This is my own work - _____

Real-time prediction of online shoppers' purchasing intention using K-Nearest Neighbor and, DecisionTreeClassifier

Amay Viswanathan Iyer
Royal Melbourne Institute of Technology

s3970066@student.rmit.edu.au

May 24, 2023

Table of Contents

Abstract:
The aim of this report was to explore how a K-Nearest-Neighbor (KNN) Supervised Machine-Learning model can be utilized to incrementally increase the accuracy in predicting whether an online shopping site visitor's visit will result in revenue generation. The dataset was harvested from the University of California, Irvine's Machine Learning Repository where it was previously used for a paper titled, "Real-time prediction of online shoppers' purchasing intention using multilayer perceptron and LSTM recurrent neural networks". (Sakar, 2018) This report concludes that when the features, 'PageValues', 'ProductRelated', 'ProductRelated_Duration', 'Month_Nov', 'Administrative', 'TrafficType_2', 'VisitorType_New_Visitor', 'Informational', 'Administrative_Duration', 'ExitRates', 'BounceRates', 'VisitorType_Returning_Visitor', and 'TrafficType_3' are engineered to fit a machine learning model in addition to Decision Tree, Grid Search, SMOTE, and Random Forest Classifier in that order, the prediction accuracy increases from 86% to 92%. It is recommended that any prospective American online retail stores aiming to generate more revenue from site visitors allocate resources to engage their visitors with good content, especially in the month of November because of special days like Black Friday and Cyber Monday. (Albright, 2011) It is also recommended that stores make their User Account Management related pages harder to navigate as users that are more likely to spend longer durations of time on Account-Management related pages are more likely to generate revenue for online stores.

Introduction:
The advent of the internet has revolutionized the retail industry, with online shopping becoming an integral part of consumers' lives. This trend has been further accelerated by the SARS-Cov-2 pandemic of 2020, which has led to an unprecedented surge in online shopping due to lockdowns and social distancing measures. As reported by Forbes, the global e-commerce market is projected to reach a staggering $6.3 trillion by 2023 (Baluch, 2023). This rapid growth presents a significant opportunity for retailers, but it also poses new challenges in understanding and predicting online shopping behavior. Understanding the dynamics of online shopping is crucial for retailers to optimize their strategies and maximize revenue. This involves analyzing various factors such as the type of web pages visited by customers, the duration of their visits, the time of the year, and other relevant variables. Previous research has shown that these factors can significantly influence online shopping behavior and the likelihood of a visit resulting in a purchase (Fader, 2004) (Montgomery, 2004). However, the complexity and high-dimensionality of online shopping data make it challenging to analyze using traditional statistical methods. This is where a KNN Supervised Machine-Learning model can aid in predicting whether a visitor's visit to an online shopping site will result in revenue generation. The predictive power of these features will be evaluated, and the model's accuracy will be incrementally improved by engineering additional features and applying other machine learning techniques. By providing insights into the factors that influence online shopping behavior and revenue generation, this report aims to contribute to the growing body of research in this area and provide practical recommendations for online retailers.

Methodology:
As mentioned before, this research paper utilized a dataset, "collected from an online bookstore built on an osCommerce platform" (Introduction, Sakar, 2018). The dataset was then loaded onto a dataframe in a jupyter notebook where it was Pre-Processed. Upon initial pre-processing, it was inferred that the dataset consisted of ten numerical features and eight categorical features. Of these eight features, Revenue is the target variable. This is consistent with Table 1 and Table 2 of the main research paper referred for this dataset. (Sakar, 2018). Some of the metrics in Table 1 are features measured by Google Analytics (Clifton, 2010).

Task 1: Retrieving and Preprocessing the data

```
In [41]:
# Checking the data types of all columns
print(df.dtypes)
print(df.isnull().sum())
```

```
Administrative             int64
Administrative_Duration    float64
Informational              int64
Informational_Duration     float64
ProductRelated             int64
ProductRelated_Duration    float64
BounceRates                float64
ExitRates                  float64
PageValues                 float64
SpecialDay                 float64
Month                      object
OperatingSystems           int64
Browser                    int64
Region                     int64
TrafficType                int64
VisitorType                object
Weekend                    bool
Revenue                    bool
dtype: object
Administrative             0
Administrative_Duration    0
Informational              0
Informational_Duration     0
ProductRelated             0
ProductRelated_Duration    0
BounceRates                0
ExitRates                  0
PageValues                 0
SpecialDay                 0
Month                      0
OperatingSystems           0
Browser                    0
Region                     0
TrafficType                0
VisitorType                0
Weekend                    0
Revenue                    0
dtype: int64
```

After the dataset was loaded into the jupyter notebook, it was meticulously checked for missing values. This is because in order to design a viable model for predicting revenue, a holistic dataset with no missing values is essential. I incorporated standard missing values checking procedures as outlined on the left.

Following this, I checked the data for any possible duplicates. I decided against dropping the duplicate values as in the original paper (Sakar, 2018) it is stated that every single instance is a unique instance of a user interacting with an online bookstore. There are a total of 12330 distinct sessions of users interacting with the online store of which, 10422 sessions resulted in no revenue and only about 1908 sessions resulted in revenue. Given such a large dataset, it is fair to assume that there can be approximately 125 instances of identical kinds of instances. Therefore, even though there are 125 duplicate rows, I decided against dropping them as these instances might be coincidentally identical to other instances.

```
In [42]:
duplicate_rows = df.duplicated()
print("Number of duplicate rows: ", duplicate_rows.sum())
```

```
Number of duplicate rows:  125
```

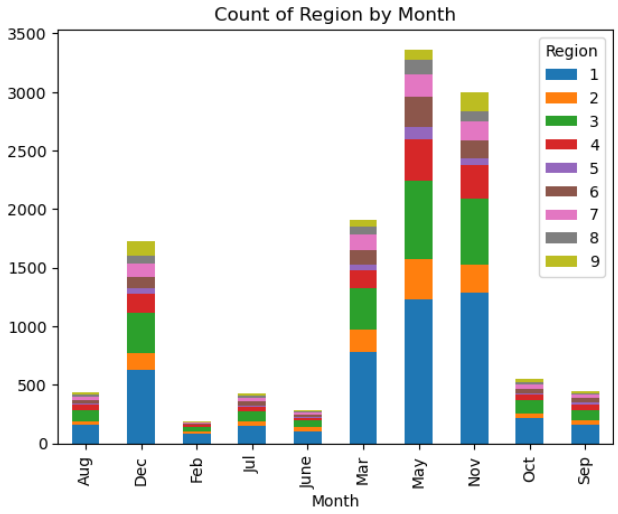
With regards to the SpecialDay categorical variable, I decided against removing all instances which weren't a 0 or a 1 as I wanted to be able to evaluate whether the proximity to a special day has an impact on the predictability of revenue. When I took a glance at the descriptive statistics, I found that the number of instances for each of the sessions mirrored the dataset used in (Sakar, 2018). Therefore I decided to not add or remove any instances to omit the possibility of an imperfect model fit.

Task 2.1: Data Exploration

The initial analysis that I employed on the dataset was the Descriptive Statistics. This would give me an understanding underlying the density of data across all the numerical features in the dataset. With descriptive statistics, we would be able to understand the user-engagement metrics of the site visitor and how they interact with the online store.

Followed by the Descriptive Statistics, I conducted a quantitative analysis across all the numerical variables to gain an understanding of the spread of data. In order to be able to understand the spread of data while also visualizing the outliers, I decided to curate Histograms, Density Plots, Box Plots, and Violin Plots for all the numerical features which include: 'Administrative', 'Administrative_Duration', 'Informational', 'Informational_Duration', 'ProductRelated', 'ProductRelated_Duration', 'BounceRates', 'ExitRates', 'PageValues', and 'SpecialDay'. This was to judge whether I should handle the outliers and reorganize the data or leave it as is. I decided to leave the data as is to ensure that the KNN model that I will employ later is the best model-based representation of the data as is possible. I will explain what I inferred from the Data Exploration in the Results and Discussion section.

The categorical variables include: 'Month', 'OperatingSystems', 'Browser', 'Region', 'TrafficType', 'VisitorType', 'Weekend', and 'Revenue'. In order to visually understand the relationship across all of these variables, I employed stacked plots. Stacked plots are a colorful way to understand multivariate relationships in a dataset. For example, in the stacked plot below:



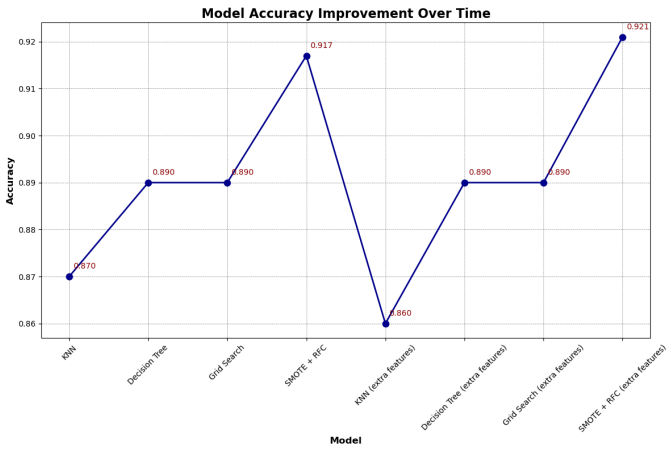
It is visually intuitive to grasp that most of the visits to the online store occurred in May and November. In addition to that, because of the choice of a stacked plot, we can infer that a larger number of visitors from region 1 visited the site in November than they did in May. This is an example of nuance that I felt geared me towards choosing stacked plots as a way to compare relationships between categorical variables.

After my categorical and quantitative analyses, I chose an intimidating approach to visualize correlation between variables. I split every categorical variable into their individual categories and plotted every single one against all other variables in a correlation heatmap. Splitting all the variables into their individual parts resulted in a total of 76 variables. Granted that the heatmap is the most intimidating to understand, I chose to approach the correlation analysis in this manner as I wanted to choose features that would be able to help in augmenting the predictive model individually.

Task 3: Data Modeling:

After obtaining the correlation coefficients across 76 individual variables, I plotted the top twenty variables most correlated with Revenue. I quantified the correlation coefficients in order to make feature selection and engineering simpler. After plotting the correlation coefficients most correlated with Revenue, I chose the following 10 variables 'PageValues', 'ProductRelated', 'ProductRelated_Duration', 'Month_Nov', 'Administrative', 'TrafficType_2', 'ExitRates', 'BounceRates', 'VisitorType_Returning_Visitor', and 'TrafficType_3' for parameter tuning of the KNN model.

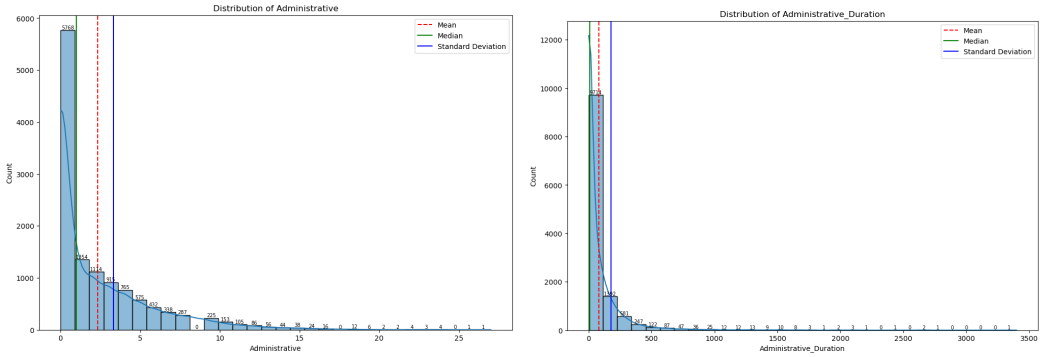
Following the KNN model tuning, I proceeded with DecisionTree, GridSearch, and Synthetic Minority Over-sampling Technique + Random Forest tuning in that order. With these consecutive training steps, I saw an incremental improvement in the model's accuracy. Following this improvement in accuracy as a result of more robust tuning, I decided to increase the number of features from the initial set of 10 to an increased set of 13: 'PageValues', 'ProductRelated', 'ProductRelated_Duration', 'Month_Nov', 'Administrative', 'TrafficType_2', 'VisitorType_New_Visitor', 'Informational', 'Administrative_Duration', 'ExitRates', 'BounceRates', 'VisitorType_Returning_Visitor', and 'TrafficType_3'. After encoding the features, I followed the same chronological steps as before and although initially the KNN accuracy dropped, the end resulting model accuracy with SMOTE and Random Forest was better than the one with only 10 features. Below is a graph that depicts the increasing accuracy as I incrementally followed the steps:



After the greatly improved accuracy, I designed the decision tree to show the likely pathway for a visitor's visit on the online site which might result in a transaction. The decision tree was made by the encoded variables and was used to classify if a decision pathway will result in Revenue or not.

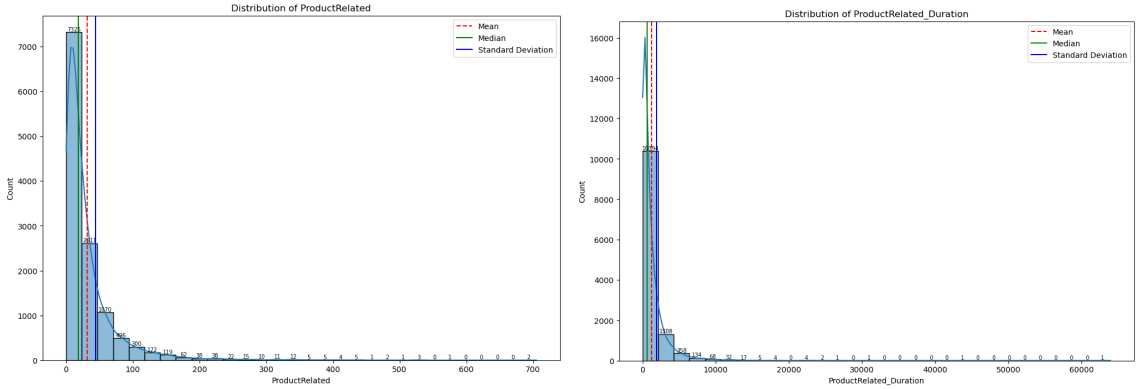
Results:

The following two Histograms depict the dispersion of the Administrative and Administrative_Duration variables.



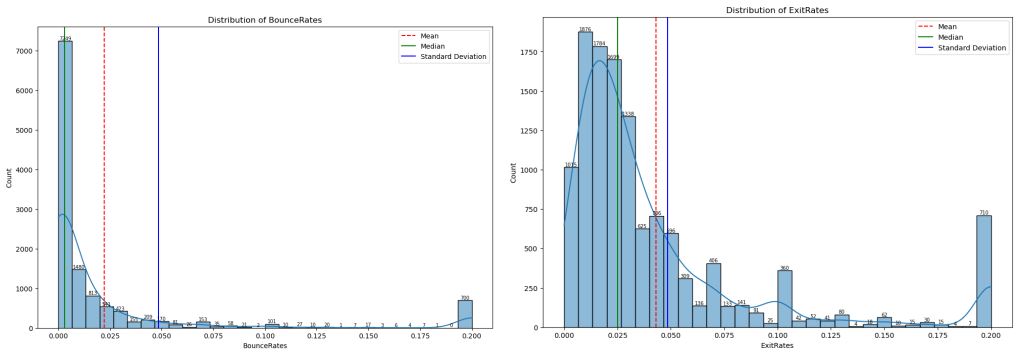
Administrative: On average, users visited about 2.3 administrative pages per session, with a standard deviation of 3.3. The median is 1, indicating that half of the sessions had at least one administrative page visit. The maximum number of administrative pages visited in a session was 27. Administrative_Duration: The average time spent on administrative pages was about 80.8 seconds, but with a high standard deviation of 176.8 seconds, indicating a wide spread in the data. The median is 7.5 seconds, suggesting that half of the sessions had an administrative duration of at least 7.5 seconds. The maximum time spent on administrative pages in a session was 3398.75 seconds.

The following two Histograms depict the dispersion of the ProductRelated and ProductRelated_Duration variables.

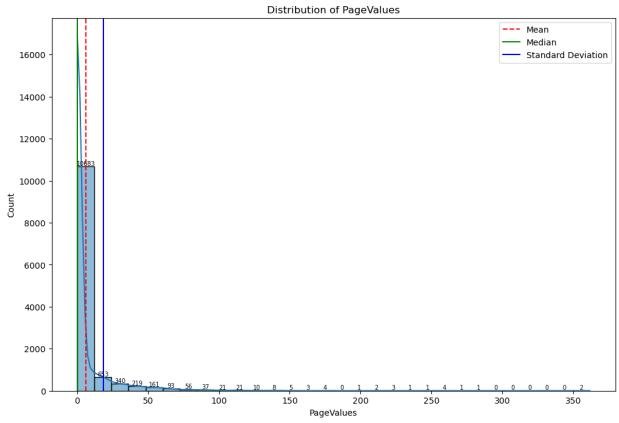


ProductRelated: On average, users visited about 31.7 product-related pages per session, with a standard deviation of 44.5. The median is 18, indicating that half of the sessions had at least 18 product-related page visits. The maximum number of product-related pages visited in a session was 705. ProductRelated_Duration: The average time spent on product-related pages was about 1194.7 seconds, but with a high standard deviation of 1913.7 seconds, indicating a wide spread in the data. The median is 598.9 seconds, suggesting that half of the sessions had a product-related duration of at least 598.9 seconds. The maximum time spent on product-related pages in a session was 63973.522 seconds.

The following two Histograms depict the dispersion of the BounceRates and ExitRates

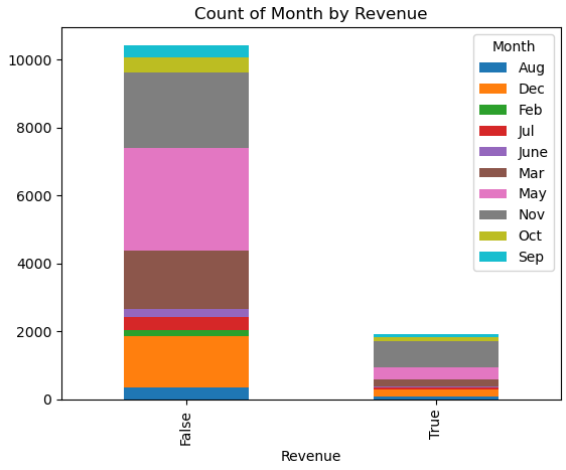


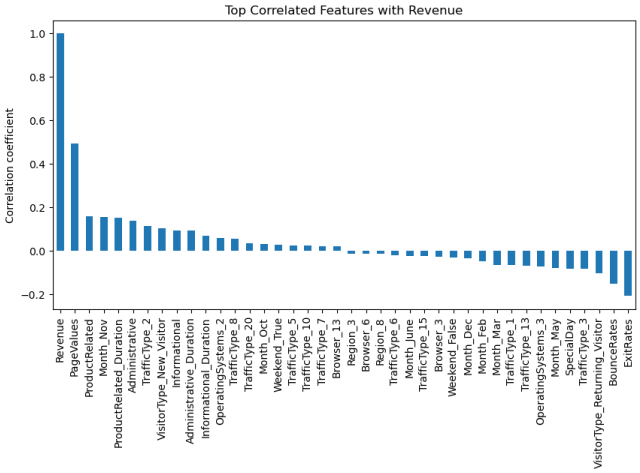
BounceRates: The average bounce rate was about 0.022, with a standard deviation of 0.048. The median is 0.003, indicating that half of the sessions had a bounce rate of at least 0.003. The maximum bounce rate in the dataset was 0.2. ExitRates: The average exit rate was about 0.043, with a standard deviation of 0.049. The median is 0.025, indicating that half of the sessions had an exit rate of at least 0.025. The maximum exit rate in the dataset was 0.2.



PageValues: The average page value was about 5.89, with a high standard deviation of 18.57, indicating a wide spread in the data. The median is 0, suggesting that at least half of the sessions had a page value of 0. The maximum page value in the dataset was 361.76.

On the right is a Stacked Bar Chart depicting the number of visits that ended in a transaction stratified by Month. As is evidently visible, even though on the whole there are much more visits to the site in May, there seem to be a larger proportion of visits in November that result in a transaction as opposed to visits in May that result in a transaction. This makes sense given that Black Friday and Cyber Monday occur in November adjacent to the Thanksgiving holidays.

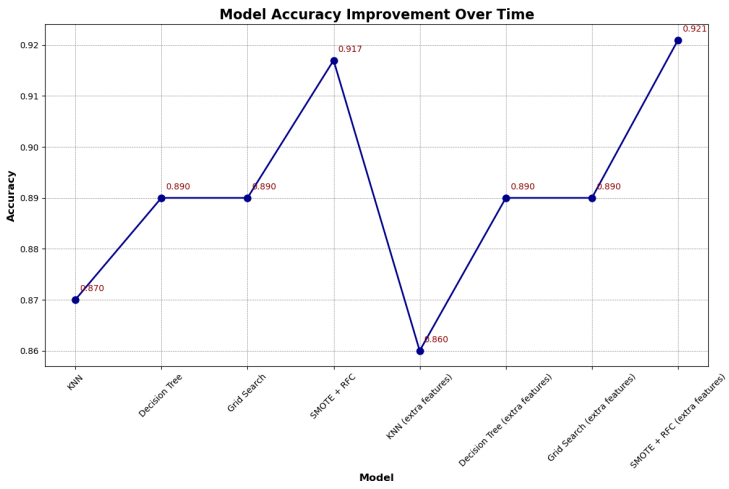




Given all these results, after splitting all the categorical variables to their individual parts for a better fit for the best correlative individual variables, on the left are the top twenty individually split variables that are most correlated with Revenue (positively or negatively).

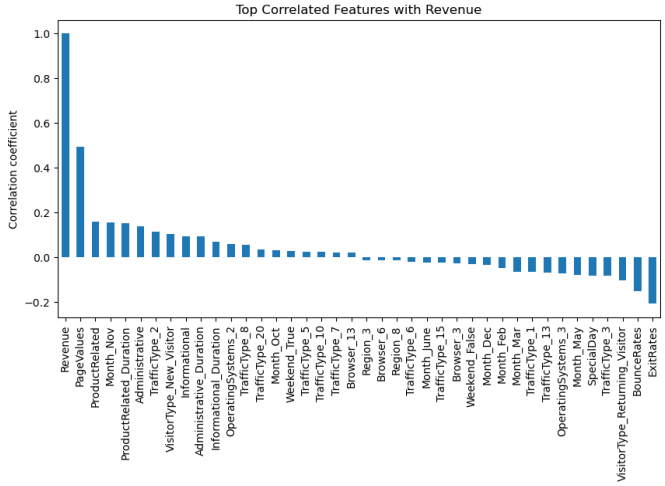
From these Correlation coefficients, I chose 13 features to encode into a Machine Learning model to be able to better predict revenue generation upon a hypothetical visit. I have already shown the incremental

improvement of the model's accuracy following every subsequent model in addition to the addition of three more features from 10. For a recap, below is the graph depicting the improvement in accuracy across every consecutively hypertuned model.



Discussion:

The dataset used in this report has a high likelihood of originating from the United States. The reason being that the highest proportion of revenue traffic is occurring in November and May which overlaps with the Thanksgiving Black Friday and Cyber Monday deals along with the start of Summer break respectively. Another reason being the Nine geographical regions which is the same number of Geographical regions that the United States is divided into. (CASC, 2023).



1. PageValues: This feature represents the average value for a web page that a user visited before completing an e-commerce transaction. The positive correlation with 'Revenue' suggests that pages with higher values (likely indicating their importance or effectiveness in the customer journey) tend to lead to more transactions. Therefore, improving the quality or relevance of these high-value pages could potentially increase revenue.
2. ProductRelated, Month_Nov, ProductRelated_Duration, Administrative, and TrafficType_2: These features are positively correlated with 'Revenue', albeit less strongly than PageValues.
    a. ProductRelated and ProductRelated_Duration likely refer to the number of product-related pages visited and the time spent on these pages, respectively. Their positive correlation with 'Revenue' suggests that customers who engage more with product-related content are more likely to make a purchase.
    b. Administrative might refer to the number of administrative-type pages a user visited. A positive correlation could indicate that visits to these pages (perhaps user account pages or help pages) are part of the purchasing journey for many customers.
    c. Month_Nov and TrafficType_2 being positively correlated with 'Revenue' suggests that November might be a particularly strong month for sales (perhaps due to events like Black Friday or Cyber Monday), and whatever traffic type 2 represents tends to bring in customers who are more likely to make a purchase.

3. ExitRates: This feature likely represents the percentage of users who exit the website after viewing a page. The negative correlation with 'Revenue' suggests that pages with high exit rates may be driving potential

customers away, leading to lower revenue. Identifying and improving these pages could potentially increase revenue.

4. BounceRates, VisitorType_Returning_Visitor, TrafficType_3, SpecialDay, and Month_May: These features are negatively correlated with 'Revenue'.
    a. BounceRates likely refers to the percentage of visitors who navigate away from the site after viewing only one page. A high bounce rate could indicate that the landing page is not relevant or engaging to visitors, which could negatively impact revenue.
    b. VisitorType_Returning_Visitor being negatively correlated with 'Revenue' is a bit counterintuitive, as one might expect returning visitors to be more likely to make a purchase. This could warrant further investigation.
    c. TrafficType_3, SpecialDay, and Month_May being negatively correlated with 'Revenue' suggests that whatever traffic type 3 represents tends to bring in less profitable customers, special days (perhaps holidays or sales events) may not be as profitable as expected, and May might be a particularly weak month for sales.

KNeighborsClassifier explanation: I chose to use the KNeighborsClassifier as a starting off point because of the relationships I observed in my correlation coefficients. The features I selected to use in the model ('PageValues', 'ProductRelated', 'ProductRelated_Duration', 'Month_Nov', 'Administrative', 'TrafficType_2', 'ExitRates', 'BounceRates', 'VisitorType_Returning_Visitor', 'TrafficType_3') were those that had a significant positive or negative correlations with the 'Revenue' variable. The KNeighborsClassifier was a good choice I believe because it captured local patterns in the data that more global methods might have missed. For example, if instances with similar feature values tend to have similar 'Revenue' values, then the KNeighborsClassifier can potentially capture this pattern and make accurate predictions. When I started off, the KNeighborsClassifier gave me an accuracy of 87% on the test set, suggesting that it was a reasonable choice for this problem. Below I will explain all the intricacies of the model.

```
              precision    recall  f1-score   support

           0       0.89      0.96      0.92      2055
           1       0.67      0.39      0.49       411

    accuracy                           0.87      2466
   macro avg       0.78      0.68      0.71      2466
weighted avg       0.85      0.87      0.85      2466

[[1975   80]
 [ 250  161]]
```

Precision: Precision is the ability of a classifier not to label a positive sample as negative. For class 0, it's 0.89, meaning that when the model predicts an instance is of class 0, it's correct 89% of the time. For class 1, it's 0.67, meaning that when the model predicts an instance is of class 1, it's correct 67% of the time.

Recall: Recall is the ability of a classifier to find all positive instances. For class 0, it's 0.96, meaning that the model correctly identifies 96% of all actual class 0 instances. For class 1, it's 0.39, meaning that the model correctly identifies 39% of all actual class 1 instances.

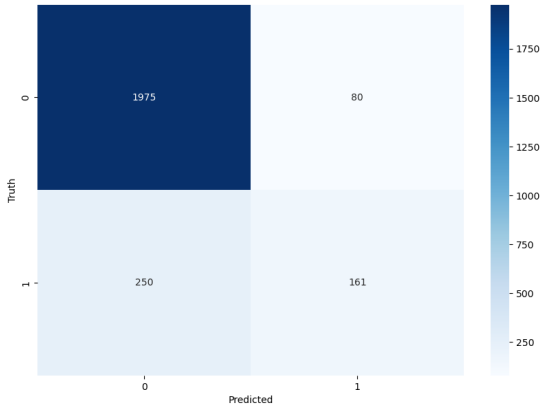
F1-score: The F1 score is the harmonic mean of precision and recall. An F1 score reaches its best value at 1 (perfect precision and recall) and worst at 0. It's a good way to summarize the evaluation of the model when you care about both precision and recall. For class 0, it's 0.92, and for class 1, it's 0.49.

Support: Support is the number of actual occurrences of the class in the dataset. For class 0, there are 2055 instances, and for class 1, there are 411 instances.

Accuracy: This is the overall accuracy of the model, which is 0.87 or 87%. This means that the model correctly predicts the class of an instance 87% of the time.

Macro avg: This is the average precision, recall, and F1 score between classes. It doesn't take class imbalance into account. So if you have class imbalance (i.e., more instances of one class than another), you should look at both the macro avg and the weighted avg to get a better picture of how your model is performing.

Weighted avg: This is the average precision, recall, and F1 score between classes weighted by the number of instances in each class.



Confusion Matrix: The confusion matrix shows the number of correct and incorrect predictions made by the classifier. The rows represent the actual classes and the columns represent the predicted classes. The first row, first column (1975) is the number of true negatives (class 0 correctly classified as class 0), and the second row, second column (161) is the number of true positives (class 1 correctly classified as class 1). The first row, second column (80) is the number of false positives (class 0 incorrectly classified as class 1), and the second row, first column (250) is the number of false negatives (class 1 incorrectly classified as class 0).

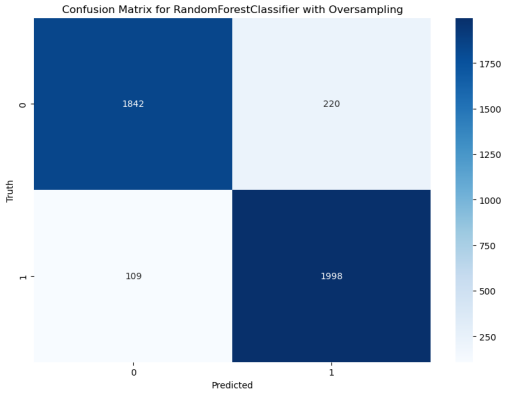
The Next Step in Modeling: Adding more Features to the model and employing SMOTE and Random Forest:

	precision	recall	f1-score	support
0	0.94	0.89	0.92	2062
1	0.90	0.95	0.92	2107
accuracy			0.92	4169
macro avg	0.92	0.92	0.92	4169
weighted avg	0.92	0.92	0.92	4169

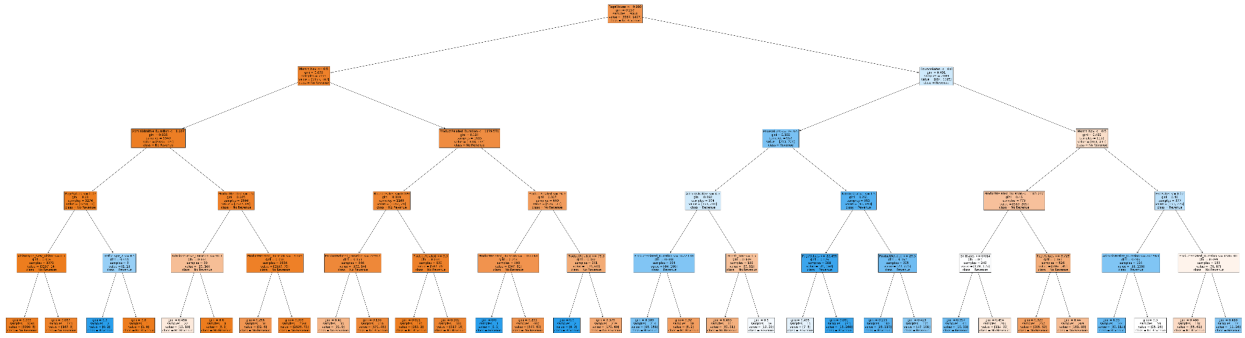
```
[[1842  220]
 [ 109 1998]]
Accuracy:  0.9210841928520028
```

In this step, I used the Synthetic Minority Over-sampling Technique (SMOTE) to balance the dataset, and a RandomForestClassifier to build the model, along with adding more features to the model. I increased the number of features in the model from 10 to 13. The additional features are 'VisitorType_New_Visitor', 'Informational', and 'Administrative_Duration'. These features were added because they were found to be significant in predicting the target variable 'Revenue' during exploratory data analysis.

The results show that the accuracy of the model has improved significantly from 87% to 92.1%. This could be due to a combination of the oversampling technique, the change in the machine learning model, and the addition of more significant features. The classification report and confusion matrix show that the model has a high precision and recall for both classes, indicating that it performs well in predicting both the positive class (Revenue = 1) and the negative class (Revenue = 0). The model correctly predicts 1842 true negatives and 1998 true positives, with 220 false positives and 109 false negatives.

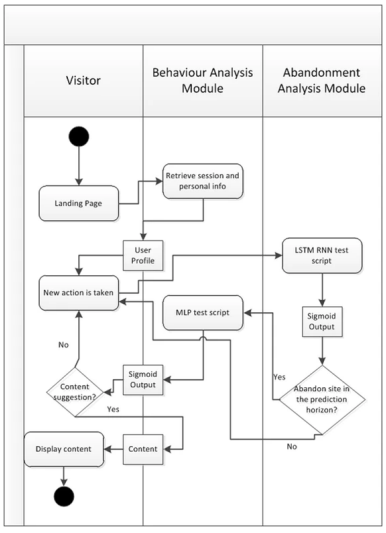


Decision Tree Discussion



There is a much higher resolution decision tree in the folder that this file is submitted with. That being said, given the model-fitting done thus far, the decision tree is comprehensive and is a good exemplification of the dataset's features and how they impact the likelihood of a transaction.

1. PageValues: This is the most important feature according to the decision tree. If the PageValues is greater than 0.99, there is a high chance of generating revenue. Therefore, retail stores should focus on improving the quality of their pages to increase their PageValues.
2. Month_Nov: The decision tree shows that customers are more likely to generate revenue in November. This could be due to holiday shopping for events like Black Friday or Cyber Monday. Retail stores should therefore consider offering special deals or promotions during November to attract more customers.
3. Administrative_Duration: Customers who spend more time on account management pages are more likely to generate revenue, especially when the Administrative_Duration is greater than 1.67. Retail stores should ensure that their administrative pages are user-friendly and contain useful information to keep customers engaged.
4. ProductRelated and ProductRelated_Duration: These features also play a significant role in revenue generation. Retail stores should focus on improving the quality of their product-related pages and try to increase the time customers spend on these pages.
5. BounceRates and ExitRates: Lower bounce rates and exit rates lead to higher chances of revenue generation. Retail stores should analyze the reasons behind high bounce rates and exit rates and take necessary actions to reduce them.
6. VisitorType_New_Visitor: New visitors are more likely to generate revenue compared to returning visitors. Retail stores should therefore focus on attracting new customers, for example through online advertising or social media marketing.
7. TrafficType_2: Traffic type 2 seems to be more beneficial for revenue generation. Retail stores should analyze what this traffic source is and try to attract more traffic from this source.



When comparing my decision tree with the Decision tree that the original paper had come up with it seems as though the decision tree's judgment on PageValues and content engagement being one of the driving factors that predicts positive revenue generation was correct. (Sakar, 2018)

Conclusion:

The key contributors to Revenue generation, according to our analysis, appear to be factors like Web-page engagement, Account management page navigation, the seasonal impact of November, and engagement with Product

Related content. Importantly, our dataset contains an overwhelming majority (85%) of visitor sessions that did not translate into transactions. In such a scenario, the SMOTE and RandomForestClassifier models have demonstrated their utility for accurately predicting non-transactional interactions with the Online Store. There is, however, a notable impact on model performance with varying numbers of features. For instance, the KNeighborsClassifier's accuracy dips slightly, from 87% to 86%, when more than ten features are incorporated for model fitting. Conversely, for the SMOTE and RandomForestClassifier models, this feature augmentation strategy proves beneficial. Adding more features to these models results in an improved accuracy that rises from 87% to 92%. In essence, while feature optimization impacts various predictive models differently, certain models are more adept at predicting non-transactional user sessions, specifically the SMOTE and RandomForestClassifier models, based on our dataset that is largely dominated by such instances. It is imperative to mention that although the RandomForest model worked well, a Support Vector Machine and Long-short-term memory recurrent neural networks (LSTM-RNN) might work much better. These insights inform our strategy for revenue generation and predictive modeling moving forward.

References:

(Albright, 2011)
https://web.archive.org/web/20120128233422/http://www.tampabay.com/news/business/retail/article1202742.ece

(Sakar, 2018) Sakar, C.O., Polat, S.O., Katircioglu, M. et al. Neural Comput & Applic (2018). [Web Link]

(Baluch, 2023) https://www.forbes.com/advisor/business/ecommerce-statistics/

(Fader, 2004) Moe, W. W., & Fader, P. S. (2004). Dynamic conversion behavior at e-commerce sites. Management Science, 50(3), 326-335. https://repository.upenn.edu/cgi/viewcontent.cgi?article=1425&context=marketing_papers

(Montgomery, 2004) Montgomery, A. L., Li, S., Srinivasan, K., & Liechty, J. C. (2004). Modeling online browsing and path analysis using clickstream data. Marketing Science, 23(4), 579-595. https://www.researchgate.net/publication/227442351_Modeling_Online_Browsing_and_Path_Analysis_Using_Clickstream_Data

(Chang, 2005) Chen, M. C., Chiu, A. L., & Chang, H. H. (2005). Mining changes in customer behavior in retail marketing. Expert Systems with Applications, 28(4), 773-781. http://didawiki.cli.di.unipi.it/lib/exe/fetch.php/dm/change-customer-behavior.pdf

(Chen, 2010) Tsai, C. F., & Chen, M. Y. (2010). Variable selection by association rules for customer churn prediction of multimedia on demand. Expert Systems with Applications, 37(3), 2006-2015. https://www.researchgate.net/publication/222415356_Variable_selection_by_association_rules_for_customer_churn_prediction_of_multimedia_on_demand

(Clifton, 2012) Clifton B (2012) Advanced web metrics with Google Analytics. Wiley, New York https://gentecomgente.files.wordpress.com/2013/10/markiert_brian_clifton_advanced_web_metrics_with_google_analytics__2010.pdf

(CASC, 2023) https://www.usgs.gov/programs/climate-adaptation-science-centers/casc-network-and-region-maps