**E-Mail Marketing Campaign Analysis**

Made by: Ankit Sharma

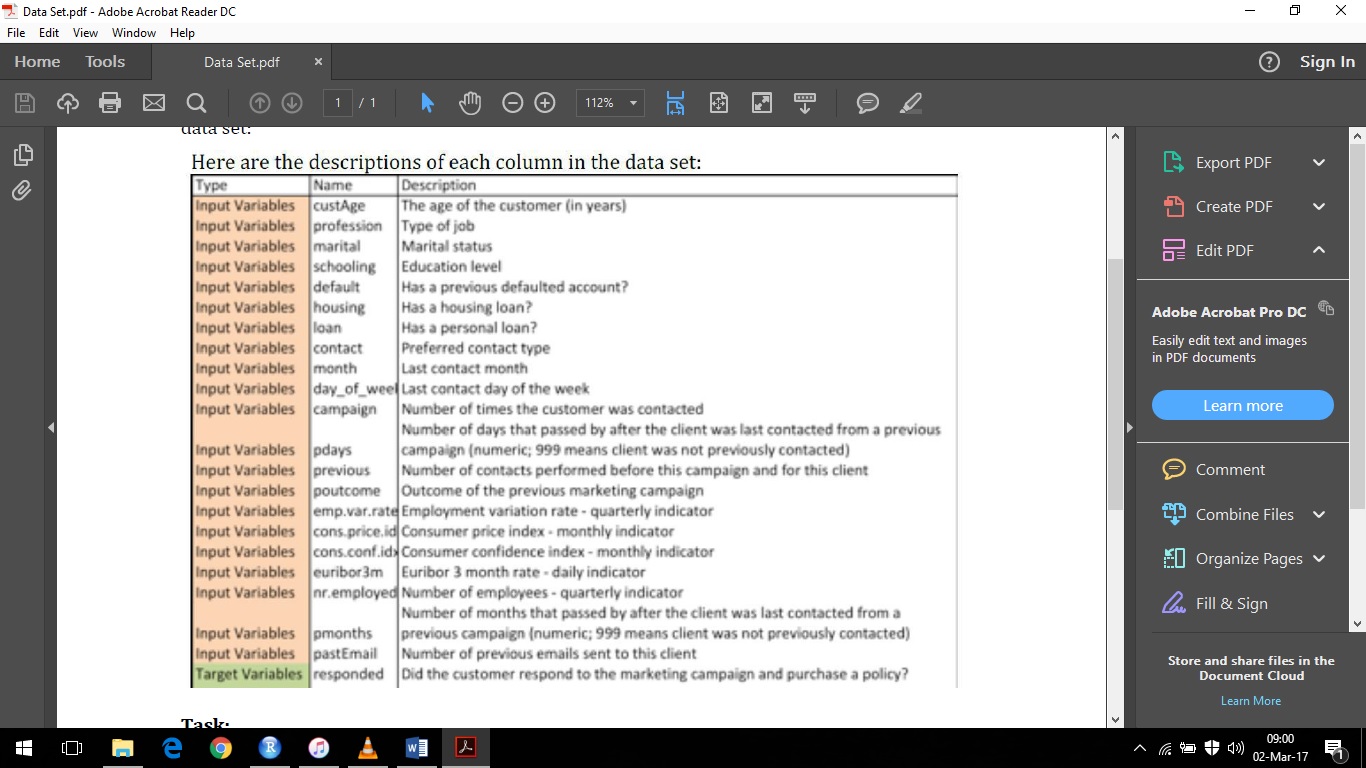
E-mail: [ankitactive@gmail.com](mailto:ankitactive@gmail.com)

**Problem**

Design a model that will be able to predict whether a customer will respond to the marketing campaign based on his/her information. In other words, predict the ‘responded’ target variable described above based on all the input variables provided.

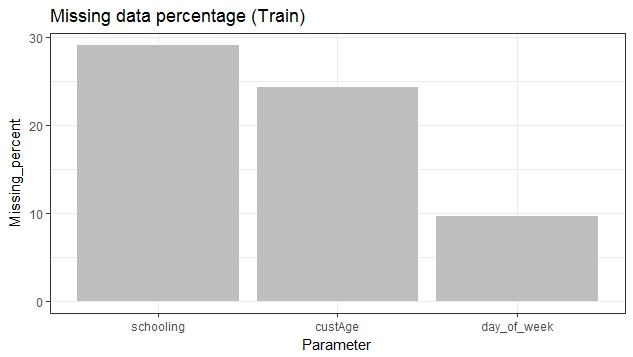
**About the dataset**

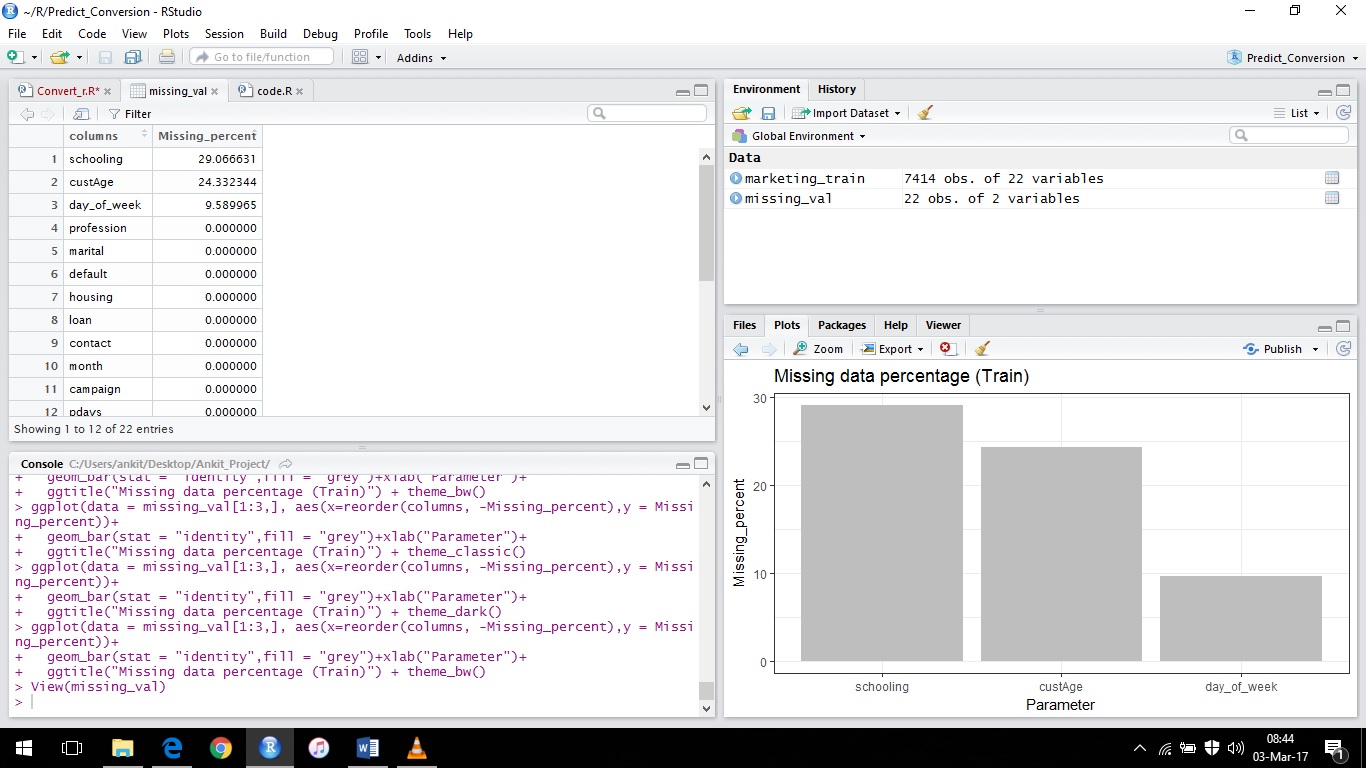
Company has been commissioned by other insurance company to develop a tool to optimize their marketing efforts. They have given us a data set as a result of an email marketing campaign. The data set includes customer information, described below, as a well as whether the customer responded to the marketing campaign or not. Here are the descriptions of each column in the data set:



**Missing Values Analysis**

I have calculated percentage and frequency of missing value by column and ordered in descending order in a new spreadsheet .Missing values treatment has been done using the KNN Imputation Algorithm. KNN imputation has been used and k=5 has been used as it is giving the nearest accuracy out of k ={1,2,3,4,5,6,7}





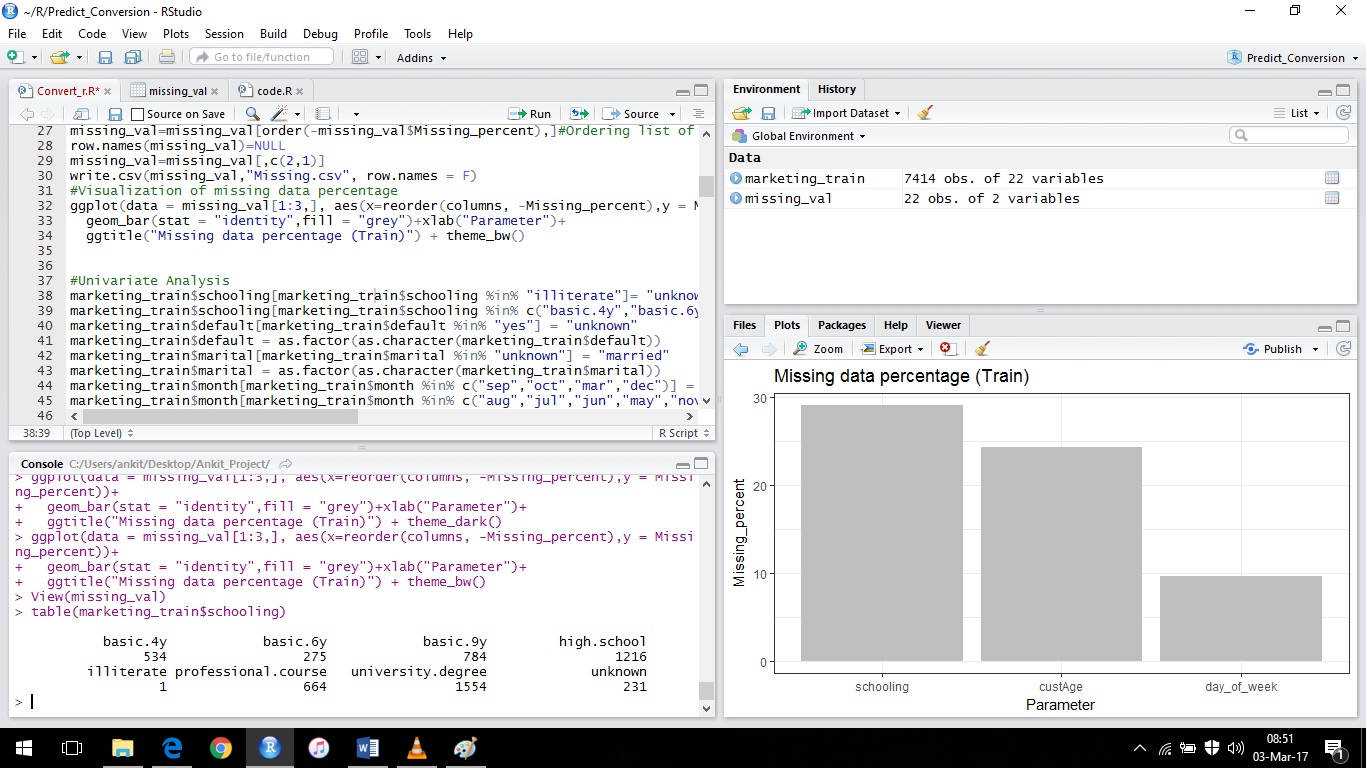
**Univariate Analysis and Merging**

We use str() function to understand data and from that after finding out about data types I decided to use table() function to understand each variable separately .

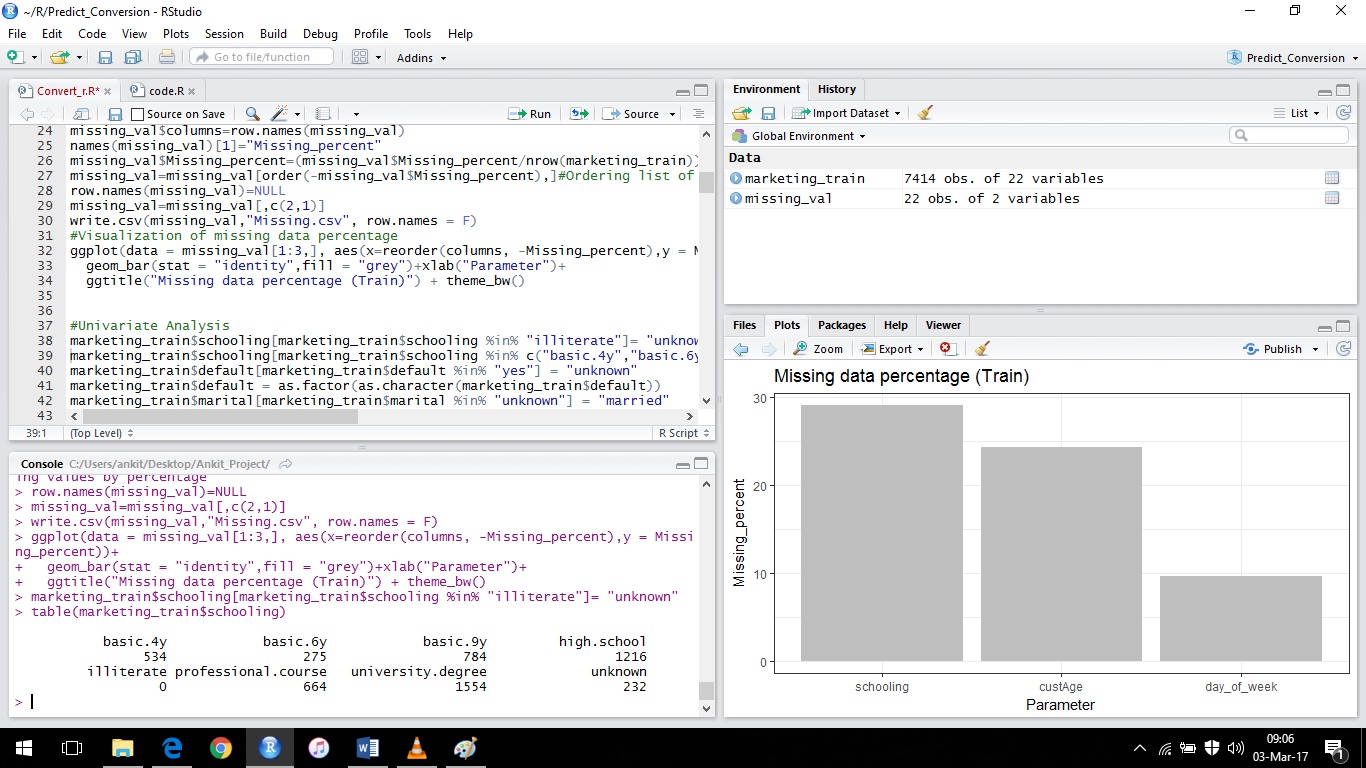
Example: Using table(marketing\_train$schooling) we are able to see that there are 231 unknown variable types and only 1 illiterate type under this variable.

The data is clearly not balanced because if unknown element in the variable.

So, “illiterate” has been merged with “unknown” to balance the data without making any heavy changes with the other elements of the variable, and avoiding another unnecessary level



After merging it looks like:



“Unknown” has that extra one variable from the “illiterate” level.

After this step we merge the schooling data “high.school” using domain understanding i.e., ‘basic.4y’, ‘basic.6y’, ‘basic.9y’, ‘professional.course’ .

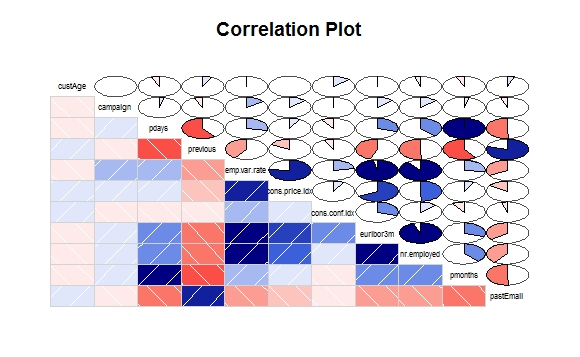
After this we’re left with only 3 main levels.

This procedure has been repeated on other variables too with similar understanding.

**Correlation Analysis**

There are both numerical and categorical variables in the dataset. So we plot the corrgram to find which all variables are highly correlated with each other.

For this procedure we first identify and then extract the numerical values from the data. Using this we plot a corrgram to estimate and find the highly correlated variables

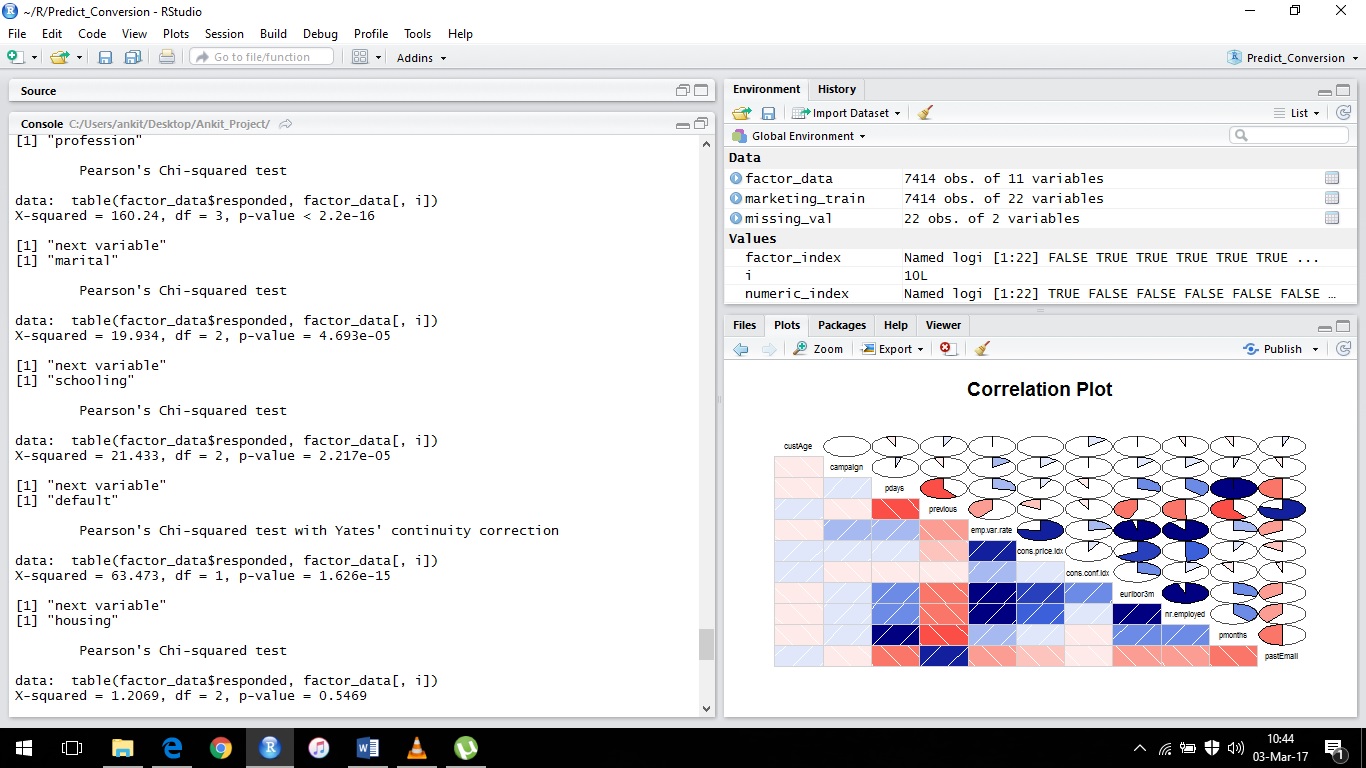


Null hypothesis: Independent and dependent are not related to each other

Alternate hypothesis: Independent and dependent are highly related.

Based on this we run the Chi-Square test.

We use the remaining categorical variable to run the Chi-Square test to extract the p values and use these hypothesis accordingly



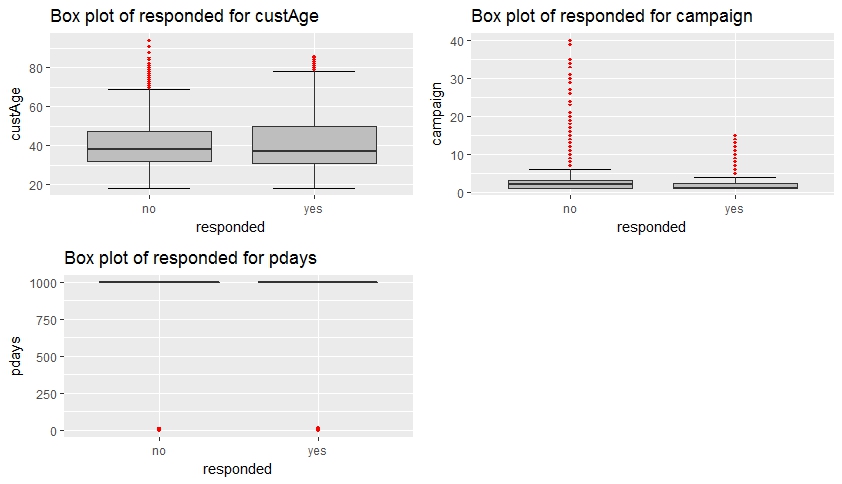
From this analysis we find and eliminate variables based on p values extracted. Here we're considering only values with p values <0.05 therefore rejecting variables with more than 0.05 thus accepting null hypothesis and rejecting the variable from the model

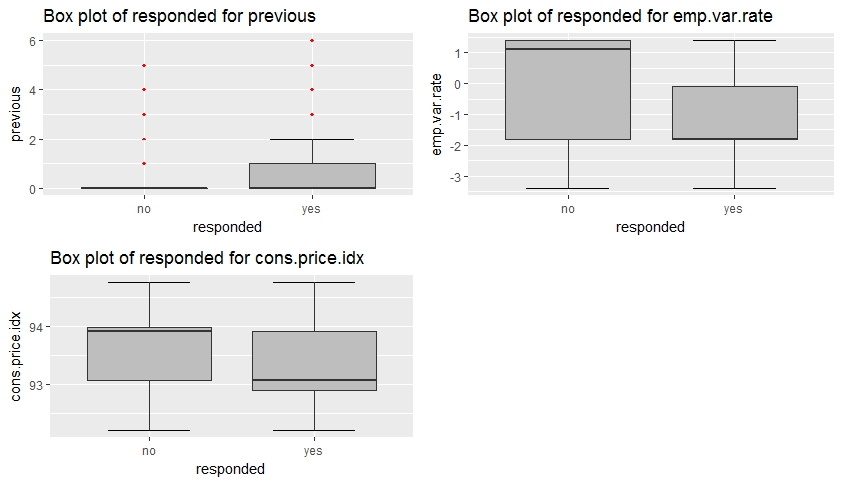
So we’re dropping the variables: pday, emp.var.rate day\_of\_week, loan, housing.

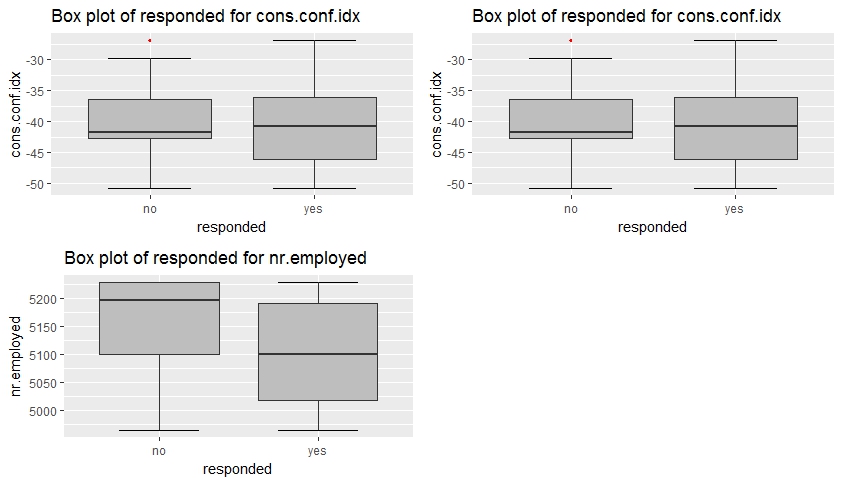
**Outlier Analysis**

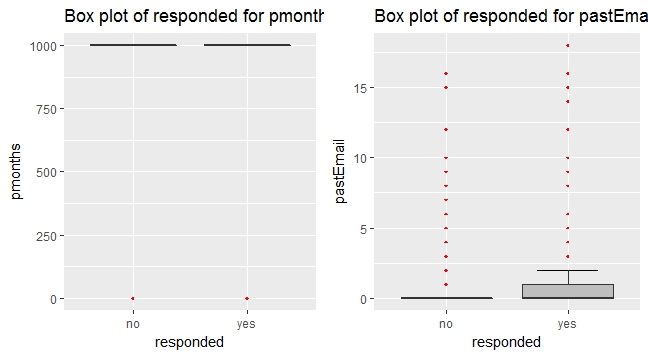
To remove outliers, boxplot method has been used by the method of assigning multiple boxplots to variables in a loop and plotting them together for better understanding of variables collectively.

But for this data outlier removal will leave it with approximately 3000 values out of approximately 7500 values. So outlier removal code has been written in comments only for larger datasets.



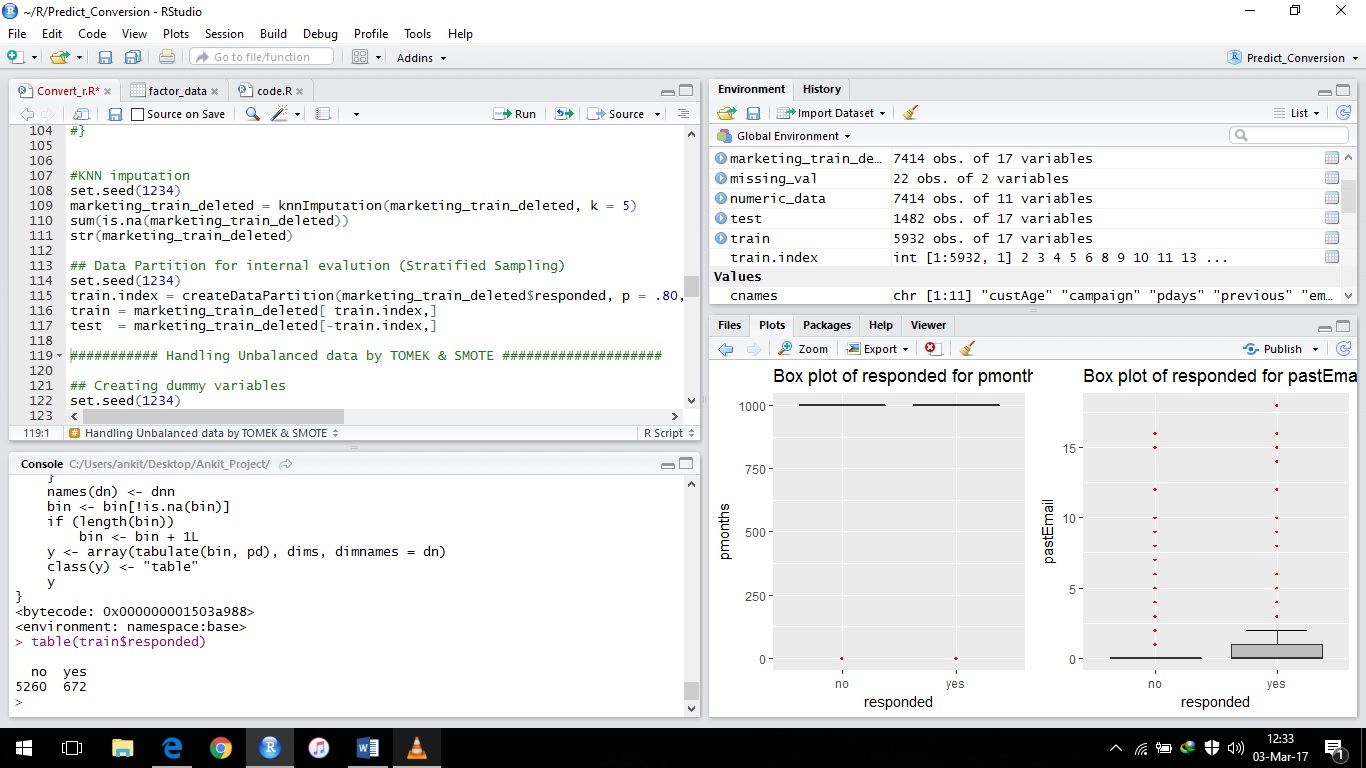






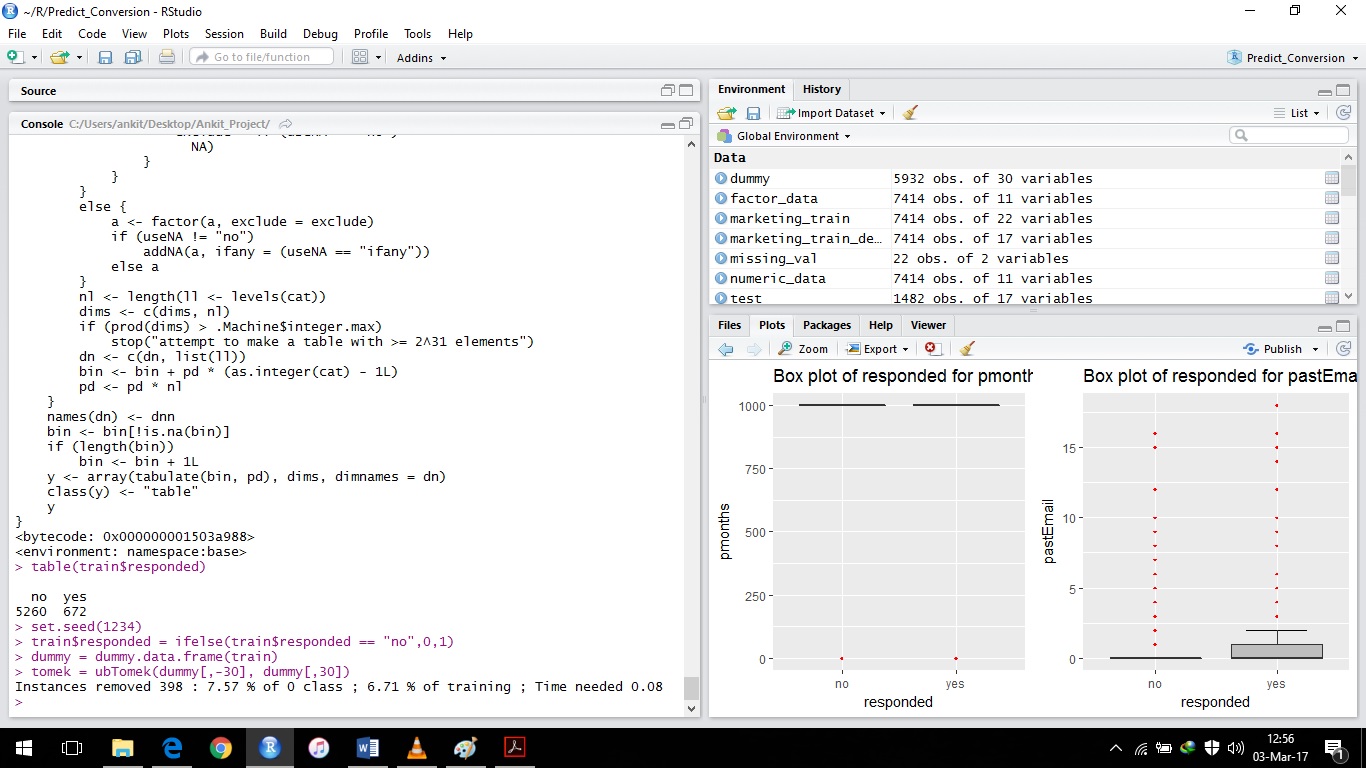
**Handling Unbalanced Data**

Analyzing the responded(target variable) we get unbalanced data.



So oversampling using SMOTE is used to handle imbalanced data.

And TOMEK is used to remove



Article read to guide towards SMOTE and TOMEK

“***Tomek Link (T-Link):***

*Let x be an instance of class A and y an instance of class B.*

*Let d(x, y) be the distance between x and y.*

*(x, y) is a T-Link, if for any instance z, d(x, y) < d(x, z) or d(x, y) < d( y, z)*

*If any two examples are T-Link then one of these examples is a noise or otherwise both examples are located on the boundary of the classes.*

*T-Link method can be used as a method of guided undersampling where the observations from the majority class are removed.*

*Several researches use T-link as a method of under sampling.*

***Synthetic Minority Oversampling Technique (SMOTE)***

*SMOTE is an advance method of over-sampling developed by Chawala. It aims to enrich the minority class boundaries by creating artificial examples in the minority class rather replicating the existing examples to avoid the problem of overfitting. The algorithm works as follows:*

*Let A be the minority class and let B be the majority class.*

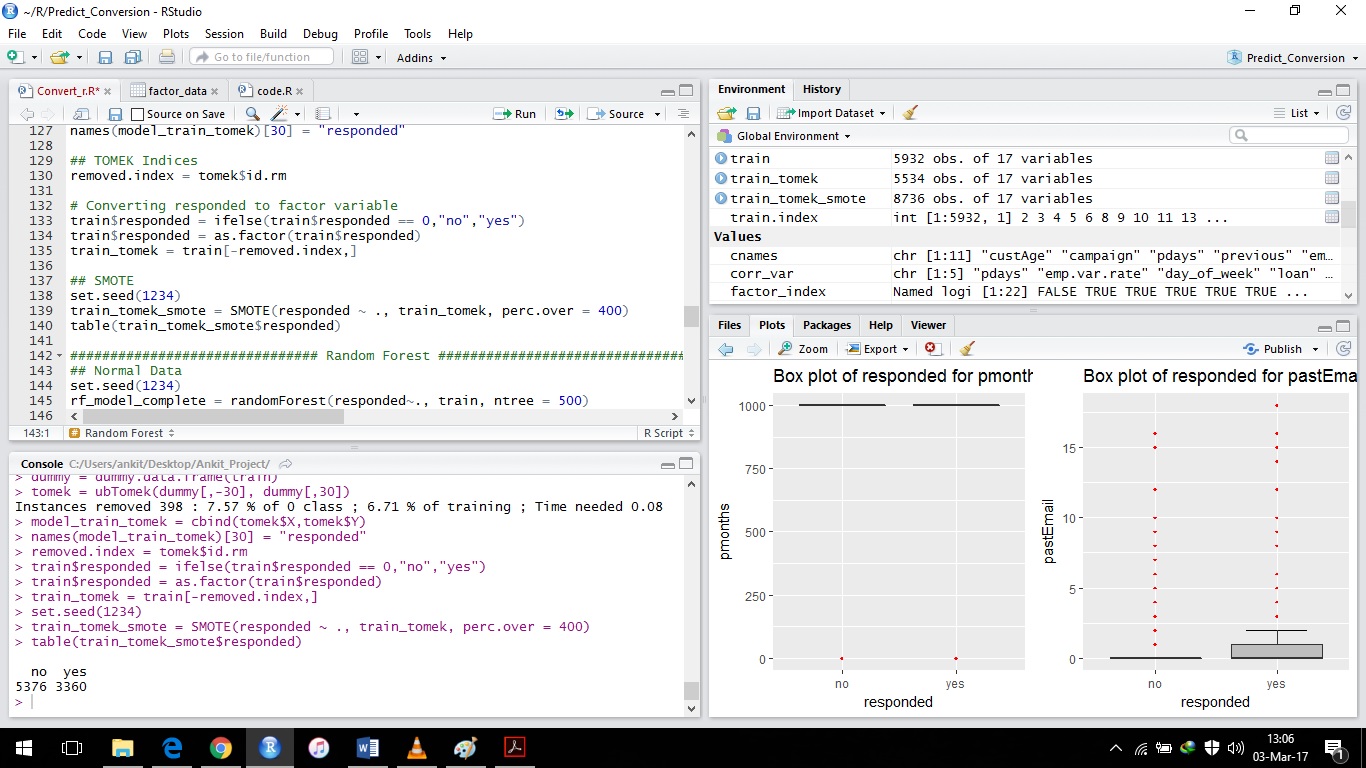
*Then, for each observation x belongs to class A, a k-nearest neighbors of “x” were identified,*

*A few neighbors are randomly selected (the number of neighbors depends on the rate of over-sampling),*

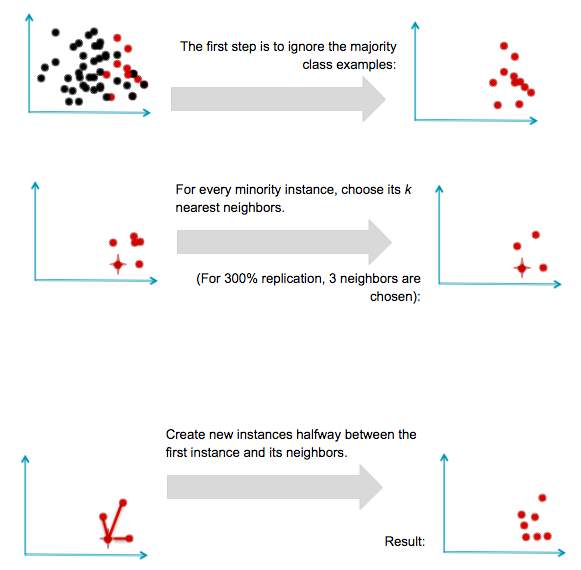
*Artificial observations are then generated and spread along the line joining the “x” to its nearest neighbors.*

*Several methods have been developed to improve the original SMOTE algorithm such as dealing with nominal features. New methods include SMOTE-NC (*[*Synthetic*](https://www.imedpub.com/search-results.php?page=4&&keyword=Synthetic) *Minority Over-sampling Technique Nominal Continuous) and SMOTE-N (Synthetic Minority Over-sampling Technique Nominal). These methods can be considered as a generalization of the original SMOTE algorithm to handle data sets with mixed Features (continuous and nominal).*

*Several works have been done in this field. Estabrooks et al. proposed a multiple re-sampling method that selects the most appropriate re-sampling rate. Jo et al. introduced a cluster-based over-sampling method which considers the between-class imbalance and within-class imbalance simultaneously. Guo et al focused on the hard examples of the majority and minority classes using the boosting algorithm, and then generated new synthetic examples from hard examples and add them to the data sets. Han et al. presented two new minority over-sampling methods based on SMOTE method, borderline-SMOTE1 and borderline-SMOTE2, where only the minority examples near the borderline are oversampled. These approaches achieve better* [*sensitivity*](https://www.imedpub.com/search-results.php?page=2&&keyword=sensitivity) *rate and F-value as compared to SMOTE and random over-sampling methods.”*

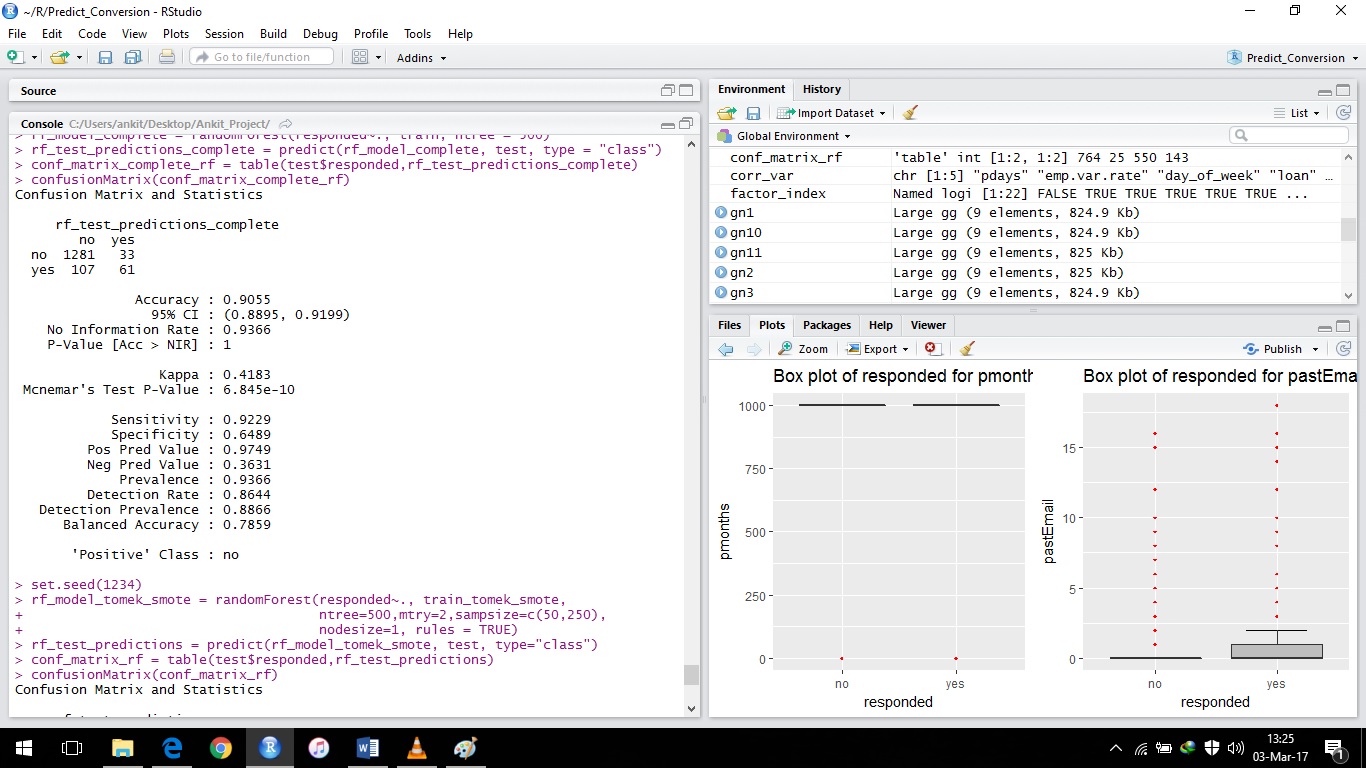
After implementing SMOTE and TOMEK, the variable/data looks like

This is how it works:



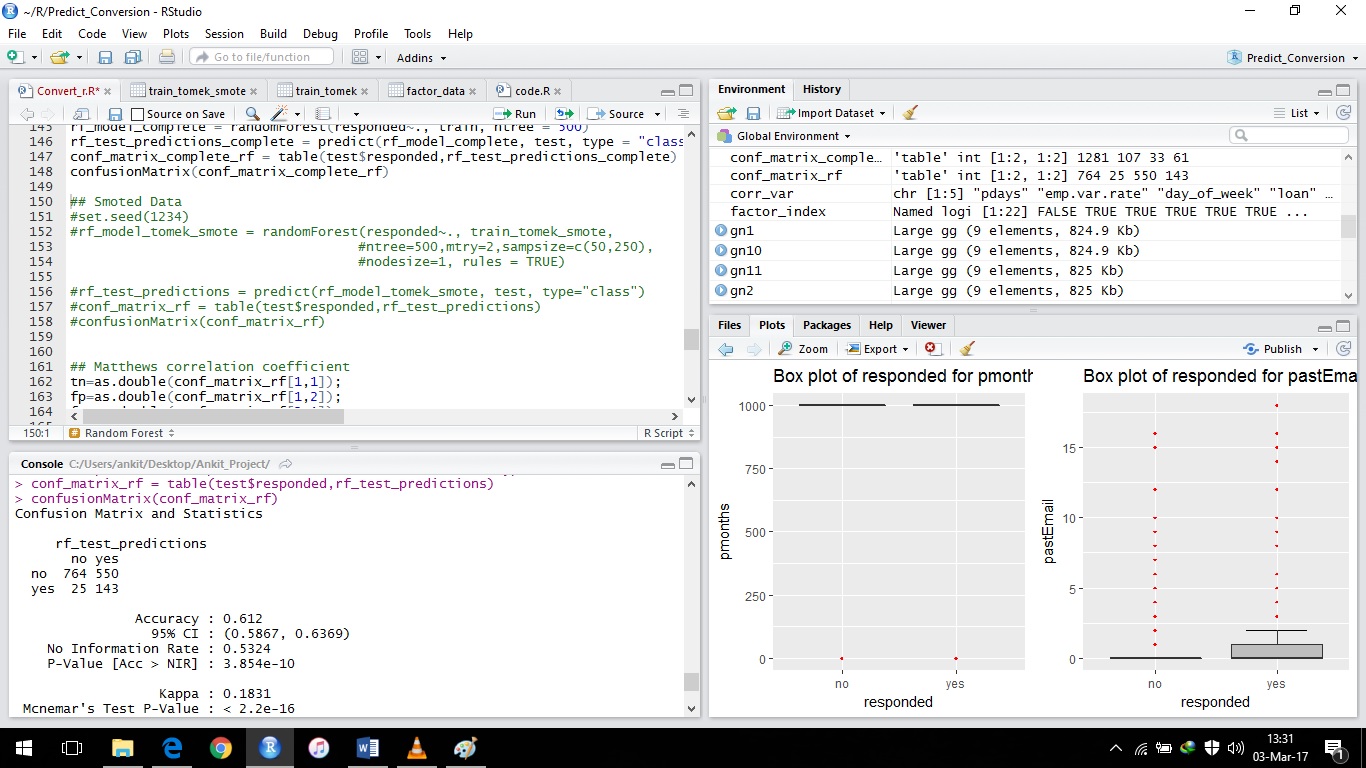
For actual and nearest instance it creates an observation in between by approximation therefore creating a new instance in between the first and nearest instance. So data wont vary much it’ll be in range. These are dummy variables created to balance data . It increased the minority observations(no) 400% times.

From 672 to 3360. Majority observations has also changed



The concern here is to reduce false negative rate and increase the true positive rate

Below matrix is from smoted data which shows both conditions have been met somewhat.



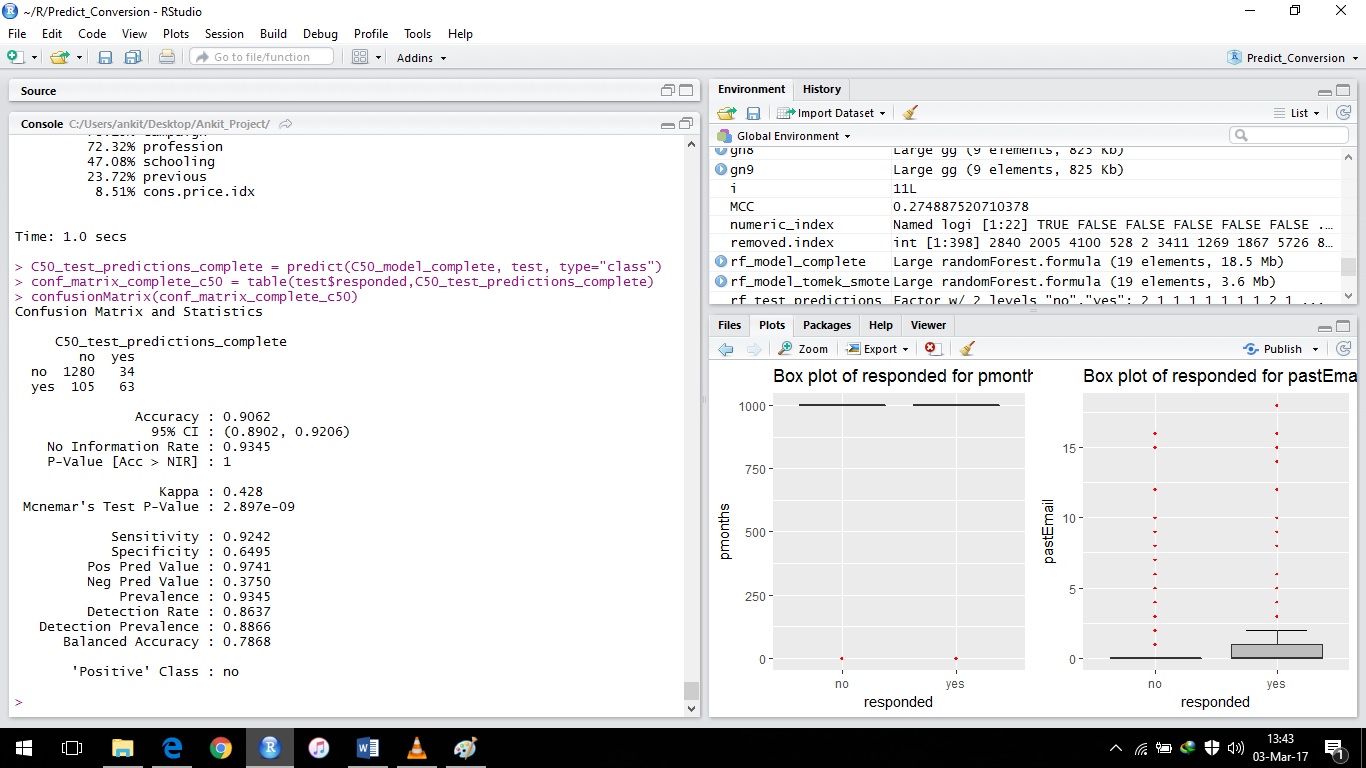
MCC(Matthews Correlation Coefficient) is used to trade between accuracy and recall. Ranges from -1 to +1

If it is

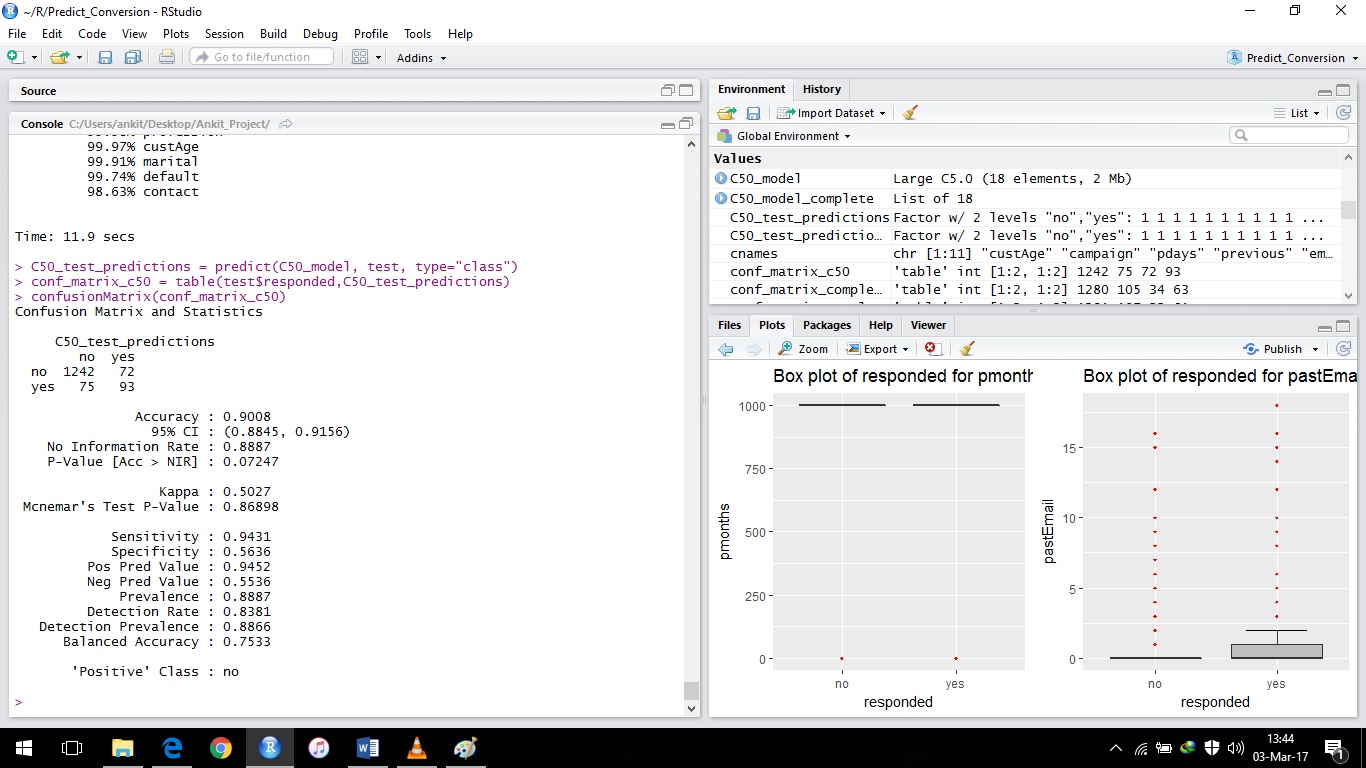
|  |  |
| --- | --- |
| MCC | Meaning |
| -1 | Not a better tradeoff between recall and accuracy to extract best model out of it |
| 0 | Not better than any random estimation/prediction |
| +1 | Better tradeoff between recall and accuracy |

In random Forest MCC= 0.278

Then using C5.0



Using smoted data



Increased true positive and decreased false positive as well

Now MCC= 0.5027 which is closing to +1 thus its better