Group18-HW5

Group 18 – Lince Rumainum & Nilam Reshim

October 8, 2019

Problem 1

Problem 1-(a)

1) First, we observed the missing percentage of each data field of the given data set. It shown below that there are a lot of missingness on some of the data fields and from the description of the data fields that were given, doing imputation or try to transform them into a numeric or factor attribute, can be tricky and might ended up hurting the linear model prediction instead of improving it. Which in that case, it is better to exclude those attributes and not consider them to be one of the parameters in our linear model. Shown below how much is missing for each of the data fields in the training data set:

```
##                      sessionId                         custId
##                   0.000000e+00                   0.000000e+00
##                           date                channelGrouping
##                   0.000000e+00                   0.000000e+00
##                 visitStartTime                    visitNumber
##                   0.000000e+00                   0.000000e+00
##             timeSinceLastVisit                        browser
##                   0.000000e+00                   1.427124e-05
##                operatingSystem                       isMobile
##                   4.381270e-03                   0.000000e+00
##                 deviceCategory                      continent
##                   0.000000e+00                   1.213055e-03
##                   subContinent                        country
##                   1.213055e-03                   1.213055e-03
##                         region                          metro
##                   5.492286e-01                   7.019024e-01
##                           city                  networkDomain
##                   5.569779e-01                   4.773444e-01
##                 topLevelDomain                       campaign
##                   4.773444e-01                   9.605971e-01
##                         source                         medium
##                   2.854248e-05                   1.687859e-01
##                        keyword                   isTrueDirect
##                   9.620528e-01                   0.000000e+00
##                   referralPath                      adContent
##                   6.145481e-01                   9.879979e-01
##          adwordsClickInfo.page          adwordsClickInfo.slot
##                   9.741548e-01                   9.741548e-01
##         adwordsClickInfo.gclId adwordsClickInfo.adNetworkType
##                   9.739407e-01                   9.741548e-01
##     adwordsClickInfo.isVideoAd                      pageviews
##                   9.741548e-01                   1.141699e-04
##                        bounces                      newVisits
##                   5.811106e-01                   3.417106e-01
```

2) From there, we then shrink our model based on what we know from the data fields. We chose attributes that can be categorized and create a numerical factor

representation for them, create a feature regarding ads, and the three numeric data fields, which are *newVisits*, *timeSinceLastVisit*, and *pageviews*. We are not considering date at the moment because we would like to approach the problem with all the attributes and we think that adding date might cause the linear model to overfit. Therefore, we analyze how the chosen attributes behavior against revenue by using Principal Component Analysis (PCA). To be able to do so, there are several new categorized attributes were created as IDs. Those new features are *channelID, deviceID, sourceID, mediumID, countryID* (refer to part (b) for details). We also add a column that indicates if ads were part of it and impute the NA values for bounces and newVisits variables to zeroes since those indicates that it did not bounce and it was not their first visit. From those, we evaluate the missingness of the data again below:

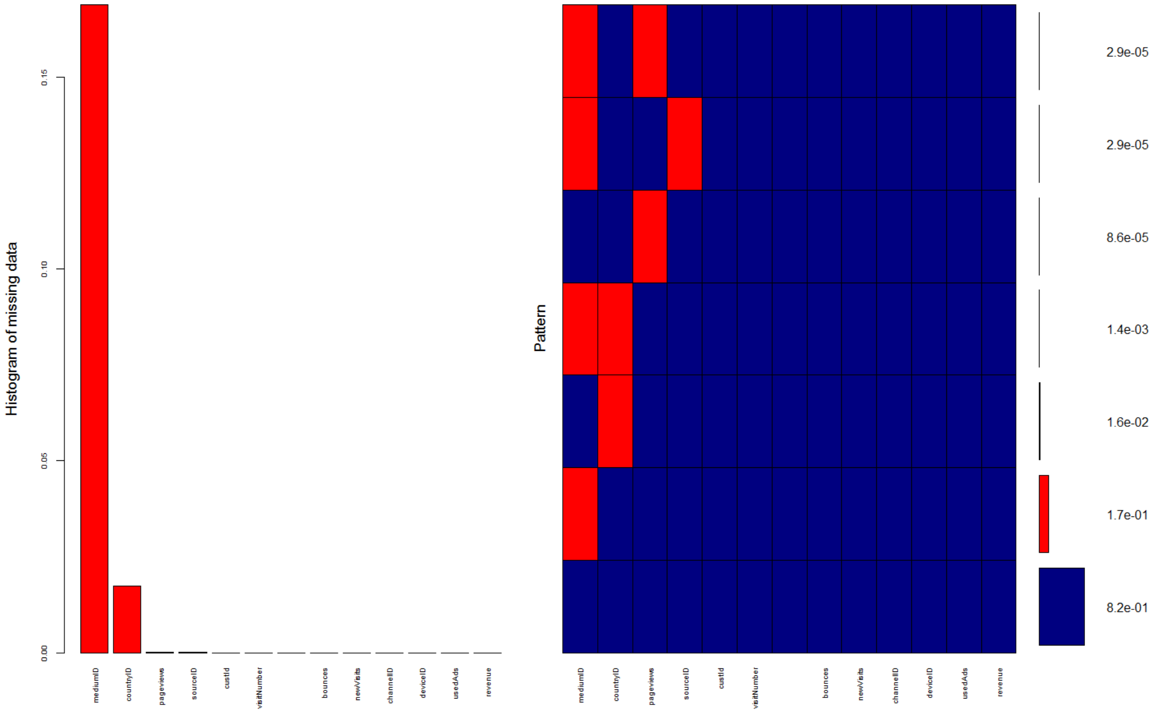
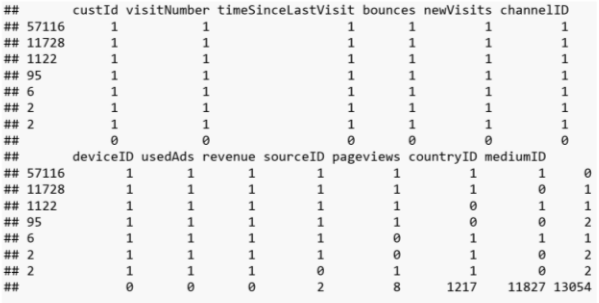


***Figure 1** Missing Data Visualization using aggr Function from VIM package*



```
##       custId visitNumber timeSinceLastVisit bounces newVisits channelID
## 57116      1           1                  1       1         1         1
## 11728      1           1                  1       1         1         1
## 1122       1           1                  1       1         1         1
## 95         1           1                  1       1         1         1
## 6          1           1                  1       1         1         1
## 2          1           1                  1       1         1         1
## 2          1           1                  1       1         1         1
##            0           0                  0       0         0         0
##       deviceID usedAds revenue sourceID pageviews countryID mediumID
## 57116        1       1       1        1         1         1        1     0
## 11728        1       1       1        1         1         1        0     1
## 1122         1       1       1        1         1         0        1     1
## 95           1       1       1        1         1         0        0     2
## 6            1       1       1        1         0         1        1     1
## 2            1       1       1        1         0         1        0     2
## 2            1       1       1        0         1         1        0     2
##              0       0       0        2         8      1217    11827 13054
```

***Figure 2** Missing Data using md.pattern from mice Package*

Figure 2 shows that there are 57,116 complete cases, 11,728 missing data in *mediumID*, 1,122 in *countryID*, 95 in combinations of *countryID* and *mediumID*, 6 in pageviews, 2 in combinations of *pageviews* and *mediumID*, and 2 in combinations of *sourceID* and *mediumID*; with the highest portion of missing data are in *mediumID* and then *countryID*, *pageviews*, and *sourceID* follows respectively (see **Figure 1**).

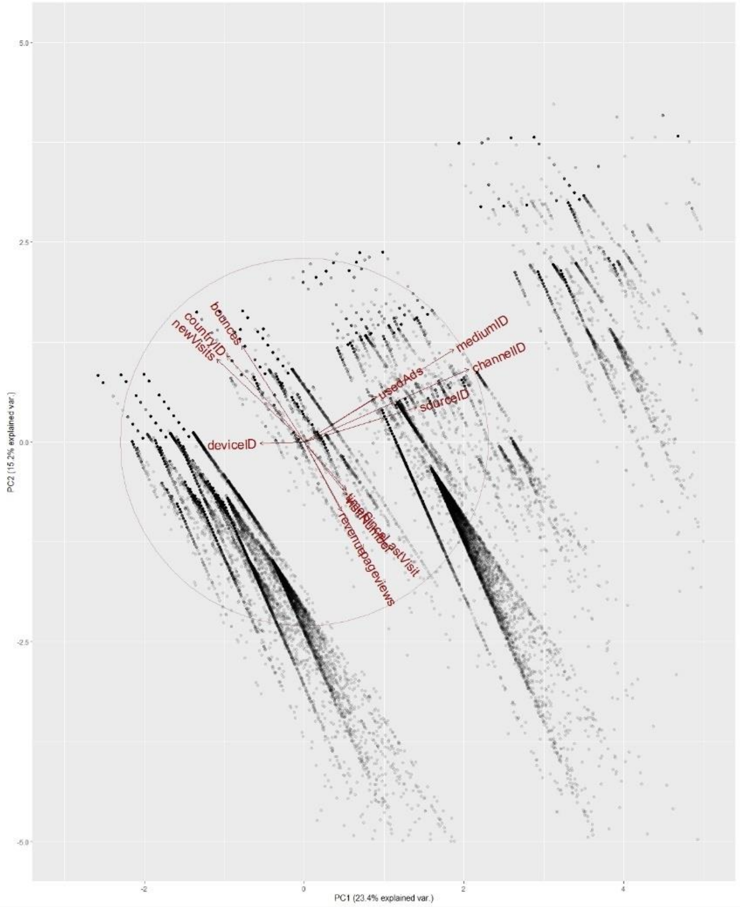


Figure 3 *Biplot of Principle Component 1 and Principle Component 2 of Training Data*

From **Figure 3**, it clearly shows that the revenue are highly correlated with *pageviews*, *visitNumber*, and *timeSinceLastVisit* while the opposite of them are *countryID*, *bounces*, and *newVisits*.

3) Since we are looking at the data at the customer level for our linear model, we went ahead and aggregate those attributes and group them by customer ID. The sum of the revenue of each customer were calculated, for *newVisits*, *timeSinceLastVisit*, and *pageviews*, the mean was calculated and for the rest of the attributes, it is each of their mode. We taken the sum of the revenue because of the fact that it will be considered as the parameter of the linear model. The mean was calculated for the numeric attributes assuming that will be the average visit, time, and views of that customer. Also, the mode were taken for the rest of features because we assume that it is their most likely behavior on that particular attribute. From there we can analysis more about each customer behavior. Below is the R code of it:

```
# group all data by customerID
# taken the mean of the first three numeric columns
# taken the mode of the logical and factor columns
dfTr_custId <- dfTrainNum %>%
  group_by(custId) %>%
  summarize(visitNumber = mean(visitNumber),
            timeSinceLastVisit = mean(timeSinceLastVisit),
            pageviews = mean(pageviews),
            bounces = getmode(bounces),
            newVisits = getmode(newVisits),
            channelID = getmode(channelID),
            deviceID = getmode(deviceID),
            sourceID = getmode(sourceID),
            mediumID = getmode(mediumID),
            countryID = getmode(countryID),
            usedAds = getmode(usedAds),
            revenue = sum(revenue))  %>%
    arrange(custId)
```

4) From the analysis that were done previously, it clears that a customer’s visit number, the time since their last visit, and their page views, should effect more on the revenue than any other *variables*. Therefore, *we plot those along with the other features to see if the data is linearly distributed and if any category generates more revenues than the others*. We also considering the feature *usedAds* in our plots to see if ads does help with generating more revenue and in what case(s) it is used the most. We also want to see if the data are highly skewed in any of the variables and the possible outliers that can be excluded or treated as a special case. The result from the plot shows outliers and skewness on **Figure 1**, **Figure 2**, and **Figure 3** while **Figure 17** and **Figure 18** shows that Ads were only efficient on paid search channel and cost-per-click paid search medium, respectively. (Refer to **Appendix A** for all the plots for this *revenue* against all the other chosen variables)

5) Since it shown in **Appendix A** that the top three correlated attributes to *revenue* are highly skewed, transformation of the data set is necessary. The approach we take to overcome the skewness is to take the log of those data so it will create a better distribution and later, better prediction parameters for the linear model and imputation. The plot of *revenue* against the updated parameters are shown below:

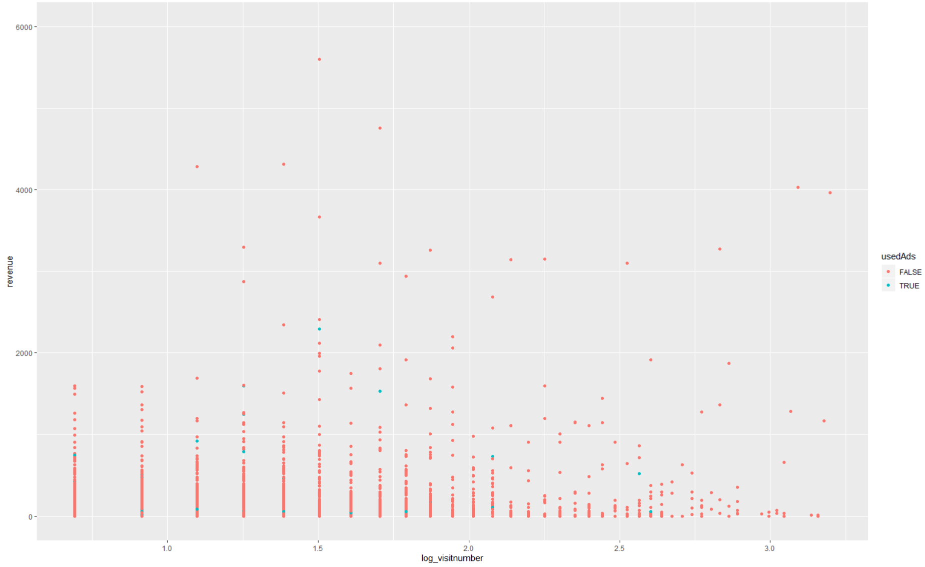


Figure 4 *Revenue vs Log of Visit Number*

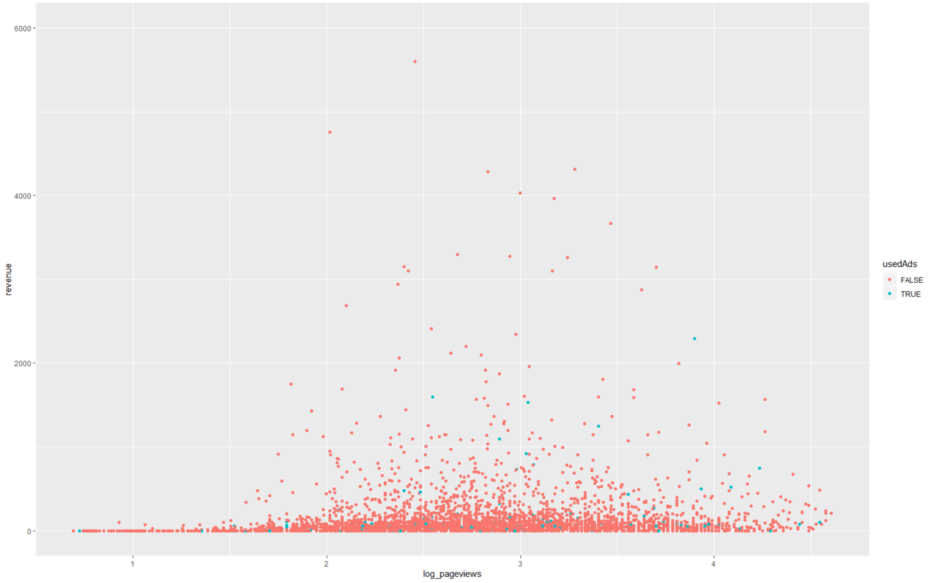


Figure 5 *Revenue vs Log of Page Views*

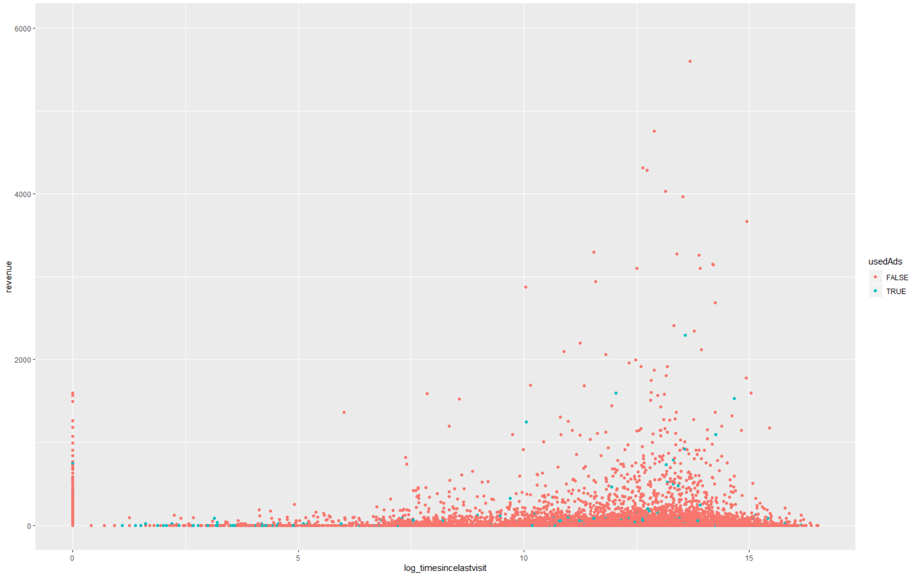


Figure 6 *Revenue vs Log of Time Since Last Visit*

Problem 1-(b)

To prepare the data for modeling, there a few that we talked about in part (a), such as, imputing the missing values for *bounces* and *newVisits* attributes and replacing them with zeroes, aggregating the data set into customer based level, and creating categorical columns into factors. In here we are going into details about how those factors were created. By using the fct_lump function from forcats package, we are able to create factors for channel, device, source, medium, and country column, where 1 is the most occurrences for that category and 2 the next most number of occurrences and so on.

- For channel, there are eight categories: 1-Organic Search, 2-Social, 3-Referral, 4-Direct, 5-Paid Search, 6-Affiliates, 7-Display, and 8-(Other).
- For device, there are three categories: 1-desktop, 2-mobile, and 3-tablet.
- For source, there are ten categories: 1-“google”, 2-“youtube.com”, 3-“(direct), 4-mall.googleplex.com, 5-analytics.google.com, 6-Partners, 7-dfa, 8-google.com, 9-sites.google.com, 10-other.

- For medium there five categories: 1-organic, 2-referral, 3-cpc, 4-affiliate, 5-cpm.
- For country there ten categories: 1-United States, 2-India, 3-United Kingdom, 4-Canada, 5-Vietnam, 6-Thailand, 7-Turkey, 8-Germany, 9-Brazil, 10-Other.

From part(a), **Appendix A** shows that there are some outliers in *pageviews, visitNumber, and timeSinceLastVisit*. To deal with that case, we reduce the *visitNumber* to less than twenty five and the *pageviews* to less than one hundred. Since there is only one revenue above 6,000 in the training data, that outliers also was excluded. By excluding those data we still have at least 99% of training data so we decided to take that approach. Before going into the linear model, the remaining NA values are being imputed by using the MICE package in R, with the norm.predict method, which is the linear regression, predicted values method. The method works well with only creating ten multiple imputations and five iterations. As shown in **Figure 7**, two iterations would already converging the mean and standard deviation values for the variable’s (*sourceID, mediumID,* and *countryID*) missing values.

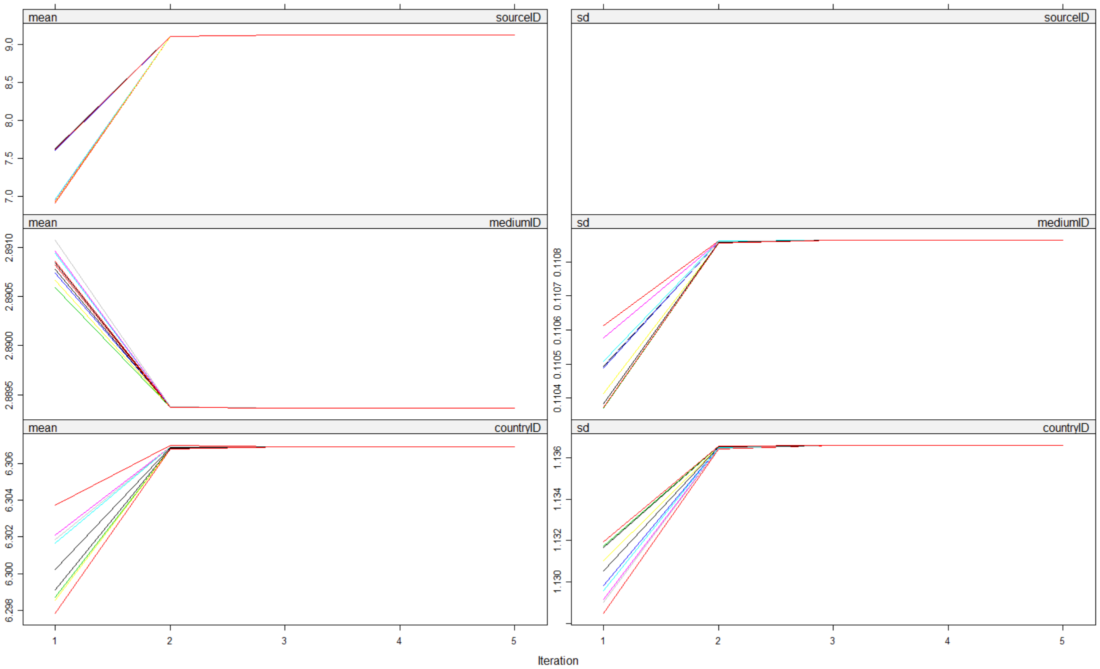


Figure 7 *Shows the Mean and Variance Behavior for Each Iteration with Linear Predicted Values for Each Missing Attribute*

Problem 1-(c)

Now, after conducting exploratory data analysis and preparing the data for modeling, we create our linear model based on those analysis. The linear model that give us the best outcome is below:

log(revenue + 1) ~ log_visitnumber + log_timesincelastvisit + log_pageviews + channelID + deviceID + sourceID + mediumID + countryID + bounces + newVisits

That linear model above is used on the imputed training data set. **Figure 8** below show the behavior of the linear model. It shows the Residuals vs Fitted plot, where we want that to be as close to zero as possible, the Normal Q-Q plot, where we want the data to be as close to normal distribution as possible (indicated by a positive slope linear trend line), the Scale-Location plot, where we want to be more equally spread across the predictor, and the Residuals vs Leverage, where we want to see if there is still case with a high leverage. In our case the linear model could still be improved but it should still be able to create good predictions for the missing revenues on the test data set.

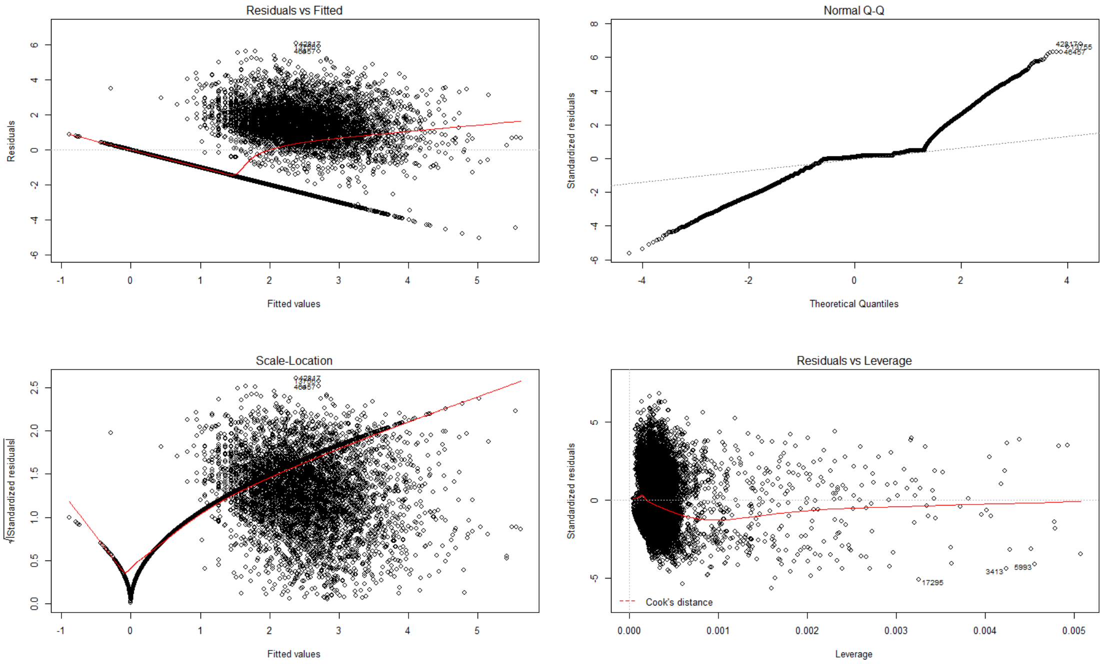


Figure 8 *Linear Model Behavior with the Imputed Training Data*

Here are the summary of the residuals, each variables' coefficient, and p-value:

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8473 -0.1886  0.0630  0.2616  6.1051
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -1.723639   0.051056 -33.760  < 2e-16 ***
## log_visitnumber         0.955833   0.034322  27.849  < 2e-16 ***
## log_timesincelastvisit  0.027904   0.001391  20.059  < 2e-16 ***
## log_pageviews           1.106675   0.007827 141.387  < 2e-16 ***
## channelID              -0.013105   0.011212  -1.169  0.24248
## deviceID               -0.165608   0.007850 -21.097  < 2e-16 ***
## sourceID                0.012050   0.002400   5.020 5.18e-07 ***
## mediumID                0.052530   0.017547   2.994  0.00276 **
## countryID              -0.018511   0.001112 -16.651  < 2e-16 ***
## bounces                 0.724086   0.011546  62.712  < 2e-16 ***
## newVisits              -0.283650   0.024862 -11.409  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8798 on 47185 degrees of freedom
## Multiple R-squared:  0.5426, Adjusted R-squared:  0.5425
## F-statistic:  5598 on 10 and 47185 DF,  p-value: < 2.2e-16
```

So, for our model, the RMSE value is 0.8798 and the R^2 value is 0.5426.

We do think our model can go through more tuning to reach better prediction for the test revenues although it is much better than the model we first made. We ran into several problems, one of them was the runtime on imputation process. We overcome it by testing the imputation for different method and see if the missing values data is converging or not. The last problem I ran to was more technical than it is modeling. As I gather everything into the Rmarkdown file, plyr and dplyr package was causing error in aggregating the data set, which took awhile to realize that it was the case, but came with an easy fix by detaching the ggbiplot library then plyr library after plotting PCA for everything to get to normal.

Appendix A

Problem 1-(a)

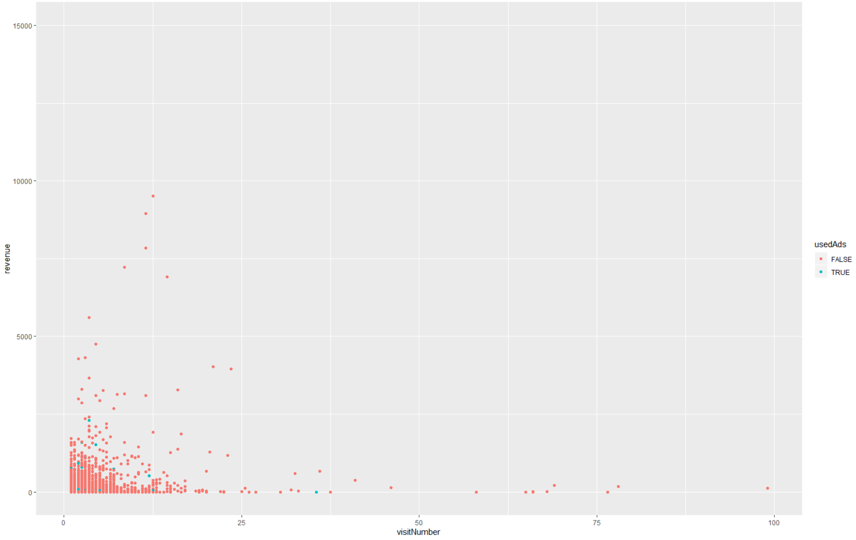


Figure 9 *Revenue Vs. Visit Number*

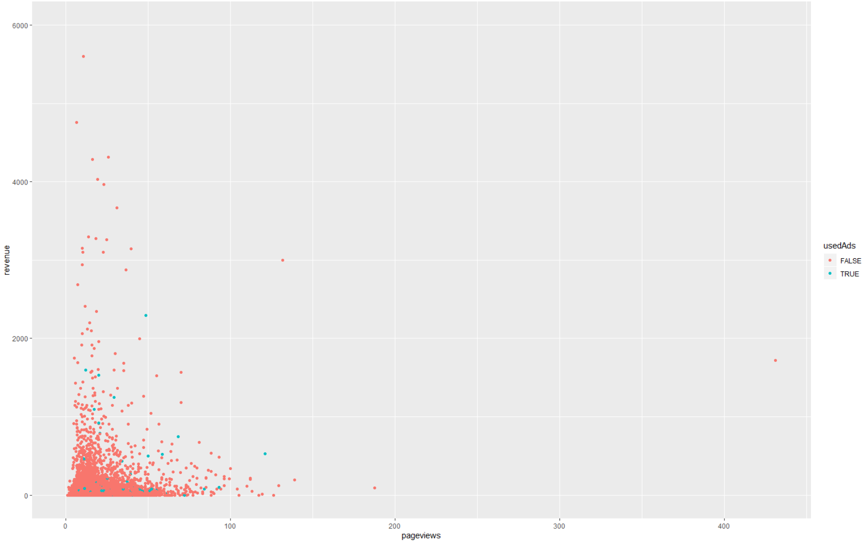


Figure 10 *Revenue Vs. Page Views*

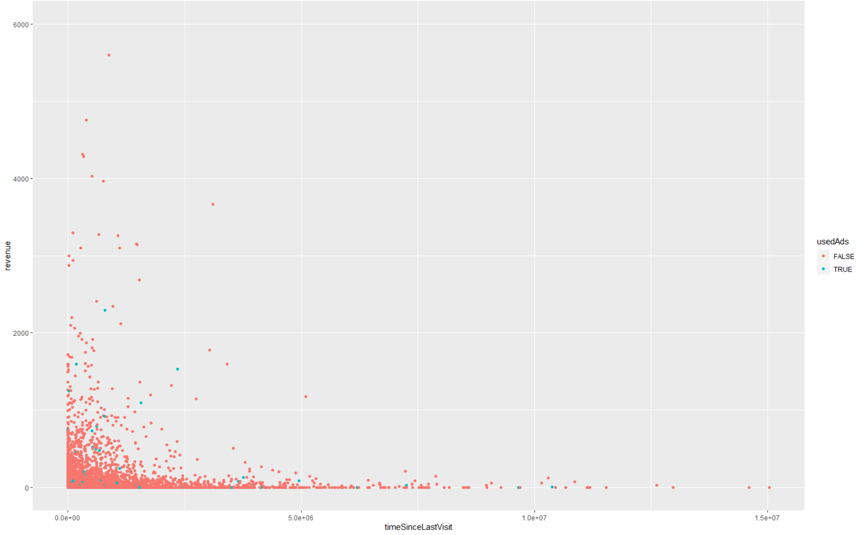


Figure 11 *Revenue Vs. Time Since Last Visit*

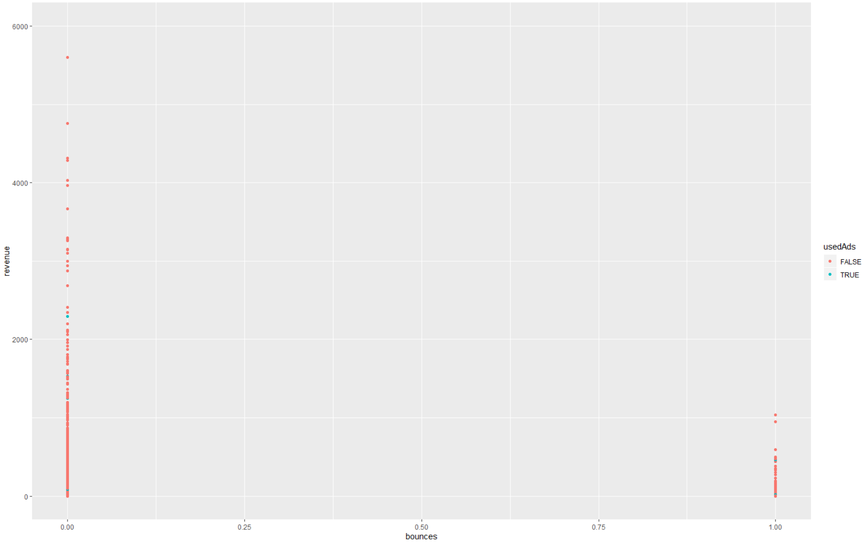


Figure 12 *Revenue Vs. Bounces*

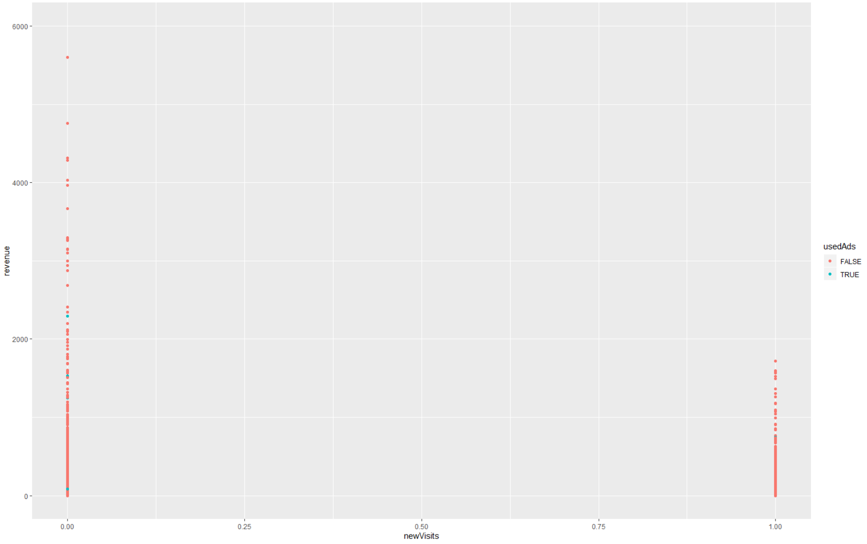


Figure 13 *Revenue Vs. New Visits*

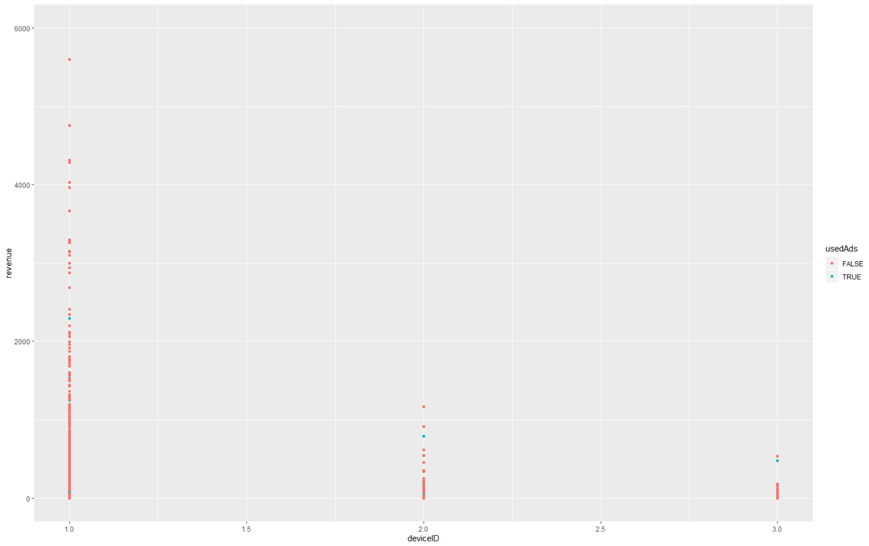


Figure 14 *Revenue Vs. Device ID*

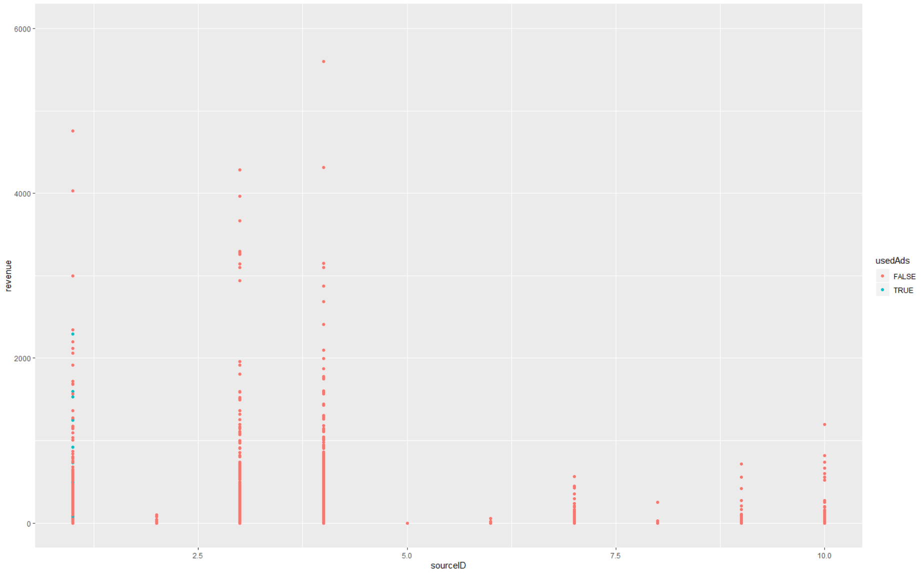


Figure 15 *Revenue Vs. Source ID*

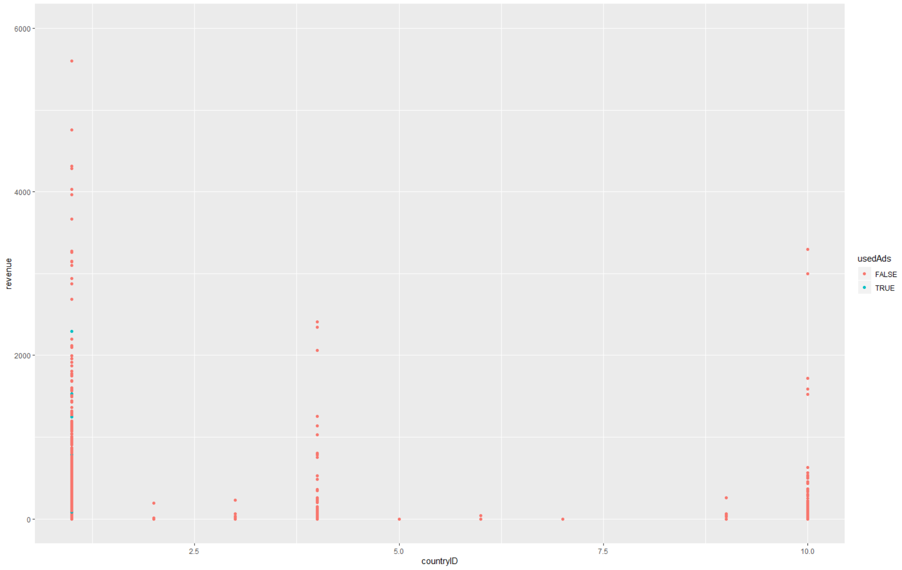


Figure 16 *Revenue Vs. Country ID*

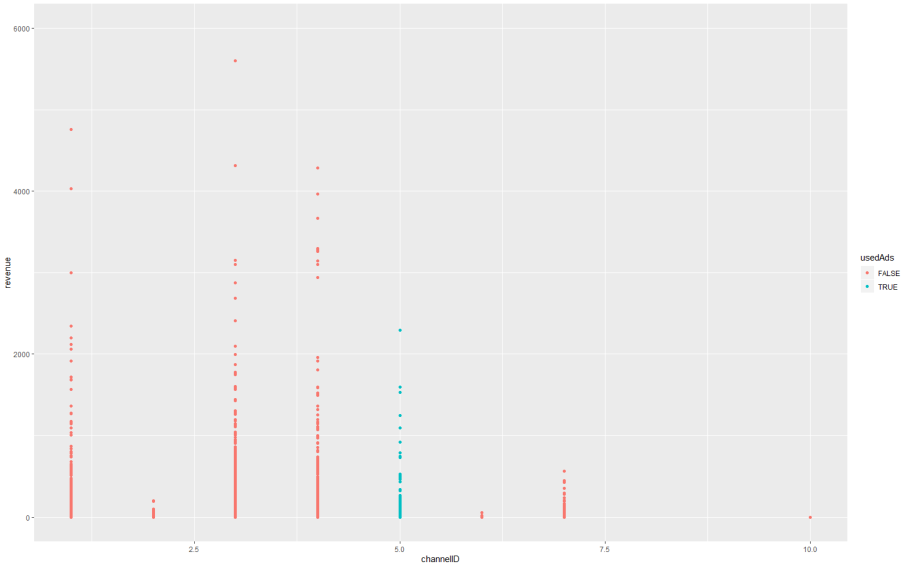


Figure 17 *Revenue Vs. Channel ID*

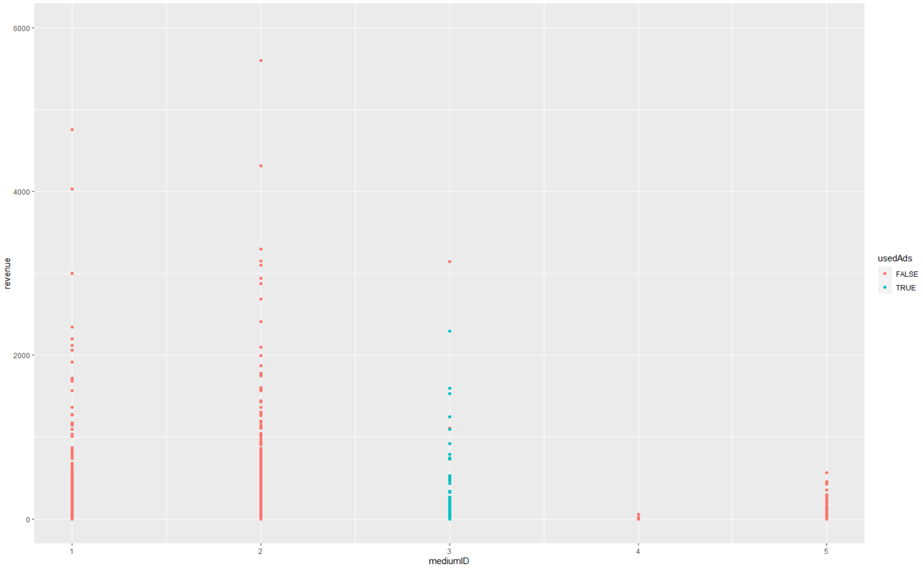


Figure 18 *Revenue Vs. Medium ID*