BUSINESS ANALYTICS & DATA SCIENCE ASSIGNMENT

Report of Group 36

Abstract

Predictive Analytics have become a crucial aspect for many e-commerce companies. The main objective of this assignment is to apply and compare different machine learning techniques to build an efficient and accurate predictive model for the analysis and identification of customers who are expected to purchase gain in the next 90 days. The main performance goal of the predictive model is to maximize profit for the e-commerce company by providing coupons to non-returning customers so that they are tempted to make follow up purchases.

Sarjo Das

Parth Singh

Gaurav Puri

Muntasir Alam

Introduction:

Business Analytics (BA) forms a very important component for many of the successful businesses around the globe. But, Business Analytics' definition can be quite ambiguous and it changes continuously. BA can be closely defined as a set of practices, skills, technologies and application for the analysis and investigation of business performance for the achievement of strategic structuring and decision making in the future [1]. The end goal of every company is to enhance and enrich its products and offering for complete customer satisfaction. Different industries have different target markets that are dynamic and their needs often change based on the society. BA helps companies to know the future trend and companies uses this information to bring unique innovative products and ideas which help them to pull ahead in the competition by readily attracting huge number of customers [2].

In Business Analytics, data gathering is crucial which is used in statistical analysis. The result of this analysis helps in decision making. Decision making is very important step which has the potential to break or make the goals of a company. Even a slightest mistake or an overlooked factor can affect a decision by delaying it and has the potential to put an entire business plan to a halt [2].

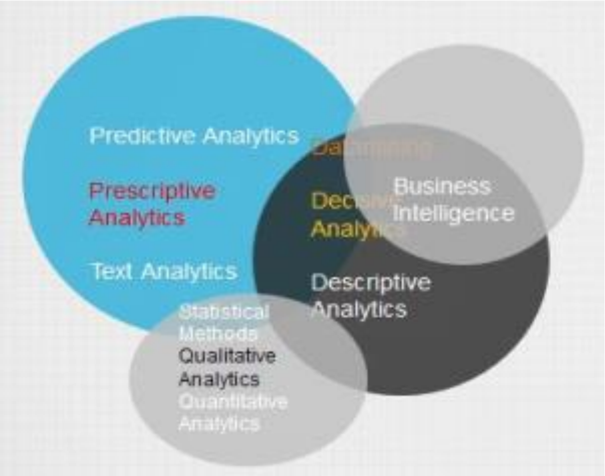


Fig. 1: Business Analytics [3]

The benefits of Business Analytics are many [2]. Some of them are: -

1. Analytics help to measure and analyse the portion of the mission statement that is accomplished. Mission statements are a set of values that are presented as market plans to their customers or as a method of checking their own development. Analytics process helps to quantify these values that leads to the defining of a common goal that is to be followed by everyone associated with the business.
2. Analytics leads to Smart Decision Making Process as access to important data provides companies with the means for accurate decision making which can be used to leverage business. Analytics not only provides critical information, but also allows companies to take decisions faster, more efficiently and effectively than before.
3. Analytics' Data Visualization helps companies to get clear insights about their business. The market information is presented in organised and visually appealing manner which makes it easier to take critical decisions.
4. Analytics also keeps companies updated about their customer's changing needs. Contemporary customers are easily swayed by better offers, they change their mind frequently. Analytics gives

an insight to the companies regarding how the market thinks and acts. Thus, the companies can remain dynamic to cater to the needs of their ever-changing customers.

5. Analytics makes a company more efficient by encouraging a culture of efficiency and teamwork. The ability to collect large amount of data, analysing and presenting them in visually appealing way helps companies to take smart decision to reach specified goals. Employees are also encouraged to share their insights and take part in the decision-making process to make it more efficient.

Predictive Analytics forms an important part of Business Analytics. Predictive Analysis helps to improve customer Engagement and helps to increase the overall company's revenue specially in case of e-commerce companies. Different types of customers usually engage with an e-commerce site in different ways. Predictive Analytics takes into consideration different variables to figure out the desired customer engagement. This engagement may include clinking on a promotion, signing up for newsletter etc [4].

As per a research from Lattice, predictive analytics are being used by big companies like Netflix and Amazon (Fig. 1) to have a sophisticated understand of the customer's behaviour which helps sales professionals to qualify their leads in a better way [4].

John Koetsier on VentureBeat [5] shows the amount ($160 million) was invested in 2014 by venture capitalists for predictive analytics tools to help marketers analyse and understand how to sell effectively offline and online.

Predictive tools like lattice helped Dell to nearly double their results in spite of sending 50% fewer leads. [6]

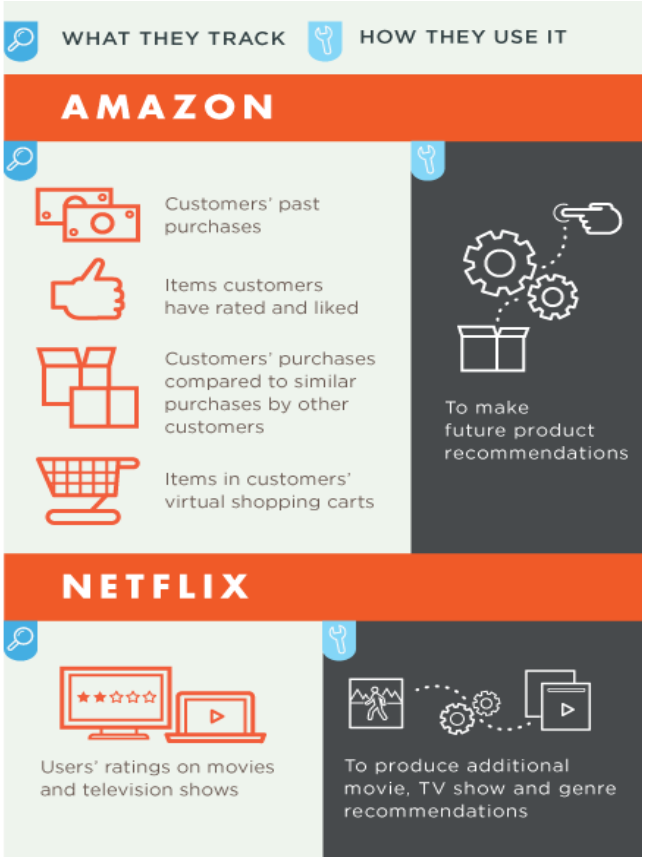

WHAT THEY TRACK
HOW THEY USE IT
AMAZON
Customers' past purchases
Items customers have rated and liked
Customers' purchases compared to similar purchases by other customers
Items in customers' virtual shopping carts
To make future product recommendations
NETFLIX
Users' ratings on movies and television shows
To produce additional movie, TV show and genre recommendations

Fig. 2: Amazon & Netflix- Predictive Analytics [4]

Predictive Analytics can be used for offering solutions [4] for different area such as: -

1. Launching of promotion that are better targeted for a company's customers
2. Maximising profit my optimising price

3. Proactive detection of fraud
4. Improved customer service at a low cost
5. Real time decision making by analysing data.

Problem Statement:

It has been seen that Online customers tend to order from a specific online shop only once. One main goal of CRM (Customer Relationship Management) is to maximize the lifetime value of customers by providing incentives to customers so that they return to shop again. The goal of this assignment as discussed in this paper is to apply different machine learning techniques to build an efficient predictive model to analyze and identify customers who are expected to purchase again in the next 90 days. This will help to target non-returning customers to convince them to return by providing them incentive in form of coupons so that they are tempted to make follow up purchases. The predictive model should be accurate so that coupons are not wasted on returning customers and only specific promising customers are targeted who have a large probability to make re-purchases. For assessing the Model Performance & to maximize revenue a cost matrix is used. The performance of the predictive model is also measured (discussed later in the paper), whose main goal is to maximize revenue.

In the predictive model discussed in this paper all stages of a typical modelling process have been considered and implemented such as data gathering, cleaning and pre-processing of gathered data to the selection of a suitable predictive model and its deployment, assessment. Two data sets have been used for this assignment. Known dataset has a target variable on which the predictive model is trained and applied on the Class dataset for predicting its target values (which are not provided as part of the problem).

Literature Review:

Customer churn has become a major problem for companies due to the rising global competition. Telecommunication industry is quite vulnerable having a churn rate of 30%. Brânduşoiu et al. [7] have discussed predictive analysis methods for the prediction of churns in mobile industry. The predictive models were trained on call details record dataset to predict customer churn and principle component analysis algorithm was applied initially for the reduction of the data dimensionality and elimination of the multicollinearity problem. Ultimately three machine learning algorithms namely Neural Network (NN), Support Vector Machines (SVM) and Bayesian Networks were applied on the dataset and the models were evaluated by ROC curve, confusion matrix and gain measure.

The models from a technical point of view gave overall accuracy of 99.10%, 99.55% and 99.70% for Bayesian Networks, Neural Networks and Support Vector Machines respectively for predicting both churners and non-churners. Esteves et al. [8] tries to prediction churn rate by applying and comparing 6 machine learning algorithms. The algorithms used are KNN, Naïve Bayes, c4.5, Random Forest, AdaBoost and ANN. The algorithms were applied on real world data. The models were evaluated on criteria such as AUC, sensitivity and specificity. The result found that Random

Forest model achieved the highest ROC of 0.9915 and Sensitivity value of 0.9110. KNN and C4.5 came in second and third place.

The paper by Li et al. [9] talks about the main difficulties faced while modelling customer churn prediction. Firstly, data set for customer churn is significantly imbalanced in real world. Secondly, the samples that are present in the feature space are comparatively scattered. Thirdly, the feature space dimension is very high and considerable dimension reduction is required for the algorithm efficiency.

The authors, to overcome the above-mentioned difficulties have proposed a new method to pre-process the dataset using a supervised one sided sampling technique. Data set are clustered meaningfully using K-means method and then from each cluster noise and redundant negative samples are removed by applying one sided sampling. Further dimensional reduction and selection is important variables is done using Random Forest. C5.0 decision tree is ultimately applied to predict the customer churn. The model provides a satisfactory prediction result with a precision ration of 80.42% and recall ration of 52.43%.

The paper by Xia et al. [10] sets up an ensemble algorithm consisting of Bayes, Artificial Neural Networks, Decision Tree and Support Vector Machine on imbalanced real world telecom dataset for churn prediction. The ensemble model has advantage especially as the Support Vector Machine are base classifiers this leads to better hit rate, lift coefficient & accuracy rate.
Lu et al. [11] uses weight assigned by the boosting algorithm to separate customers into two clusters. This results in the identification of the higher risk customers. Logical regression is then applied as a basic learner and churn prediction model on each of the two clusters. The result shows that compared to a single logistic regression model boosting provides effective separation of churn data and hence boosting can be a valuable tool for churn prediction analysis.

In the field of e-commerce also churn rate is very high. To improve the accuracy of churn prediction and get the identity of non-churn customers Wu et al. [12] discussed a model for churn prediction on imbalanced dataset based on improved SMOTE which consists of the combination of oversampling and under sampling methods to take care of the imbalance problem and integrates AdaBoost algorithm for prediction. The result shows that this model compared to other mature customer churn prediction algorithms have better efficiency and accuracy.

Methodology:

After understanding the project assignment in detail the first thing was to review about the various prediction models and get a brief idea on how they work and how should they be implemented.

We kept our approach very flexible in terms of choosing the features as well as the parameters while applying these models.

In order to keep the code as clean as possible without making it lengthy, we created several different R scripts for different purposes such as for loading binaries we created a separate script, helper function for both known and class datasets are in a separate script.

Decision regarding which prediction models will be tried was not easy as we started with the most basic one, such as logistic regression and kept on moving towards the more competent algorithms towards the middle and end stages.

While testing various algorithms, we were flexible with the parameters used within the model. We tried each model with different parameters in a recursive manner.

Iterative process:

In the initial stages when the models were trained, it was noticed that the result was not coming as per our expectations. So, it became necessary to identify methods which will tell on what features the return customer is highly dependent and on which it does not depend at all. Once we had the result from the feature selection process, high AUC value was witnessed. So now, the next step was to add and remove certain features depending upon their relevance. We noticed that adding/removing some features affected the value of AUC. Thus, we had better interpretation and understanding of results from the feature selection methods and prediction models. This will be further noticed in the competition matrix which is discussed in later section of this report.

Experimental Design:

Use of split-sample setup:

Before running any model to give us the prediction on the dataset provided it had to setup the model with a training dataset to know how well the model was doing and what parameters it needs to select in order to give the maximum result. This is done because the class dataset does not have any column named return customer and hence it was not sure that the prediction the model gives is correct or not. Hence we had to first train the model on known dataset where it can evaluate the return customer prediction. However, after training we also have to test the model to see how accurate the evaluation is. Therefore, to create a train and test model we split the known dataset into 90/10 parameter [15]. The model is trained on 90% of the dataset in the train file then, it cross validates the result with the remaining 10% of the data.

Use of cross validation:

Cross Validation is a model evaluation technique used in Data Analytics. It is better than the previously used method called residual [14]. The reason that cross validation is preferred over residual is, residual evaluations did not tell the user about the predictions it will give based on the model applied for the dataset which it has not already seen. In cross validation this problem is solved [13]. Before training the model we split the data into 90/10 ratio. Then when training is done, the data that was removed can be used to test the performance of the learned model on "class"' data. This is the basic idea for a whole class of model evaluation methods called *cross validation*.

Types of Cross Validation:

The simplest kind of cross validation method is called the **Handout Method.** The dataset as usual is divided into two sets training and test [14]. Only the training dataset has the function approximator into it. In test dataset, this function approximator is asked to predict the output values. The advantage is that it uses less time to compute however, the disadvantage is that its evaluations have a high variance.

A better method to improve this situation is to use the **K-fold cross validation**. In this the dataset is divided into k subsets and the holdout method is applied k times [14]. The advantage of using this method is that, it does not matter how the data is divided into training and test parts. As every data subset gets to be in test set exactly once and in training set k-1 times. The disadvantage is that it takes k times more to compute than the holdout method. To keep the variance as low as possible value of k should be high.

Another modification of k-folds is to use **leave-one-out cross validation.** This method takes k-folds to its extreme. In this, the value of k is kept equal to N (number of data points in the dataset) [14]. That means that N separate times, the function approximator is trained on all the data except for one point and a prediction is made for that point. The advantage is that it gives the best result possible. However, the disadvantage is that it is very expensive to compute.

In our project we have used the k-fold cross validation method.

Data Preparations:

The most important part of any machine learning model is to clean the data before it can be used with any prediction model. The data which is generated and provided to any data scientist is usually raw, which mean it contains a lot of missing values, outliers and noisy data as well. It is a job of a data scientist to go through the data and figure out what he needs in order to get a better accuracy from the various machine learning algorithms.

For this BADS assignment we are provided with two dataset files (known and class) in .csv format. The first task was to clean the given known dataset and apply the same cleaning methods to the class dataset. After loading the dataset in R studio we first checked what type of dataset is provided to us. For this I ran the "***sapply***" command to check the default class of data frames given in the known file. All the data frames were of two types: integer and factor. Upon reading both the files we got to know that the files contain a lot of missing values in the form of NA in various data frame fields.

The very basic step of any data cleaning process is to either remove all the missing values from the dataset (which is not a very effective way of dealing with missing values) or replace them with the median or mean value of that data frame.

Basic approach to handle missing values:
At first just for testing we started by removing all the NA's from the known dataset. We hence programmed our helper function to ***"delete the rows which contain NA"*** value in any column. Removing all the missing values from the entire dataset caused the dataset to shrink and this resulted in a low AUC (area under the curve). Hence the resultant accuracy from the various machine learning models was also low.

Sophisticated strategy to handle missing values:
Now we tried the other approach of replacing the NA values in our dataset. To achieve our goal, we first had to know which all data frames in the dataset contain NA's. We used the powerful R programming language to figure out that only four data frames contain NA's:

(i) form_of_address
(ii) deliverydate_actual
(iii) account creation date
(iv) weight

Continuous variables have been scaled:
Replacing NA values in weight was the easiest as it was of class integer, we quickly run the summary command to know the mean value of the data frame and replace all NA values with the sum of mean value and the upper bound value (upper quartile + 1.5*IQR). The reason for following this sophisticated approach was because in most of the models where the missing value NA and outliers in a column is needed to be replaced by not the mean but also with the upper bound limit such that a consistency between the dataset is maintained. Upper bound limit is calculated as the sum of upper quartile value and 1.5 time the inert quartile range (IQR, which is the difference between the upper quartile value and the lower quartile value). To confirm the presence of outliers

in the weight data frame we followed the method suggested in the slides. We plotted a boxplot for the weight data frame and the presence of outliers can be observed in the image below.

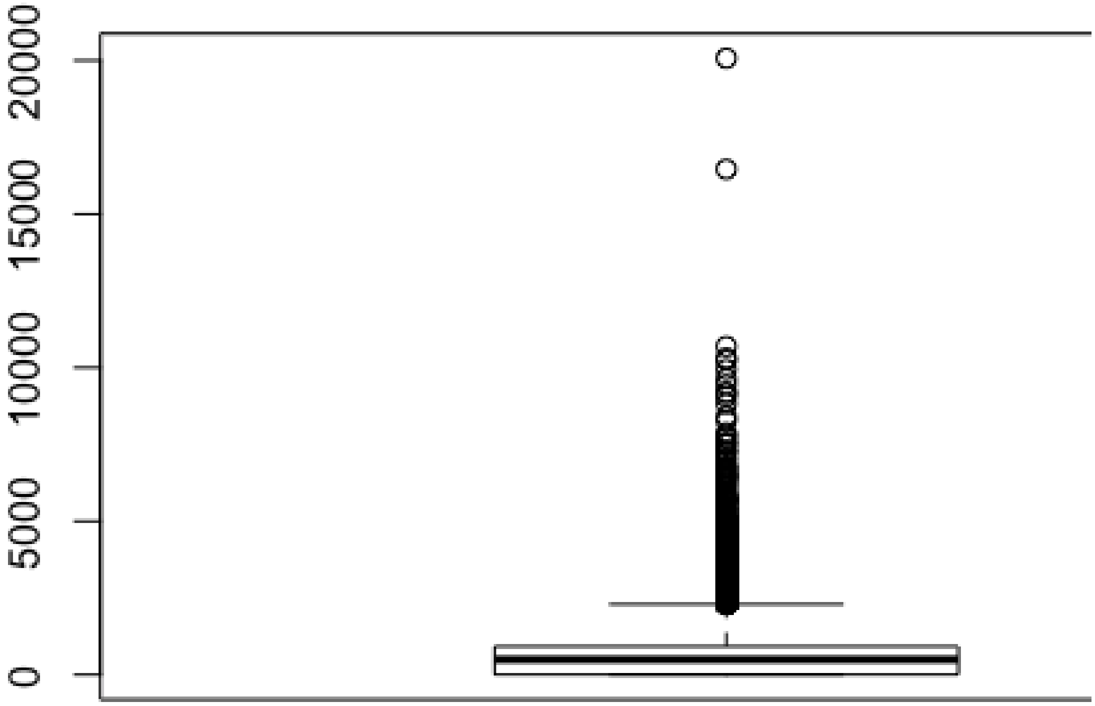


Fig. 3: Boxplot for weight data frame on known dataset

After converting these outliers to NA in the data frame to check the summary again and know that the difference between mean and median is less as compared with previous result. The data frame now had only the missing values which were replaced with the sum of mean and the upper bound value of the data frame. Applying the same steps in class dataset for weight data frame and we are finished with the cleaning process for weight data frame in both known and class datasets.

Suitable treatment of categories:

Cleanly the account creation date, deliverydate_actual and deliverydate_estimated was the tricky part. As these data frames where in the form of ***"factor"*** I first changed the class of these data frames to ***"Date"*** format. The reason for changing the class type was to take advantage of more powerful R packages (such as dplyr and lubridate) for date cleaning process.

Reading the account creation date data frame, it was clear that there are a lot of missing values which were present in the data frame. Upon analyzing the dataset, we got to know that most of the times in the dataset, the account creation date has the same value as order date. So it was decided that all the missing values in account creation date will be replaced by order date. This way the account creation date data frame was cleaned in both known and class datasets.

Sophisticated treatment of categories:
Reading the deliverydate_actual data frame, it was clear that there are a lot of missing values in the form of 0000/00/00 entries and it also had outliers. As discussed earlier when we converted this data frame from ***"factor"*** class to ***"Date"*** class, this changed all the 0000/00/00 entries to NA's. We than matched these missing entries with data frames of audiobook_count, ebook_count and audiobook_download_count. The reason for doing this was that if the purchased item is a downloadable item than the deliverydate_actual will be equal to order date itself. So it was decided that where the value in the above three mentioned data frames is more than 0, replace NA in deliverydate_actual data frame by order date.

With the above logic we were able to replace most of the missing values but not all. In order to replace the remaining missing values along with the outliers it had to replace them with the sum of mean value and the upper bound value of the data frame. In order to calculate the average delivery date, we first had to minus the delivery date actual data frame with order date. This result is stored in a vector named date_diff. than we calculate the mean of this data frame and replace all the missing values and outliers with order date + the sum of mean value and upper bound value (in number of days' format). This way the cleaning of delivery date actual data frame in both the dataset was performed.

Dealing with the deliverydate_actual data frame made us realize the presence of outliers in deliverydate_estimated data frame. We cleaned it using the same approach. Took the difference between it and the order date and stored the result in a vector named clean_date_diff. We then calculated the mean number of days and replaced the outliers with order date + the sum of mean value and upper bound value.

To this point most of our dataset has been cleaned and only form_of_address data frame was left. As this was in factor class and with no dependency, we simply replaced all the missing values in this data frame with the value "others". With this last step done both our known and class datasets were now cleaned on the same parameters.

Variable Selection:

Variable Selection is one of the most important steps of model building. In our case, we have 37 predictive variables in our dataset. Removing unnecessary variables from the analysis model will increase our model accuracy whereas including them in the model will decrease model accuracy and will result in a flawed prediction. So, in order to craft a better model, we have used all the following algorithms to perform exploratory analysis in order to see which variables are the most significant in crafting a prediction of return customer.

Boruta Analysis (Random Forest Algorithm):

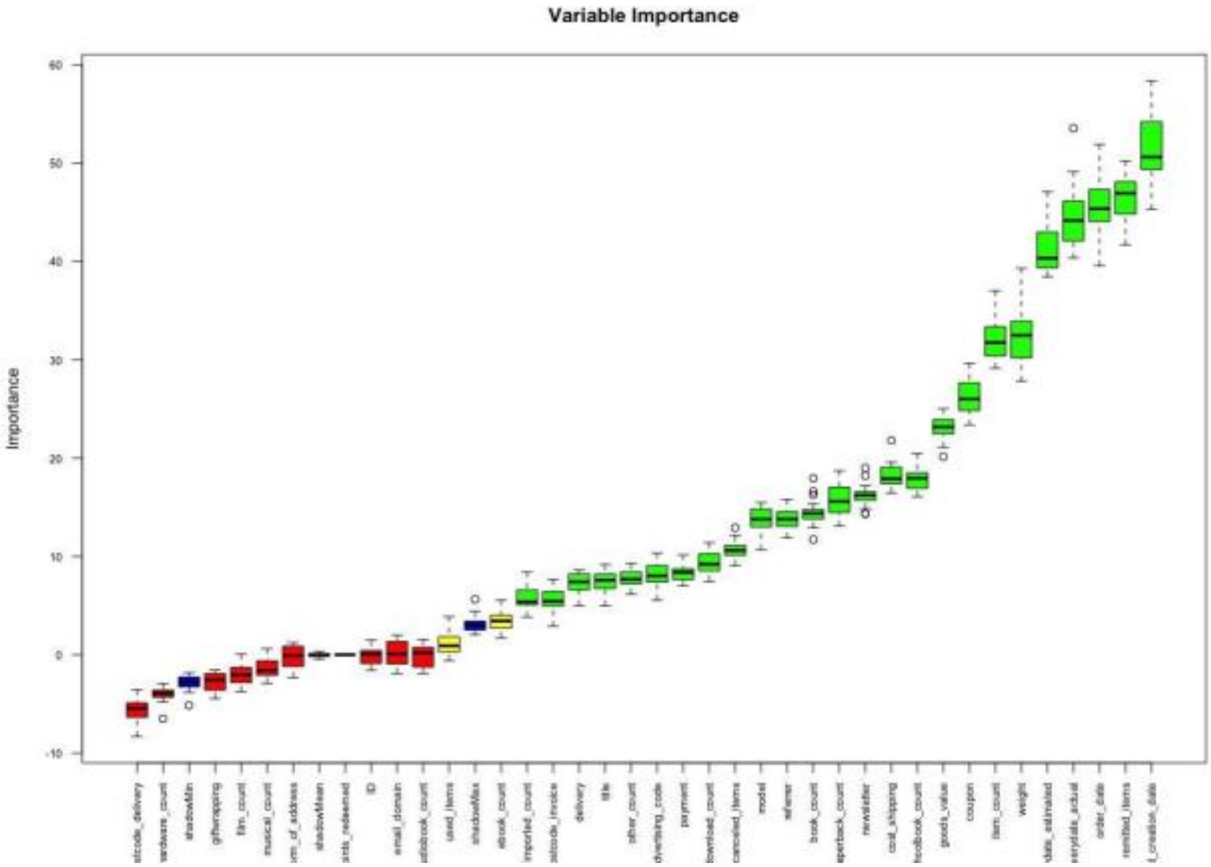


Fig. 4: Variable importance through Boruta

We have performed Boruta on all our available variables. As it is a wrapper around Random Forest algorithm, it builds multiple trees and provides us with the variables which are significant. As we can see, Boruta makes a very clear output on which variables are important. It shows 25 variables as having high importance, 4 variables as having 0 importance and others as having negative importance.

Multivariate Adaptive Regression Splines (MARS):

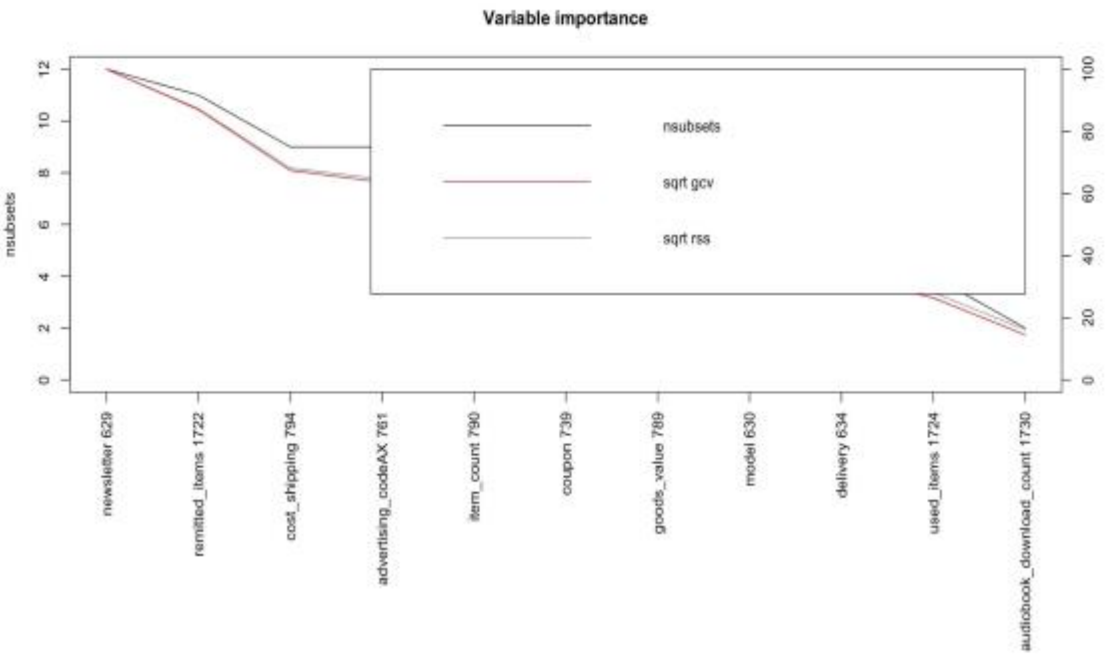
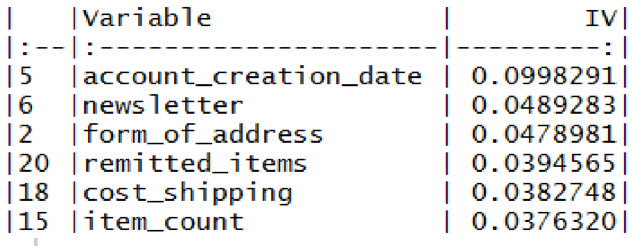


Fig. 5: Variable importance through MARS

We performed Multivariate Adaptive Regression Splines(MARS) on our data in order to find significance of the dependent variables. MARS is a regression algorithm which uses non-parametric regression analysis that finds interactions between variables. MARS is particularly useful if there are complex relationships between the data variables. From the graph we can see that the MARS algorithm has assigned the variables newsletter, remitted_items, cost_shipping, advertising code, item_count, coupon, goods_value, model, delivery, used_items, audiobook_download_count as particularly significant. We take all of these variables in our final model except advertising_code, which has been as not particularly useful through multiple iterations of testing.

Information Value (IV):

We also run Information Value(IV) on our variables. Broadly speaking, IV provides us with an idea of how well our variable will be able to see between a binary value of a target variable-in our case return_customer(1=customer places another order within 90 days, e.g returning customer, 0=not returning customer). If a variable has low information value, usually it will not be able to classify our target variable, and thus we discard it as a non-significant variable. From our IV analysis, we can see that it has deemed the following six variables as of high significance- account_creation_date, newsletter, form_of_address, remitted_items, cost_shipping and item_count.



```
|   |Variable             |        IV|
|:--|:--------------------|---------:|
|5  |account_creation_date| 0.0998291|
|6  |newsletter           | 0.0489283|
|2  |form_of_address      | 0.0478981|
|20 |remitted_items       | 0.0394565|
|18 |cost_shipping        | 0.0382748|
|15 |item_count           | 0.0376320|
```

Fig. 6: Results of the IV analysis

Stepwise Regression:

We have performed stepwise regression analysis in our exploratory analysis. The function step() can be used to perform a stepwise regression on our data which gives us a list of variables which are best fit for the model. Stepwise regression is an automatic method in which the elimination of variables is carried out automatically. In each step of the algorithm, we can either add or subtract a variable to reach our result variable list, using a filter test. This test can be chi-square test , F-test, t-test or some other suitable algorithm. We use Akaike information criterion (AIC). AIC is a way of measuring relative quality between a set of models. The stepwise algorithm recommends 19 variables to us - newsletter, cost_shipping, remitted_items, advertising_code, delivery, item_count, audiobook_download_count, model, used_items, coupon, ebook_count, paperback_count, weight, goods_value, other_count, order_date, deliverydate_actual, imported_count, audiobook_count.

Based on all of the previous Analysis, we have determined that the following about our dependent variables -

newsletter	This nominal variable shows whether a customer signed up for a newsletter or not. Intuitively we know this will have significance on whether a customer will return or not. Our exploratory analysis also confirms this. So, we are going to include this variable in our model.
cost_shipping	From analysis, we see that this is a significant variable. We include this variable in our model.
remitted_items	From our analysis, we can see that this is also an important variable. We include this variable in our model.
delivery	From our exploratory analysis, we can see that this is also an important variable. We include this variable in our model.

item_count	This variable shows the number of items a customer has bought. Obviously, this variable will have an effect on our dependant variable. So, we include this variable in our final analysis.
account_creation_date, deliverydate_estimated ,deliverydate_actual, order_date	Our exploratory analysis shows that these date variables have high level of importance and thus will be included in our model.
coupon	This variable shows whether a customer used coupon at a particular order or not. Since coupon usage shows customer engagement, this variable has high importance and thus will be included in our model.
referrer	From our analysis, we can see that this is also an important variable. We include this variable in our model.
goods_value	This shows the value of the product the customer has bought between a scale of 1 to 5, 5 being the highest.
schoolbook_count, paperback_count ,audiobook_count, book_count, ebook_count , audiobook_download_count	These variables show various types of purchased items and their count. These have been included in our final model based on our exploratory analysis.
imported_count	From our analysis, we can see that this is an important variable. We include this variable in our model.
email_domain	This variable shows the domain of the customer's email provider. We also include this in our model.
postcode_invoice	This shows postcode of where an invoice for a purchase has been sent. We have included this variable in our model.
used_items	This shows the number of used items. In our analysis, we see that it's a significant variable. Included in our model.
model	This variable shows the website design type shown to customer. Included in our model.

Tools used

We have used the following packages in our analysis:

Boruta [16]

Boruta is a package which implements a feature selection algorithm centered around the random forest algorithm. When we have a lot of predictor variables we can use Boruta to select our variables of interest. It's a very useful algorithm to work with as it follows a all-round approach to selecting the variables, showing all the different levels of relevance of the selected variables. Boruta gives the user all the levels by which variables are highly relevant or weakly relevant to the decision variable.

pROC [17]

We have used this package to show our ROC plots and to show the Area under the Curve(AUC) plots.

Random Forest [18]

Random forest is a robust decision forest algorithm that uses a number of decision trees to output a class that is a mode or mean prediction of the singular trees. It grows many classifier trees, of which each tree outputs a classification. By this classification we count total number of 'votes' for that single class. Ultimately the forest chooses the classification which has the the most number of votes. Random forest algorithm is a very accurate one and it is suitable for large models, which goes perfectly with our case.

Xgboost [19]

Xgboost is an acronym for the term eXtreme Gradient Boosting package. It is basically an efficient implementation of a Tree Boost algorithm. It's a very efficient and scalable algorithm and its very widely used because of its versatility. It's also capable of parallel computation on multicore processors and thus is extremely fast.

Nnet [20]

We have used the nnet package to run a single layer neural network on our data.

doParallel [21]

We have used the doParallel package to accelerate our algorithms in real time. This package allows to take advantage of a machine of multiple processor cores. In its origin, it's a parallel backend for the for each package-it allows forking of multiple for each loops on a system with multiple processors or multiple cores or both.

Lubridate [22]

We have used this package to clean up date values in our data files. The ‘lubridate’ package allows us to work with date values in R. It allows us to do algebraic operations on date variables and this makes it easy to work with dates in our data.

Information Value [23]

We used this package during our variable selection phase to find out the information value of the variables. Using this we can select the variables with high information value and potentially filter out some with low information value.

Model development:

After performing a set of pre-processing functions over the feature set and then splitting the data for cross-validation, we applied different prediction models which led to different results.

Sensible set of algorithms has been tested:

In the starting, simple cleaning and simple prediction models were performed with their default values.

Logistic regression: Logistic regression was our first choice because the dependent variable (i.e., the response) in this model is categorical. Logistic regression comes under generalized linear model (GLM) where the prediction model is trained over a set of independent variables to give a probability between 0 and 1. As this was the first model used, the data supplied to this training model was premature, i.e., only NA’s were omitted and outliers were not treated in data cleaning. In the initial stage, only few features were selected for training model. The result we were expecting has to be either 0 or 1, i.e., binomial in nature, so the method used for training model is “logit (binomial)”. The group unanimously took an assumption that logistic regression will give a better training model if the independent variables were also categorical in nature. However, the results that we got after training the model was far below our expectation.

Random forest: Random forest played a crucial part in our learning about various training models. Random forest was used in order to deal with bias and variance trade-off. Decision tree has low bias but significantly high variance. On the other hand, random forest reduces the high variance by averaging the set of decision trees. However, there is a slight increase in bias in order to significantly reduce the variance. Initially, we performed random forest without caret package in order to understand the insight of the model. Random forest uses a technique known as bagging or bootstrap aggregation. In bootstrap aggregation, various training sets are bagged randomly along with their responses with replacement. Bagging reduces variance but there is slight increase in bias. Increase in bias is because of unused 1/3rd training set for training the model due to continuous replacement policy in bagging. We performed random forest over the features which were of high importance to return customer. Few subsets of features were calculated by using various variable importance model such as MARS, boruta, stepwise regression feature importance technique and information value gain. Later on, the group used prediction model Random Forest over CARET package. CARET package allows us to exploit various parameters for defining the boundaries of models directly. In trainControl(), we defined method as cross validation along with other parameters such as number of resampling folds, return type should be class probabilities and allow parallelism. Expand.grid is used for manually tuning the parameters in training model. In manually tuning the model, all the combinations that should be tried can be specified as parameters. After performing the training model, we drew several graphs for boxplot, Specificity vs Sensitivity, Area under ROC curve in order to understand the accuracy of our model.

Neural Networks: Another significantly important model was neural networks which comes under "nnet" package. The group took the same approach for Neural networks and trained the model firstly without using the CARET package and then by using the CARET package. The fanciest part of Neural Network model without using the CARET package was the visible iterative cross validation within the training dataset. This learning model takes a lot of time for convergence. However, the convergence and the AUC was not equivalent to our expectation. Such bad result questioned the data pre-processing techniques used by us. We did change few parameters and again performed neural networks. The prediction improved but not significantly. Moving to Neural networks with CARET package, we defined trainControl and expand.grid in similar fashion as done for Random forest. Over applying the trained model on the test data set, we plotted graphs and calculated confusion matrix and compared our result to different models.

Gradient Boosting: The gradient boosting model was the model which gave us the best prediction in all our cases (even better than trained ensemble models). Gradient boosting helps in regression and classification by creating a trained prediction model by ensembling different weak prediction models. Gradient boosting model comes under XGBoost package. The convergence time of XGBoost is very less even with high number of features. Similar to Random forest and neural networks, we can define the trainControl() and expand.grid parameters for better tuning and improved prediction. It also delivers a model which returns class probabilities and later on can be turned into categorical results. An average data cleaning before gradient boosting can give you considerable results. The group performed various other pre and post processing over this model in order to get a result with least false negatives and false positives.

Model choice convincingly explained: (e.g. (Dis)Advantages identified, diversity in model types)

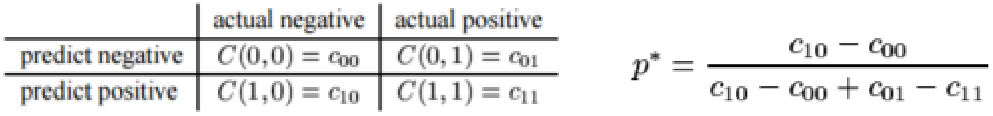
We chose various models as prediction models; however, not able to achieve a very high accuracy in predictions.

Logistic regression had its own disadvantage of considering a linear relationship between independent variables and dependent variables. Furthermore, it was after a while the group came to know that logistic regression can give a misleading yet good prediction results for continuous variables. In Neural Network prediction model, it's a cumbersome process to train a multi layered model. Parallel to this, the convergence time of this model is quite high. So it's exhausting to try different data cleaning process over the features and repeatedly preparing the model. Neural Network is a black box model; it is hard to get a deep insight of which parameters are treated in what manner.

In gradient boosting model, we have used various techniques such as oversampling, undersampling and cost sensitivity approaches to give weight to false negatives and false positives in order to reduce the misclassification. Moreover, we have used ROSE package to perform over and undersampling by synthetically generating the balanced dataset. Better results were expected as the return_customer column was quite imbalanced (80% value were 0 and 20% values were 1). However, performing the over and undersampling didn't provide us the same. Performing explicit bagging to balance the dataset should improve the prediction accuracy was also not witnessed. The best results were still the one which was XGB over the preprocessed data.

Use of cost matrix:

We used a cost matrix function [24] to calculate the threshold in order to give the highest accuracy. This means after the prediction model provides us the predicted responses in terms of probability, the value above threshold will be considered 1 and value below the threshold will be considered as 0.



	actual negative	actual positive
predict negative	$C(0,0) = c_{00}$	$C(0,1) = c_{01}$
predict positive	$C(1,0) = c_{10}$	$C(1,1) = c_{11}$

$$p^* = \frac{c_{10} - c_{00}}{c_{10} - c_{00} + c_{01} - c_{11}}$$

The above mention formula gave us 0.42 as the threshold value for the cost matrix given in the assignment. It gave us a better accuracy but false negatives were still a lot. So, by hit and trial, we found that 75^{th} quantile was good threshold value to reduce the false negatives significantly and maximize the revenue by sending the coupons to non-repurchasers.

Some ensembling approach tested:

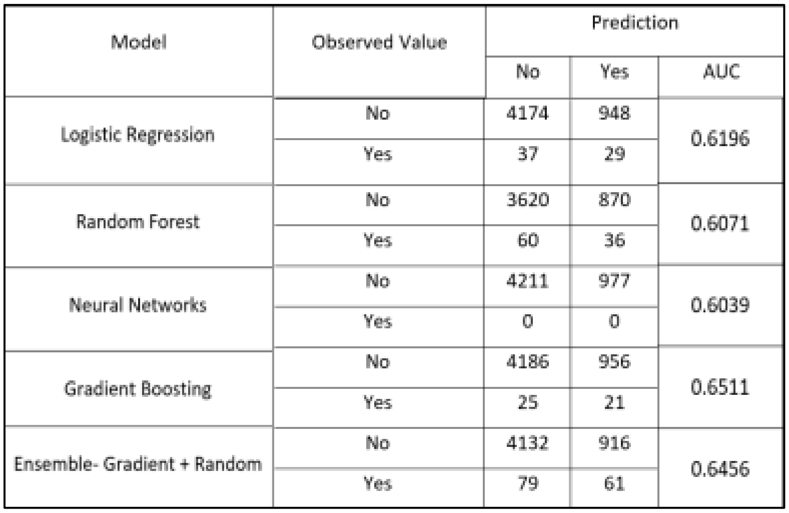
Gradient Boosting in itself is an ensemble model over weak prediction model such as decision trees. Another homogenous ensemble model was Random Forest. Heterogeneous ensemble approaches were also used in the project such as Ensemble over Random forest and Gradient Boosting, Ensemble over Random Forest, Neural Networks and Gradient Boosting. These models are taken as Base models and a stacking classifier is appended over these base models to provide an improved prediction. The stacking classifier used in our case was GLM. However, we were not able to perform the ensemble model over all the features as the time of convergence was too high and the devices were crashing with "Memory: time-out" error. So we chose few important variables and performed ensembling approach.

Result Analysis:

The trained model was then used to predict return customer in test dataset. We have taken 10% of the whole dataset as test dataset.

Notation: No: Non-repurchaser/Coupon, Yes: Repurchaser/No-Coupon, AUC: Area under ROC curve.

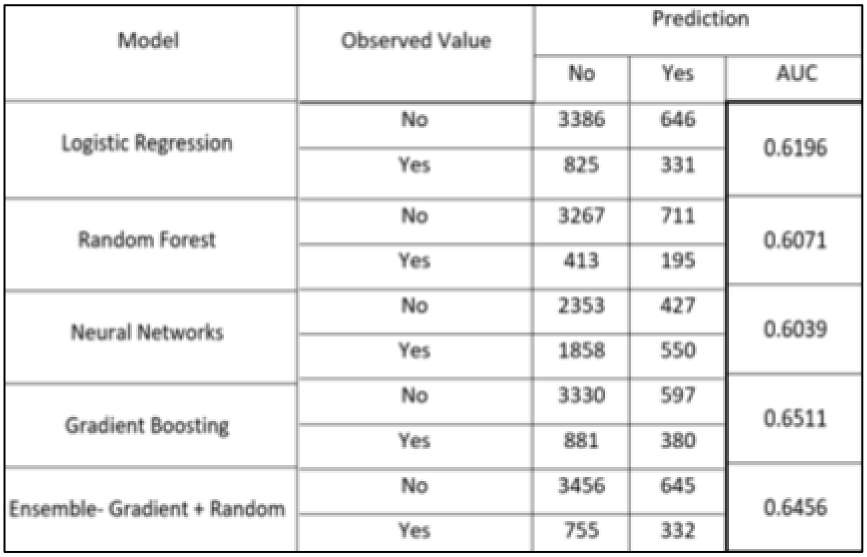
First table contains the use of cost matrix to decide the cutoff and improve accuracy. But the increase in accuracy was at the cost of higher false negatives.



Model	Observed Value	Prediction		
		No	Yes	AUC
Logistic Regression	No	4174	948	0.6196
	Yes	37	29	
Random Forest	No	3620	870	0.6071
	Yes	60	36	
Neural Networks	No	4211	977	0.6039
	Yes	0	0	
Gradient Boosting	No	4186	956	0.6511
	Yes	25	21	
Ensemble- Gradient + Random	No	4132	916	0.6456
	Yes	79	61	

Table: 1

However, in second table, the threshold to convert class probabilities into categorical values was near 75th quantile. In this table the accuracy was less but the revenue gain was more than before.



Model	Observed Value	Prediction		
		No	Yes	AUC
Logistic Regression	No	3386	646	0.6196
	Yes	825	331	
Random Forest	No	3267	711	0.6071
	Yes	413	195	
Neural Networks	No	2353	427	0.6039
	Yes	1858	550	
Gradient Boosting	No	3330	597	0.6511
	Yes	881	380	
Ensemble- Gradient + Random	No	3456	645	0.6456
	Yes	755	332	

Table: 2

It is clearly evident from the above table that maximum AUC achieved was in gradient boosting and the maximum gain in revenue is also under gradient boosting. And this gain in revenue is not only because of gain in True negative and True positive but also by reducing false negatives.

Acknowledgement:

We would like to thank Prof. Stefan Lessmann for accepting us to the course and continuously guiding us throughout this semester through lectures and tutorial classes. We would further like to thank Mr Johannes Haupt for providing us all sorts of possible help including clearing our concepts and doubts which made completion of this assignment possible.

Conclusion:

Predictive Analysis has become very important aspect for online retails specially e-commerce companies like Amazon, ebay, Zalando etc to succeed in this global competitive market and should be given special consideration. The benefits of predictive analytics were utilized by Macy's [25] through the deployment of a solution made by SAP that resulted in improved targeting of the website's registered users. It was seen that within a period of 3 months there was an 8-12% increase in Macy's online sales which was achieved by the combination of browsing behaviours within the product categories and by sending each customer segment targeted emails. Predictive analytics when combined with big data helps change the way inventory is currently managed [26]. However, it should be remembered that proper deployment of Predictive Analytics is very important to get benefits from it. The most common mistakes take place during the predictive analytics solution deployment from non- removal of junk data, not having the end objectives clear etc [27].

Predictive Analytics as we have seen brings a lot of benefits when properly implemented. The benefits of Predictive Analytics are usually seen after some time, so it is very crucial that the deployed models are continuously monitored and refined and at the same time waiting for the benefits that business gets from this new feature.

References:

[1] The Importance of Business Analytics: A Look Ahead to 2014
http://www.consultparagon.com/blog/the-importance-of-business-analytics-a-look-ahead-at-2014
Last accessed on 10/02/2017

[2] 5 Benefits of Using Business Analytics
http://www.datamensional.com/5-benefits-of-using-business-analytics/
Last accessed on 10/02/2017

[3] What is Business Analytics? Business Analytics Tools
http://www.predictiveanalyticstoday.com/business-analytics/
Last accessed on 10/02/2017

[4] How Predictive Analytics Is Transforming eCommerce & Conversion Rate Optimization
https://conversionxl.com/predictive-analytics-changing-world-retail/
Last accessed on 10/02/2017

[5] Predictive apps are the next frontier of marketing tech (infographic)
http://venturebeat.com/2014/07/14/predictive-apps-are-the-next-frontier-of-marketing-tech-infographic/
Last accessed on 10/02/2017

[6] How Dell Predicts Which Customers Are Most Likely to Buy
http://blogs.wsj.com/cio/2012/12/05/how-dell-predicts-which-customers-are-most-likely-to-buy/
Last accessed on 10/02/2017

[7] Brânduşoiu, I., Toderean, G., & Beleiu, H. (2016, June). Methods for churn prediction in the pre-paid mobile telecommunications industry. In Communications (COMM), 2016 International Conference on (pp. 97-100). IEEE.

[8] Esteves, G. C. (2016). Churn prediction in the telecom business.

[9] Li, H., Yang, D., Yang, L., & Lin, X. (2016, October). Supervised Massive Data Analysis for Telecommunication Customer Churn Prediction. In *Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications*

(SustainCom)(BDCloud-SocialCom-SustainCom), 2016 IEEE International Conferences on(pp. 163-169). IEEE.

[10] Xia, G. E., Wang, H., & Jiang, Y. (2016, November). Application of customer churn prediction based on weighted selective ensembles. In *Systems and Informatics (ICSAI), 2016 3rd International Conference on* (pp. 513-519). IEEE.

[11] Lu, N., Lin, H., Lu, J., & Zhang, G. (2014). A customer churn prediction model in telecom industry using boosting. *IEEE Transactions on Industrial Informatics*, *10*(2), 1659-1665.

[12] Wu, X., & Meng, S. (2016, June). E-commerce customer churn prediction based on improved SMOTE and AdaBoost. In *Service Systems and Service Management (ICSSSM), 2016 13th International Conference on* (pp. 1-5). IEEE.

[13] An Introduction To Cross-Validation
https://www.salford-systems.com/videos/tutorials/how-to/an-introduction-to-cross-validation
Last accessed on 10/02/2017

[14] Cross Validation
https://www.cs.cmu.edu/~schneide/tut5/node42.html
Last accessed on 10/02/2017

[15] Fung, G., Rao, R. B., Rosales, R., Apte, C., Park, H., Wang, K., & Zaki, M. J. (2008). On the dangers of cross-validation. an experimental evaluation (SIAM). In *Proceedings of the 2008 SIAM International Conference on Data Mining* (pp. 588-596).

[16] "Package 'Boruta' - R Project." *The Comprehensive R Archive Network.* Web. <http://cran.r-project.org/web/packages/Boruta/Boruta.pdf>.
Last accessed on 10/02/2017

[17] “CRAN - Package pROC.”
< https://cran.r-project.org/web/packages/pROC/pROC.pdf >.
Last accessed on 10/02/2017

[18] “CRAN - Package randomForest”
< https://cran.r-project.org/web/packages/randomForest/randomForest.pdf >.
Last accessed on 10/02/2017

[19] “Package 'xgboost' - R Project”
<https://cran.r-project.org/web/packages/xgboost/xgboost.pdf>.

Last accessed on 10/02/2017

[20] “Package 'nnet' - R Project”
<https://cran.r-project.org/web/packages/nnet/nnet.pdf>.
Last accessed on 10/02/2017

[21] “Package 'doParallel' - R Project”
<https://cran.r-project.org/web/packages/doParallel/doParallel.pdf>.
Last accessed on 10/02/2017

[22] “Package 'lubridate' - R Project”
<https://cran.r-project.org/web/packages/lubridate/lubridate.pdf>.
Last accessed on 10/02/2017

[23] “Package 'Information' - R Project”
<https://cran.r-project.org/web/packages/Information/Information.pdf>.
Last accessed on 10/02/2017

[24] Elkan, C. (2001, August). The foundations of cost-sensitive learning. In *International joint conference on artificial intelligence* (Vol. 17, No. 1, pp. 973-978). Lawrence Erlbaum Associates Ltd.

[25] Macy's boosts Web sales, email marketing with predictive analytics
http://www.fierceretail.com/operations/macy-s-boosts-web-sales-email-marketing-predictive-analytics
Last accessed on 10/02/2017

[26] Waller, M. A., & Fawcett, S. E. (2013). Data science, predictive analytics, and big data: a revolution that will transform supply chain design and management. *Journal of Business Logistics*, *34*(2), 77-84.

[27] 12 Predictive Analytics Screw-Ups
http://www.predictiveanalyticsworld.com/patimes/12-predictive-analytics-screw-ups/
Last accessed on 10/02/2017