COSC 2670 | Practical Data Science

Assignment 2: Data Modeling and Presentation

Binary Classification - Online shopper's purchase intention

28th May, 2019

Vishwa Gandhi - s3714805
Jigar Mangukiya - s3715807
Master of Data Science, RMIT University

Contents

1. Abstract

The project demonstrates an end to end data modeling project, roughly following CRISP-DM Process guidelines. Starting off with acquiring data for online shopper's purchase intention, the project follows standard data modeling process, moving through data cleaning, transforming, exploration, modeling and performance analysis. Focus of study is on exploring the data and using different classification techniques and comparing them. We end up using two types of machine learning algorithms, information-based algorithms (decision tree & random forest) and similarity based algorithm (K-nearest Neighbors). We evaluate performance of these algorithms based on 'Area under the Curve' metric in the concluding part of the report. The project is implemented in python 2.7 and 'Jupyter Notebook' is the IDE used to perform the analysis.

2. Introduction

The project can be divided into three major parts – Data Loading & Preprocessing, Data Exploration, & Data Modeling.
In part 1, we load the data and do initial checks to ensure the data has been loaded correctly. Subsequently, we clean the data by removing null values & outliers, inspect the summary statistics to better understand the data and resample to balance out the distribution of target feature's levels.
In part 2, we use visualizations and other methods to explore the underlying relationships among features of the dataset. We use univariate as well as multivariate visualizations. We state some plausible hypothesis from the relationships that we observe among the feature from the visualizations.
In part 3, we encode the data and feed it to various machine learning classification algorithms and evaluate their performance.

3. Data Set Introduction

Data set

Data set used in this report is 'Online shopper's purchasing intention' dataset published on UCI Machine learning repository **[1]** by 'C. Okan Sakar' & 'Yomi Kastro '**[2]**. The dataset consists of feature vectors containing information of 12,330 sessions of online shopping website. To avoid any bias for repeating users or specific campaign over any period of time, authors have recorded instances for unique users over a period of 1 year. Dataset consists of total 18 features, out of which 10 are numeric and 8 are categorical attributes.

3.1 Target Feature:

The Dataset has a binary target feature 'Revenue' with two distinct levels True & False.

- 'True' value for this attribute indicates that particular user generated revenue for the company during that session by shopping online.
- 'False' value indicates that no revenue was generated during that particular session.

3.2 Descriptive Features:

The dataset has 10 numeric and 7 categorical descriptive features.

Numeric features represents information related to category of page visited, total time spent on pages of those categories and values derived from the URL information of actions taken on visited pages in real time. Bounce Rate, Exit Rate and Page Value are metrics measured by "Google Analytics" for each page of e-commerce website.

- **administrative:** Continuous - Pages - Number of Administrative pages visited for that session
- **administrative_duration**: Continuous - Seconds - Total time spent on Administrative pages for that session
- **informational:** Continuous - Pages - Number of Informational pages visited for that session

- **informational _duration:** Continuous - Seconds - Total time spent on Informational pages for that session
- **product_related:** Continuous - Pages - Number of Product related pages visited for that session
- **product_related_duration**: Continuous - Seconds - Total time spent on Product related pages for that session
- **bounce_rate:** Continuous - Percentage of users entered the site from that page and left without triggering any requests to the analytics server during a session
- **exit_rate:** Continuous - number of exits on a page / total number of views of the page
- **page_value:** Continuous - Average value for a webpage that a user visited before completing an e-commerce transaction
- **special_day:** Continuous - Closeness of the time of visiting a website to any specific special day

Categorical variables provides general information regarding each session.

- **operating_system**: Category [1 to 8] - Type of OS used by a user
- **browser:** Category [1 to 12] - Type of browser used by a user
- **region:** Category [1 to 9] - Location of a user in terms of region
- **traffic_type:** Category [1 to 20] - Network traffic type
- **month:** Category – Feb to Dec
- **visitor_type:** Category - 'New_Visitor' -> First time user, 'Returning_Visitor' -> User has visited website previously , 'Other' - No determined information regarding type of user
- **weekend:** Binary - Day of visiting website - 0 -> is a weekend, 1 -> is not a weekend

4. Data Preprocessing

1. **Preliminaries -** Data set is available at UCI Machine Learning Library [3] at - https://archive.ics.uci.edu/ml/machine-learning-databases/00468/online_shoppers_intention.csv in csv format. Data is loaded and saved in a data-frame '**raw_data**' and its columns are labeled with associated feature names. To confirm that data is loaded properly, we inspected dimensions and first few records of the data. Data has 12330 Instances and 18 features.

```
Confirming proper data loading
Loaded Data has 12330 instances & 18 features
```

Out[2]:

	administrative	administrative_duration	informational	informational_duration	product_related	product_related_duration	bounce_rates	exit_rates	page_values
0	0	0.0	0	0.0	1	0.000000	0.20	0.20	0.0
1	0	0.0	0	0.0	2	64.000000	0.00	0.10	0.0
2	0	0.0	0	0.0	1	0.000000	0.20	0.20	0.0
3	0	0.0	0	0.0	2	2.666667	0.05	0.14	0.0
4	0	0.0	0	0.0	10	627.500000	0.02	0.05	0.0

2. **Data Cleaning -** As a part of data cleaning and transformation process, we performed few checks and operations on our raw_data to make further processing easy and faster.

```
Data Types are:
                          Original type >>> Converted Type
administrative                 int64              int64
administrative_duration      float64            float64
informational                  int64              int64
informational_duration       float64            float64
product_related                int64              int64
product_related_duration     float64            float64
bounce_rates                 float64            float64
exit_rates                   float64            float64
page_values                  float64            float64
special_day                  float64            float64
month                         object           category
operating_systems              int64           category
browser                        int64           category
region                         int64           category
traffic_type                   int64           category
visitor_type                  object           category
weekend                         bool           category
revenue                         bool           category
```

   a. **Data type check and casting -** Our data has Numeric and Categorical features. All the numeric data is already in int64 or float64 type, no casting is required. Whereas all the categorical features are type casted to data type 'category' to efficiently perform operations during further steps of process.

b. **Null Value, Typo & Sanity Checks** - The data set we have used in this analysis is already cleaned by authors before publishing on source website. However, we confrimed this and found no null values. We also used summary tables to see if there are impossible values as part of sanity checks for abnormal values. Tables in following images represent the summary statistics as well as the sanity checks. Categorical features have expected numbers of unique levels, hence we concluded that there are no impossible values or typos in the categorical features.

1: Summary of continuous features

Table 1.1: Summary of features indicating duration of visit

	administrative_duration	informational_duration	product_related_duration
count	12330.000000	12330.000000	12330.000000
mean	80.818611	34.472398	1194.746220
std	176.779107	140.749294	1913.669288
min	0.000000	0.000000	0.000000
25%	0.000000	0.000000	184.137500
50%	7.500000	0.000000	598.936905
75%	93.256250	0.000000	1464.157213
max	3398.750000	2549.375000	63973.522230

Table 1.2: Summary of features indicating page count

	administrative	informational	product_related
count	12330.000000	12330.000000	12330.000000
mean	2.315166	0.503569	31.731468
std	3.321784	1.270156	44.475503
min	0.000000	0.000000	0.000000
25%	0.000000	0.000000	7.000000
50%	1.000000	0.000000	18.000000
75%	4.000000	0.000000	38.000000
max	27.000000	24.000000	705.000000

Table 1.3: Summary of feature indicating paging attributes

	bounce_rates	exit_rates	page_values
count	12330.000000	12330.000000	12330.000000
mean	0.022191	0.043073	5.889258
std	0.048488	0.048597	18.568437
min	0.000000	0.000000	0.000000
25%	0.000000	0.014286	0.000000
50%	0.003112	0.025156	0.000000
75%	0.016813	0.050000	0.000000
max	0.200000	0.200000	361.763742

Table 2: Summary of categorical features features

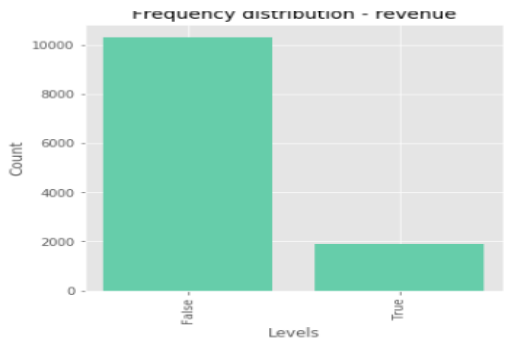
	month	operating_systems	browser	region	traffic_type	visitor_type	weekend	revenue
count	12330	12330	12330	12330	12330	12330	12330	12330
unique	10	8	13	9	20	3	2	2
top	May	2	2	1	2	Returning_Visitor	False	False
freq	3364	6601	7961	4780	3913	10551	9462	10422

3. **Handling outliers** - We have checked for outliers by scanning numerical features indivually and have used mean & standard deviation identify the outliers. We removed the instances whose values fell more than 3 standard daviation away from mean of that feature. We found that there are 128 outliers present in 6 features : 'product_related', 'informational' and 'administrative' - durations & 'bounce_rate', 'exit_rate' and 'page_value'. We dropped instances with outliers presence from the dataset.

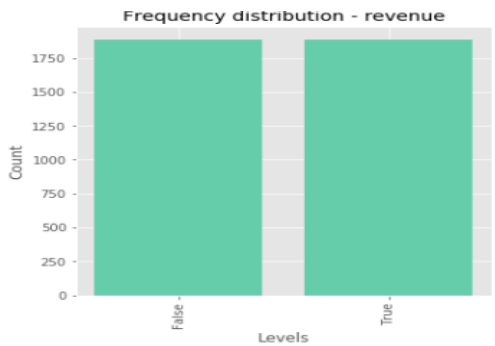
4. **Resampling -** The data set is highly imbalanced with target feature levels False:True in proportion roughly equal to 85 : 15.

- Most of the machine learning algorithms perform very poorly when they are trained on imabalanced datasets, hence we chose to resample the dataset using undersampling.
- RandomUnderSampler() method from imblearn.under_sampling package is used which under samples the majority class by randomly picking values. [imbdocs]

Before Undersampling



After Undersampling



After completing the preprocessing of data, we will start exploring the data using visualizations. However, for most of the visualizations, we will use original data and not the resampled data to understand true distributions and relationships among the features.

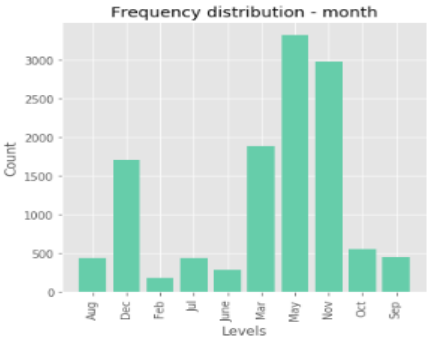
5. Data Exploration

We will use univariate and multivariate visualizations to explore some interesting aspects of the features in our dataset.

5.1 Univariate Visualizations

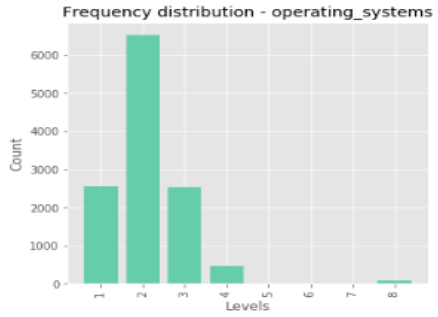
a. '**Month'** – Categorical feature Month has 10 levels in the dataset, each of which represents a certain month.
Mode of the feature is 'Jul' (July).

- That means there are more recorded sessions in July than any other levels. While March, October & December have comparatively low number of recorded sessions.



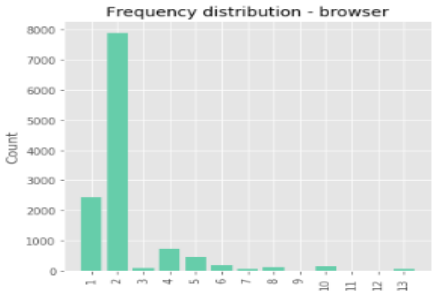
b. '**Operating_system**' – From examining the bar plot of this categorical feature, we could conclude that there are certain operating systems, which are more popular than others. OS labeled '1', '2' & '3' has more than 80 % instances among them.

- It will be interesting to see if a certain type of OS users are more likely to generate revenue or not.
- Dominating Oss can MS Windows, Mac or Linux, however No labels are provided.



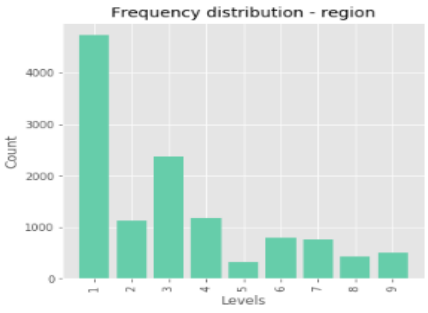
c. '**Browser**' – Just like Operating System, browser type 1 & 2, dominate the feature's values. Distribution among levels is highly uneven.

- These can be most popular browsers like Chrome or Safari, However, the textual labels are not provided.
- Mode of the feature is category '1'. While levels '11' & '12' have significantly low Instance Count.



d. '**Region**' – from the barplot, it is observable that certain regions have more users than other regions. With region '1' being the mode.

- From the graph, it can be argued that in certain regions online shopping on this particular website is not as popular other regions. This insight can be used by marketing team to focus on regions with lower count. But the information is, at best, abstract and no concrete decision can be made on this information alone.

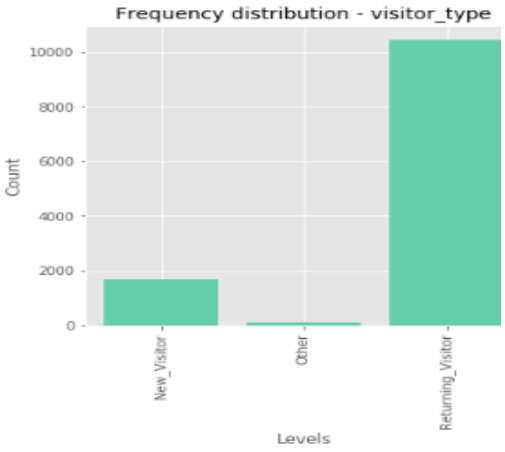


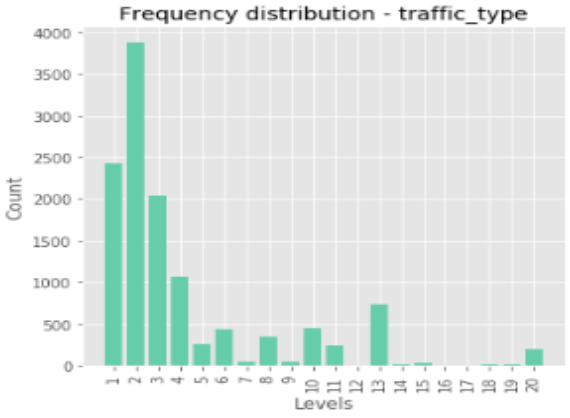
e. '**visitor_type**' – From the barplot beside, it can be infered that most of the sessions are from returning visitors, and very few are new visitors.

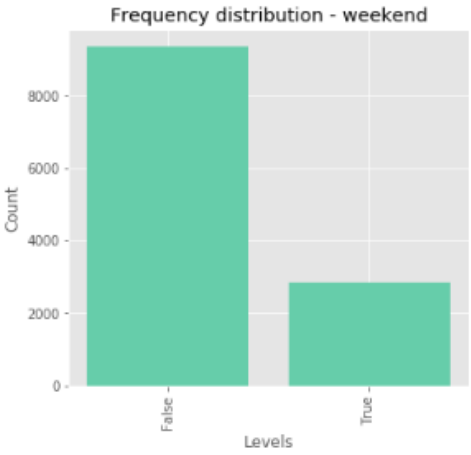
- It will be interesting to see which type of visitors are more likely to generate revenue.

f. '**traffic_type**' – Most dominant traffic type is '2', however, apart from the right skewness of the graph, there isn't much information.

g. '**weekend**' – feature represents if observed session was recorded on weekend or not. The distribution is approcimately in 3 : 1 ration in favor of level 'False', Meaning that the weekend traffic is 1/3rd of the rest of the traffic.

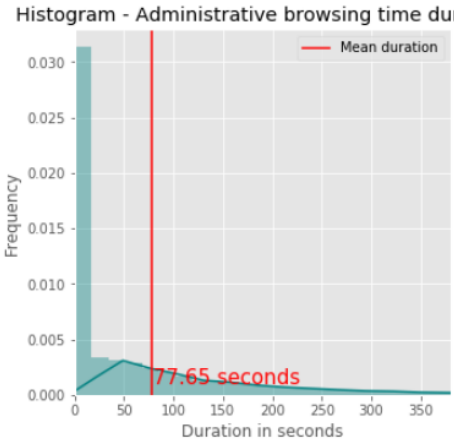


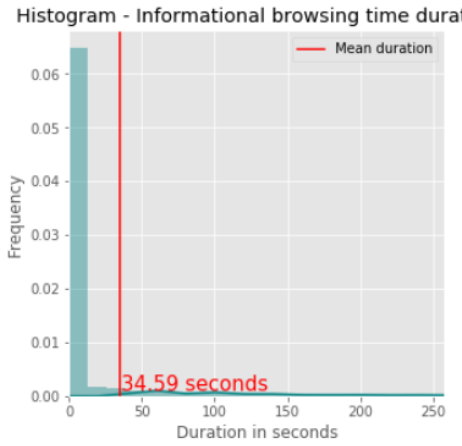




h. '**Adminstrative_duration**', '**Informational_duration**' & '**Product_related_duration**'

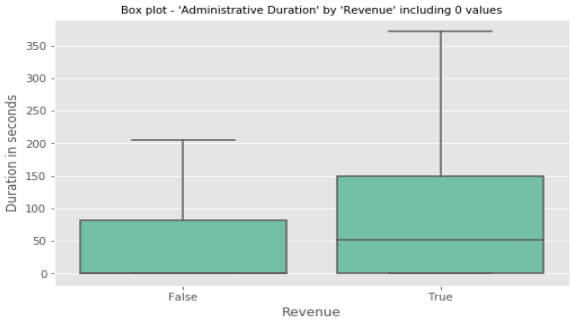
- Next, we observe 3 histograms of features indicating browsing time duration for various type of pages in a session. Types of pages are 'Administrative', 'Informational' & 'Product_related'.
- It is understandable that many of the values for these features will be zeros. Because, in a single session, if user performs some product search, s/he might not perform administrative tasks. Hence all 3 histograms have significant frequency on 0.
- However, from the means of the features plotted on the graphs, it is clear that highest browsing time is for product related browsing, which has a mean duration of 1192.52 seconds, while administrative browsing mean is 77.65 seconds and Informational browsing duration is 34.59 second, which is lowest of all 3 features.
- The significant difference in means of 3 durations means that if we calculate distance between two instances using minkowski distance, product_related_duration will dominate the distance, especially for higher values of p, e.g. 2,3…etc.
- This prompts us to perform data scaling before feeding data to classifiers.

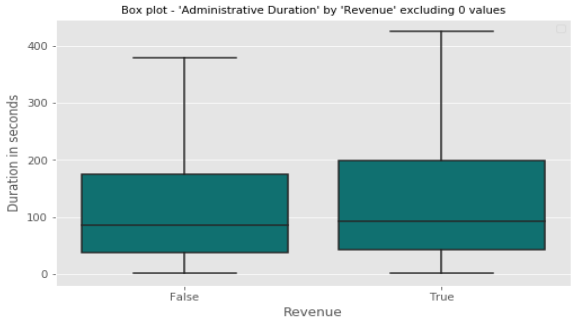




5.2 Multivariate Visualizations

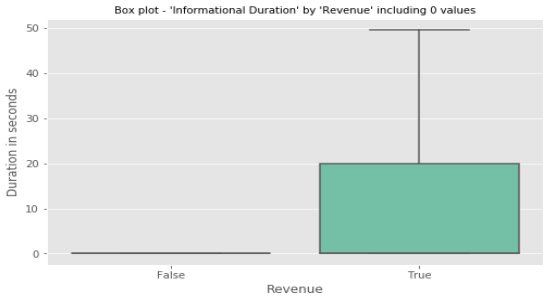
1. **Hypothesis - People who spend more time doing administrative tasks generate more revenue.**

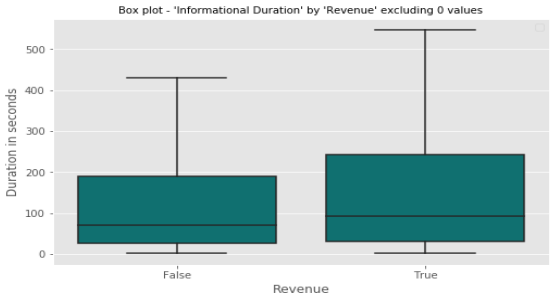




- Boxplots shows how the values of 'Administrative_duration' is distributed across levels of target feature 'Revenue'. We will consider the plot which exculdes 0s (right side) to get a clearer picture, as lots of 0 readings will skew the mean.
- As per the right graph, there is not a major difference between the ranges and means of box plots. Hence we don't have a strong evidance to state that our hypothesis is plausible.

2. **Hypothesis - People who spend more time doing Informational browsing generate more revenue.**

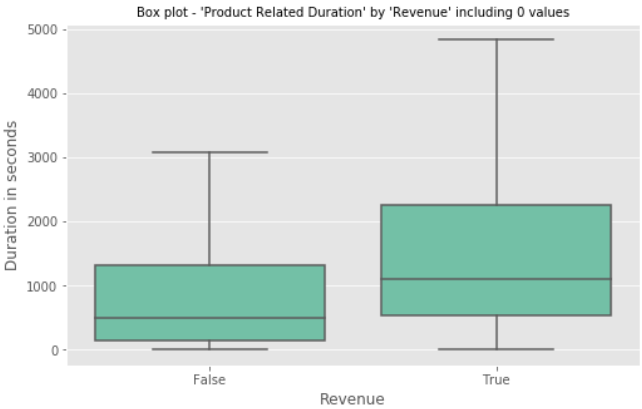


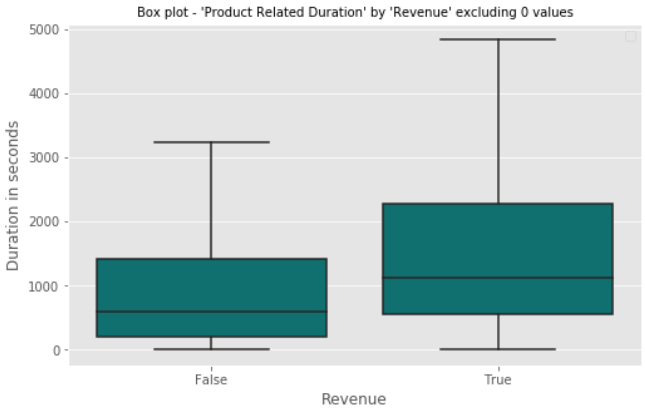


- Boxplots shows how the values of 'Informational_duration' is distributed across levels of target feature 'Revenue'. Left-box plot indicates that majority of the instance whose Revenue is false has Informational_duration equal to 0, so that it pulls entire plot to 0, with 0 mean and near 0 standard deviation. Removing the observations with 0 values, reveal the true picture.
- Here again, there is a difference between mean and ranges of two boxplots in right side plot, however, it doesn't seem significant to make our hypothesis plausible.

3. **Hypothesis - People who spend more time doing Product related browsing generate more revenue.**

- In this set of boxplots, there is a major difference between the boxes in the right side. Revenue = True box has significantly higher mean and upper IQR.

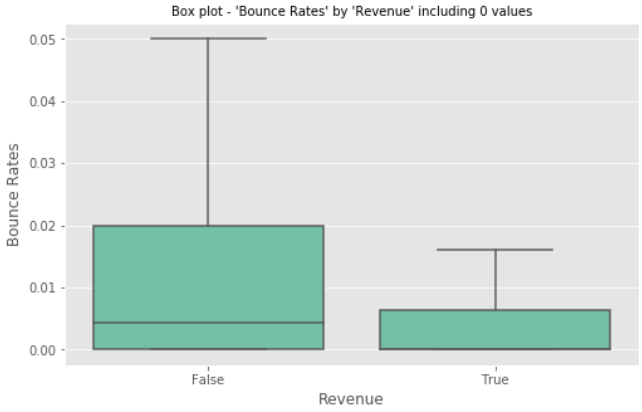


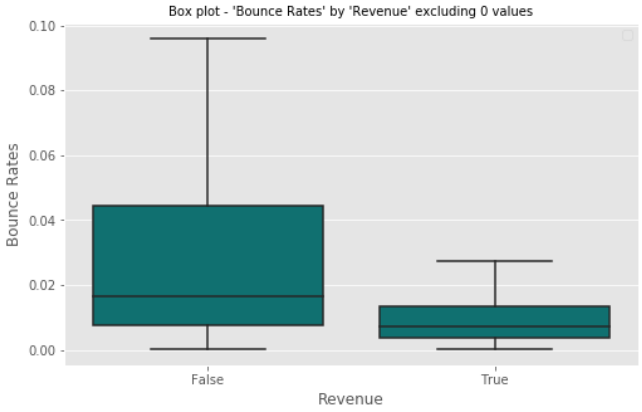


- Hence, we can term our hypothesis plausible and can be investigated further if needed.

4. **Hypothesis – Higher bounce rate is less likely to generate revenue**

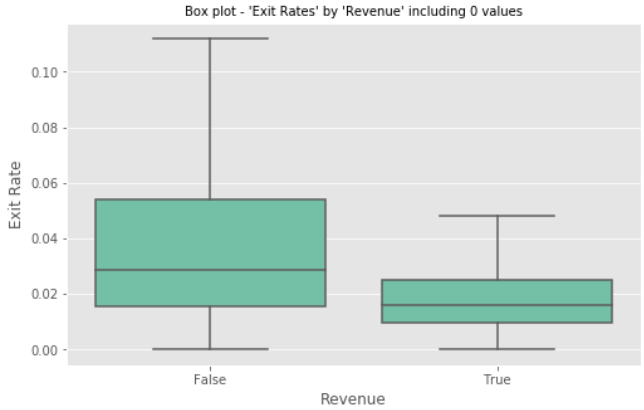
- From the right side box plot, we can see that sessions which generated revenues have significantly lower bounce rate. Which makes our hypothesis plausible.

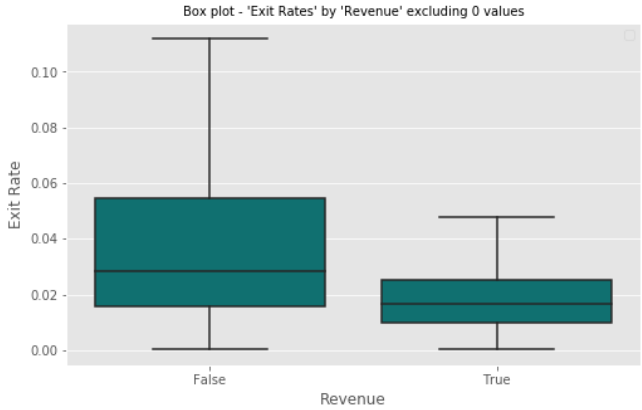




5. **Hypothesis – Lower exit rate is more likely to generate revenue**

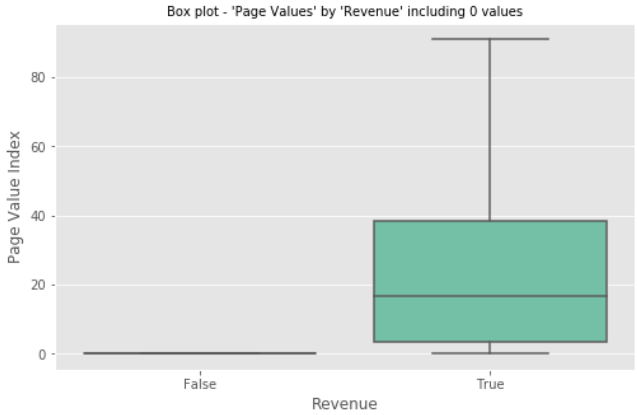
- Again, from the right side box plot, revenue = True boxplot has significantly lower mean and other statistics. Which indeed strengthens our hypothesis saying that lower exit rates are more likely to generate revenue.

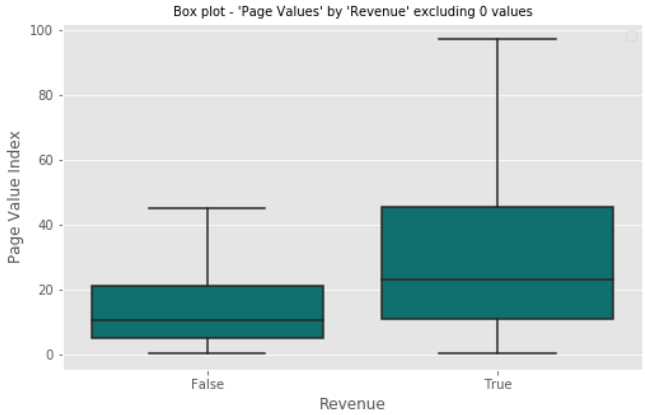




6. **Hypothesis – Higher page value for session has positive effect of possibility of generating revenue**

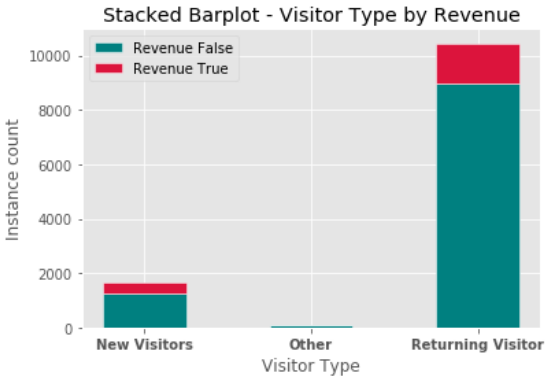
- It is evident from the left side plot that majority of instances with target level 'False' has page_value equal to 0. Which pulls down mean and standard deviation to nearly 0. Hence, focusing on rest of the instances can give a better understanding of the relationship among these features.

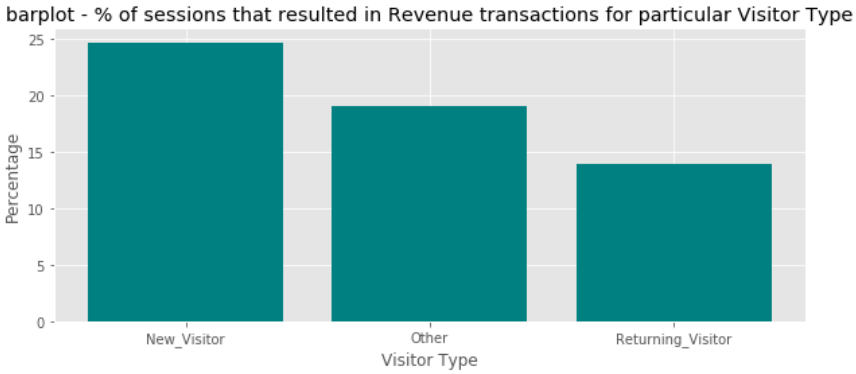




- From the right plot, we can observe that instances with target level 'True' has higher mean and IQR than features with target level 'False'. Which supports our hypothesis and in turn makes it plausible.

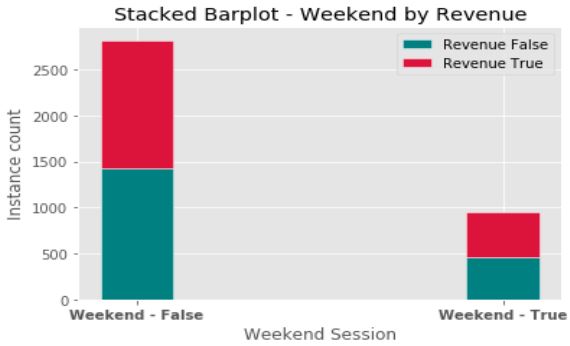
7. **Hypothesis – Returning customers are more likely to generate revenue than new customers**

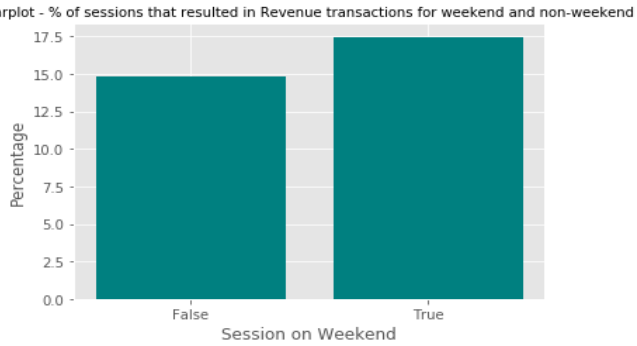




- Due to imabalnce in distribution of instances, we can not directly compare the instances that generated revenue in each visitor type level. So we compare % of instances that generated revenue from total instances in that level.
- Doing so gives a surprising result, in which 'New Visitor' are more likely to generate revenue than 'Returning Visitor'. (New Visitor ~ 24% & Returning Visitor ~ 14%). Which contradicts our hypothesis.

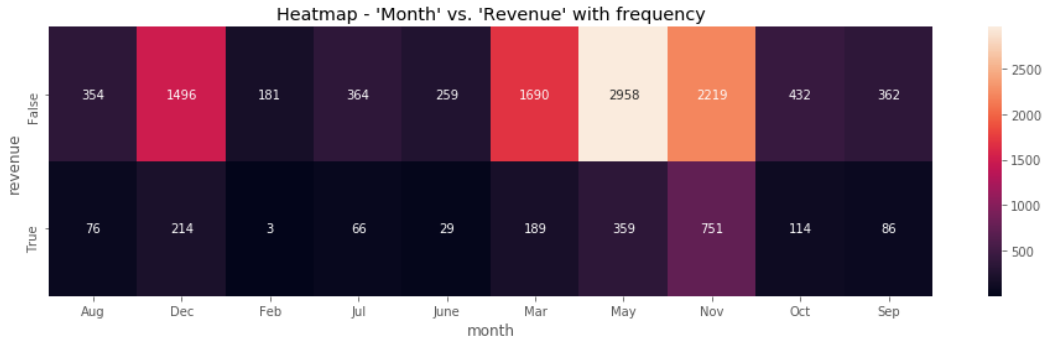
8. **Hypothesis – Weekend session instances are more likely to generate revenue than new customers**



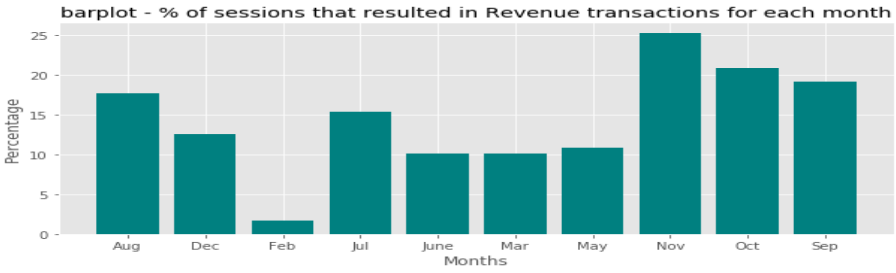


- Going by the same logic as hypothesis 7, sessions on weekend generates 17.5% of times, while 15% of session on non weekend days resulted in revenue. Here difference is not that big, but it does give our hypothesis some level of confidence.

9. **Hypothesis – Customers shop more during some months**



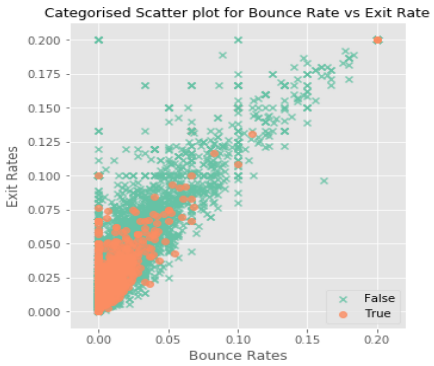
- From the heatmap on the left hand side, it is evident that months like October, November & May has significantly higher sessions recorded.



- From the bar plot, it is evident that November also has highest % of revenue generating Sessions with around 25%. While only 2 % of the sessions in February. It makes our hypothesis plausible.

10. Hypothesis- Higher bounce rate results in higher exit rates

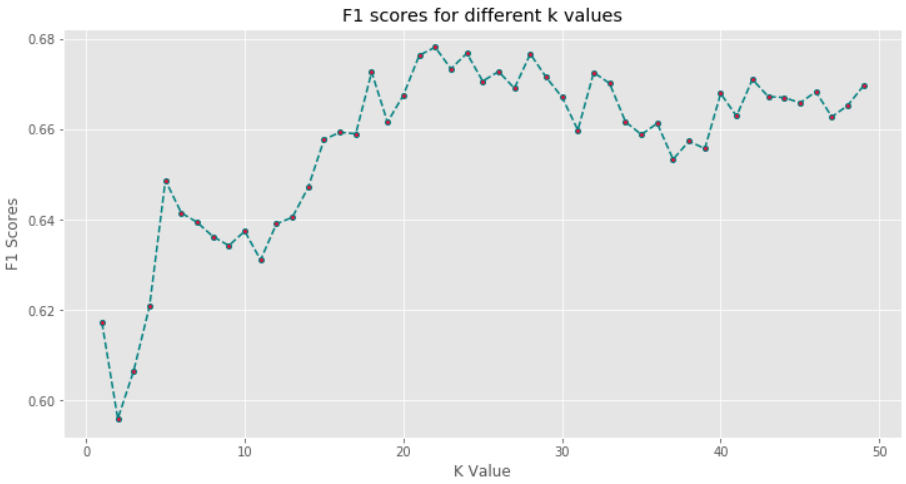
- To confirm the hypothesis, we will check the correlation among the features.
- Scatterplot shows that as the exit rate does strongly increase with increasing bounce rate.
- A calculation revealed the coefficient of correlation to be around 0.91. Which suggests a strong linear correlation. Which concludes our hypothesis to be plausible.



6. Data Modelling

6.1 Model training and performance comparision

- After exploring the relationships among the features, we proceed to create classification models to predict label for target feature 'Revenue' based on descriptive features.
- We start by encoding the data. We encode the non binary categorical features with one-hot encoding and encode the binary features with label encoding. This results in a all numeric dataset, which is a requirement for all ML Algorithms in python. One hot encoding increases feature count to 72 from original 18, as it generates binary feature for each level of a particular feature.
- We will be using Area under the ROC Curve metric, which is a common choice for a binary target feature. ROC curve - **Receiver Operating Characteristic curve**, is is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters:
  A. True Positive Rate
  B. False Positive Rate



- From summary statistics it can be observed that numeric features has skewed data. To supress the impact of extremes of data on the performance of models, we will perform scaling. We will compare the result of MinMax scaling, Standard scaling and Robust scaling for each model we apply.
- To establish a baseline performance, we have chosen to use KNN algorithm, with the k value being 22. To choose the K value, we played around with different K values and compared f Scores for all the value of k in range

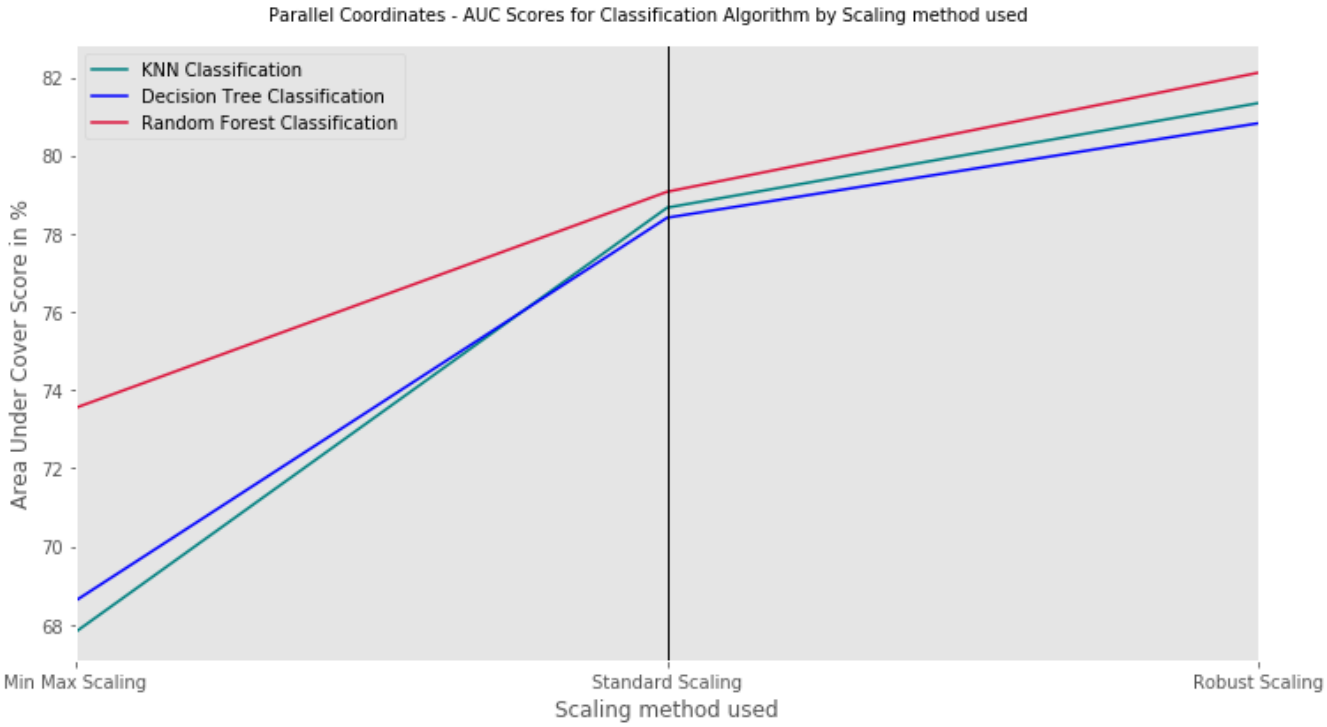
(1,50). Following graph shows the results of the process. Hence we will be using K = 22 for rest of the tasks.

6.2 Model training and performance comparision

- We have used 2 variants of machine learning algorithms,
  1. Similarity based learning algos - KNN classifier
  2. Information based learning algos – Decision Tree classifier & Random Forest Classifier

- Random Forest Algorithm is essentially decision tree algorithm ran multiple times over a random sub sample of data in a model ensamble.
- We will train all 3 algorithms on 3 sets of data, which generated by scaling the data using
  A. MinMax Scaling
  B. Standard Scaling
  C. Robust Scaling

- All 3 variants of datasets were splitted in train test pairs, with test size set at 20%. As mentioned earlier, we used Area under ROC curve to compare the performance of all 3 algorithms and all 3 scaling techniques.
- It is important to note that AUC score for random forest can vary for each iteration as it generates multiple random samples using sub sampling, and depending on the sub samples created, prediction can change.
- The following graph and table summarises the AUC-ROC measurements for our run.



	Min Max Scaling	Standard Scaling	Robust Scaling
KNN Classification	0.678106	0.780163	0.829382
Decision Tree Classification	0.686103	0.781475	0.821312
Random Forest Classification	0.735445	0.789374	0.831933

6.3 Performance Comparisions

- AUC-ROC scores for KNN, Decision Tree & Random Forest Classifiers are 0.829, 0.821, 0.831 respectively.
- It is evident that Min Max Scaling performs rather poorly, as all 3 algorithms have lowest score of when trained on MinMax Scaled data.
- Models trained on Robust Scaled Data consistently outperformed models trained on Min Max & Scaled data.
- After running multiple iterations, we observed that most often, Random forest models will outperform other two models. While decision tree and KNN classifiers have mostly similar scores.
- That prompted us to use Robust Scaling for our final modeling exercise.
- We used multiple train test splits with ratios (50:50), (60:40) & (80:20) and observed the following results.

Table – F1 Scores for Train Test Split Sizes & 3 classifiers

	KNN Classifier	Decision Tree	Random Forrest
50 : 50	0.75	0.79	0.83
60 : 40	0.74	0.79	0.80
80 : 20	0.73	0.80	0.83

- It can be argued that, due to highly skewed distribution of features, as observed in data exploration section, Robust Scaler performs better than other two scalers. Reason for that can potentially be immunity of robust scaling from skewness in the data.
- The difference in performance all 3 classifiers is very subtle when trained on Min Max Scaled Data, however, even then, Random Forest Classifier significantly out performs other classifiers. However, when standard or robust scaling is used, the difference in performance is not that significant.
- From our project we can also conclude that a model ensamble has higher chance of performing well if exact nature of data is not know. Also we can recommend to use Robust Scaling if data is highly skewed.

6.4 Limitations & Future work

- We have not performed feature selection as the number of features were comparatively low. However, In future iterations of project, we can perform feature selection to understand how model performance gets affected. There is a scope of advance hyper parameter tuning to increase model performance, but due to limitation of the project report, we have not included it in here, but it will be interesting to know how tuned parameters can increase scores of models.

7. Bibiliography

1. **Dataset Authors - C. Okan Sakar, Yomi Kastro, Accessed on - Viewed on (16/05/2019), link -** <https://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Intention+Dataset>
2. **Dataset hosted on – UCI Machine Learning Library -** https://archive.ics.uci.edu/
3. **Sakar, C.O., Polat, S.O., Katircioglu, M. et al. Neural Comput & Applic (2018)**
4. **Dr.Yongli Ren - Practical Data Science: Data Classification, pdf, internal lecture notes for RMIT**
5. **Python Graph Gallery – Viewed on (12/05/2019) -** https://python-graph-gallery.com/
6. **Ye Wu & Rick Radewagen – Imabalanced Data Handling – Viewed on (15/05/2019) -** https://www.kdnuggets.com/2017/06/7-techniques-handle-imbalanced-data.html
7. **Jeff Hale – Feature Scaling in Python – Viewed on (14/05/2019) -** https://towardsdatascience.com/scale-standardize-or-normalize-with-scikit-learn-6ccc7d176a02
8. **Chris Moffitt - Encoding Categorical Values in Python – Viewed on (17/05/2019) -** https://pbpython.com/categorical-encoding.html
9. **Renuka Joshi – Performance Measurement Metrics – Viewed on (19/05/2019) -** https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/