min. : 0   
## 1st Qu.: 343 1st Qu.: 13120   
## Median : 347 Median : 24213   
## Mean : 29402 Mean : 22386   
## 3rd Qu.: 1419 3rd Qu.: 33347   
## Max. :3031213 Max. :596101

## Feature1 - Here we start looking for interesting relationships in feature 1  
t1 = table(test.pred.one$feature1)  
t1 = t1[order(t1, decreasing = T)]  
  
## from the above the table we see that app = 19 and device = 0   
## has a lot of ones  
test.190 = subset(test.pred.one, feature1 == '190')  
  
## from above let's take top 10 results  
test.190[1:10,]

## click\_id ip app device os channel click\_time pred feature1  
## 424 423 4052 19 0 50 347 4 1 190  
## 558 557 46114 19 0 38 347 4 1 190  
## 1039 1038 72695 19 0 24 347 4 1 190  
## 1078 1077 30683 19 0 24 213 4 1 190  
## 1344 1342 48282 19 0 24 347 4 1 190  
## 1422 1421 4145 19 0 24 343 4 1 190  
## 1604 1603 96777 19 0 21 213 4 1 190  
## 2102 2101 69583 19 0 24 213 4 1 190  
## 2140 2139 100335 19 0 24 213 4 1 190  
## 2222 2221 36636 19 0 29 333 4 1 190  
## feature2 feature3 feature4 feature5 feature6  
## 424 1950 19347 50 347 50347  
## 558 1938 19347 38 347 38347  
## 1039 1924 19347 24 347 24347  
## 1078 1924 19213 24 213 24213  
## 1344 1924 19347 24 347 24347  
## 1422 1924 19343 24 343 24343  
## 1604 1921 19213 21 213 21213  
## 2102 1924 19213 24 213 24213  
## 2140 1924 19213 24 213 24213  
## 2222 1929 19333 29 333 29333

# Feature2 - Here we start looking for interesting relationships in feature 2  
t2 = table(test.pred.one$feature2)  
t2 = t2[order(t2, decreasing = T)]  
  
## from the above the table we see that app = 19 and os = 24   
## has a lot of ones  
test.1924 = subset(test.pred.one, feature2 == '1924')  
test.1924[1:10,]

## click\_id ip app device os channel click\_time pred feature1  
## 1039 1038 72695 19 0 24 347 4 1 190  
## 1078 1077 30683 19 0 24 213 4 1 190  
## 1344 1342 48282 19 0 24 347 4 1 190  
## 1422 1421 4145 19 0 24 343 4 1 190  
## 1643 1642 73124 19 21 24 101 4 1 1921  
## 2102 2101 69583 19 0 24 213 4 1 190  
## 2140 2139 100335 19 0 24 213 4 1 190  
## 2622 2621 65591 19 0 24 213 4 1 190  
## 2875 2874 100335 19 0 24 213 4 1 190  
## 2960 2959 7874 19 0 24 213 4 1 190  
## feature2 feature3 feature4 feature5 feature6  
## 1039 1924 19347 24 347 24347  
## 1078 1924 19213 24 213 24213  
## 1344 1924 19347 24 347 24347  
## 1422 1924 19343 24 343 24343  
## 1643 1924 19101 2124 21101 24101  
## 2102 1924 19213 24 213 24213  
## 2140 1924 19213 24 213 24213  
## 2622 1924 19213 24 213 24213  
## 2875 1924 19213 24 213 24213  
## 2960 1924 19213 24 213 24213

# Feature3 - Here we start looking for interesting relationships in feature 3  
t3 = table(test.pred.one$feature3)  
t3 = t3[order(t3, decreasing = T)]  
  
## from the above the table we see that app = 19 and channel = 347   
## has a lot of ones  
test.19347 = subset(test.pred.one, feature3 == '19347')  
test.19347[1:10,]

## click\_id ip app device os channel click\_time pred feature1  
## 424 423 4052 19 0 50 347 4 1 190  
## 558 557 46114 19 0 38 347 4 1 190  
## 1039 1038 72695 19 0 24 347 4 1 190  
## 1344 1342 48282 19 0 24 347 4 1 190  
## 3336 3335 87604 19 0 0 347 4 1 190  
## 3791 3790 5147 19 0 29 347 4 1 190  
## 4188 4187 11290 19 0 24 347 4 1 190  
## 5959 5959 100959 19 0 21 347 4 1 190  
## 6370 6369 46970 19 0 0 347 4 1 190  
## 6818 6818 59472 19 0 24 347 4 1 190  
## feature2 feature3 feature4 feature5 feature6  
## 424 1950 19347 50 347 50347  
## 558 1938 19347 38 347 38347  
## 1039 1924 19347 24 347 24347  
## 1344 1924 19347 24 347 24347  
## 3336 190 19347 0 347 347  
## 3791 1929 19347 29 347 29347  
## 4188 1924 19347 24 347 24347  
## 5959 1921 19347 21 347 21347  
## 6370 190 19347 0 347 347  
## 6818 1924 19347 24 347 24347

# Feature4 - Here we start looking for interesting relationships in feature 4  
t4 = table(test.pred.one$feature4)  
t4 = t4[order(t4, decreasing = T)]  
  
## from the above the table we see that device = 0 and os = 24   
## has a lot of ones  
test.24 = subset(test.pred.one, feature4 == '24')  
test.24[1:10,]

## click\_id ip app device os channel click\_time pred feature1  
## 1039 1038 72695 19 0 24 347 4 1 190  
## 1078 1077 30683 19 0 24 213 4 1 190  
## 1344 1342 48282 19 0 24 347 4 1 190  
## 1422 1421 4145 19 0 24 343 4 1 190  
## 1982 1981 53454 83 0 24 171 4 1 830  
## 2102 2101 69583 19 0 24 213 4 1 190  
## 2140 2139 100335 19 0 24 213 4 1 190  
## 2622 2621 65591 19 0 24 213 4 1 190  
## 2875 2874 100335 19 0 24 213 4 1 190  
## 2960 2959 7874 19 0 24 213 4 1 190  
## feature2 feature3 feature4 feature5 feature6  
## 1039 1924 19347 24 347 24347  
## 1078 1924 19213 24 213 24213  
## 1344 1924 19347 24 347 24347  
## 1422 1924 19343 24 343 24343  
## 1982 8324 83171 24 171 24171  
## 2102 1924 19213 24 213 24213  
## 2140 1924 19213 24 213 24213  
## 2622 1924 19213 24 213 24213  
## 2875 1924 19213 24 213 24213  
## 2960 1924 19213 24 213 24213

# Feature5 - Here we start looking for interesting relationships in feature 5  
t5 = table(test.pred.one$feature5)  
t5 = t5[order(t5, decreasing = T)]  
  
## from the above the table we see that device = 0 and channel = 347   
## has a lot of ones  
test.347 = subset(test.pred.one, feature5 == '347')  
test.347[1:10,]

## click\_id ip app device os channel click\_time pred feature1  
## 424 423 4052 19 0 50 347 4 1 190  
## 558 557 46114 19 0 38 347 4 1 190  
## 1039 1038 72695 19 0 24 347 4 1 190  
## 1344 1342 48282 19 0 24 347 4 1 190  
## 3336 3335 87604 19 0 0 347 4 1 190  
## 3791 3790 5147 19 0 29 347 4 1 190  
## 4188 4187 11290 19 0 24 347 4 1 190  
## 5959 5959 100959 19 0 21 347 4 1 190  
## 6370 6369 46970 19 0 0 347 4 1 190  
## 6818 6818 59472 19 0 24 347 4 1 190  
## feature2 feature3 feature4 feature5 feature6  
## 424 1950 19347 50 347 50347  
## 558 1938 19347 38 347 38347  
## 1039 1924 19347 24 347 24347  
## 1344 1924 19347 24 347 24347  
## 3336 190 19347 0 347 347  
## 3791 1929 19347 29 347 29347  
## 4188 1924 19347 24 347 24347  
## 5959 1921 19347 21 347 21347  
## 6370 190 19347 0 347 347  
## 6818 1924 19347 24 347 24347

# Feature6 - Here we start looking for interesting relationships in feature 6  
t6 = table(test.pred.one$feature6)  
t6 = t6[order(t6, decreasing = T)]  
  
## from the above the table we see that os = 33 and channel = 347   
## has a lot of ones  
test.33347 = subset(test.pred.one, feature5 == '33347')  
test.33347[1:10,]

## click\_id ip app device os channel click\_time pred feature1 feature2  
## NA NA NA NA NA NA NA NA NA NA NA  
## NA.1 NA NA NA NA NA NA NA NA NA NA  
## NA.2 NA NA NA NA NA NA NA NA NA NA  
## NA.3 NA NA NA NA NA NA NA NA NA NA  
## NA.4 NA NA NA NA NA NA NA NA NA NA  
## NA.5 NA NA NA NA NA NA NA NA NA NA  
## NA.6 NA NA NA NA NA NA NA NA NA NA  
## NA.7 NA NA NA NA NA NA NA NA NA NA  
## NA.8 NA NA NA NA NA NA NA NA NA NA  
## NA.9 NA NA NA NA NA NA NA NA NA NA  
## feature3 feature4 feature5 feature6  
## NA NA NA NA NA  
## NA.1 NA NA NA NA  
## NA.2 NA NA NA NA  
## NA.3 NA NA NA NA  
## NA.4 NA NA NA NA  
## NA.5 NA NA NA NA  
## NA.6 NA NA NA NA  
## NA.7 NA NA NA NA  
## NA.8 NA NA NA NA  
## NA.9 NA NA NA NA

## Here we interect the data sets  
interesting0 = subset(test.pred.one, app == '19' & device == '0' & (os == '24' | os == '33') & channel == '347')  
  
interesting1 = subset(test.pred.one, (app == '19' | app == '29') & (device == '0' | device == '1') & (os == '0' | os == '24') & (channel == '213' | channel == '347'))

In the above results, we only consider the two labels with the highest frequency for each of the features. The interesting results that observed. There are few app, device, operating system and channel they have highest number of clicks compare to others. We observed with feature1 - the app 29 and device 1 has the higest frequecy of clicks feature2 - the app 19 and os 24 has the higest frequecy of clicks feature3 - the app 19 and channel 347 has the higest frequecy of clicks feature4 - the device 0 and os 24 has the higest frequecy of clicks  
feature5 - the device 0 and channel 347 has the higest frequecy of clicks  
feature6 - the os 33 and os 347 has the higest frequecy of clicks

## Future Study

Note that the results of this analysis is based on analyzing the first 20 millon observations of the training data set. A follow-up would be to consider all the observations in the training data set. The main reason why we could not include all the observations in the training data set was due to computing limitations.