Volume 11, Issue 3, July 2024 ISSN: 2348-8565 (Online)

International Journal of Modern Engineering and Research Technology

Website: http://www.ijmert.org Email: editor.ijmert@gmail.com

Purchase Intention Prediction of Online Shoppers Using XGBoost Machine Learning Model

Shivangi Chouksey
Research Scholar, M.Tech.
Computer Science and Engineering
Takshshial Institute of Engineering and Technology
Jabalpur (M.P.), India
Email: shivangi.chouksey13april@gmail.com

Neha Khare
Assistant Professor
Department of Computer Science and Engineering
Takshshial Institute of Engineering and Technology
Jabalpur (M.P.), India
Email: nehakhare@takshshila.org

ABSTRACT

With the rapid growth of e-commerce, understanding and predicting online shoppers' purchase intentions has become critical for online retailers. This paper focuses on developing a predictive model using the Gradient Boosting machine learning algorithm to accurately predict whether an online shopper is likely to make a purchase. By leveraging the power of Python programming and advanced machine learning techniques, this work aims to provide valuable insights and practical applications for online retailers to optimize their marketing efforts. First, we apply a feature selection technique to select the best features. Then the extracted features are used to train supervised learning models. Different classifiers such as Logistic Regression, Random Forest (RF), Decision Tree (DT), Gradient Boosting, and XGBoost classifiers were used along with the oversampling method to balance the dataset. The experimental results show that the XGBoost classifier with feature selection techniques and oversampling method yielded significantly higher values for the area under the ROC curve (auROC) and the area under the precision-recall curve (auPR), which are 0.937 and 0.754, respectively. On the other hand, the precision achieved by XGBoost and the decision tree is significantly better and is 90.65% and 90.54%, respectively. The overall performance of the XGBoost classifier is significantly better compared to other classifiers.

Keywords:—*Machine learning, Logistic Regression (LR), Random Forest (RF), Decision Tree (DT), Gradient boosting, XGBoost Classifier, area under the ROC curve (auROC), Area under the precision – recall- curve (auPR).*

I. INTRODUCTION

The rise of e-commerce has transformed the retail landscape, necessitating advanced analytical methods to understand and predict consumer behavior. Purchase intention prediction is a critical component for online retailers aiming to tailor their marketing efforts, personalize user experiences, and ultimately increase sales. Traditional statistical methods often fall short in capturing the complex, non-linear relationships present in user behavior data. This paper investigates the use of the XGBoost algorithm, known for its high performance in classification tasks, to predict the purchase intentions of online shoppers.

Research Objectives

Develop a Prediction Model: Create an accurate model to predict online shoppers' purchase intentions using the XGBoost classifier.

Feature Selection: Identify and select the most relevant features that influence purchase intention from online shoppers' behavior data.

Address Data Imbalance: Apply oversampling techniques to balance the dataset, improving the model's learning from both purchase and non-purchase instances.

Performance Evaluation: Assess the model's effectiveness using metrics like auROC, auPR, and accuracy, and compare it with other classification algorithms.

Explainability Analysis: Interpret the model's predictions, analyze feature importance, and provide insights into the factors influencing purchase intention in e-commerce.

Achieving these objectives will enhance e-commerce strategies by accurately predicting purchase intentions, improving customer targeting, and optimizing online shopping interventions.

Machine Learning and Its Types

Machine Learning (ML) is a branch of artificial intelligence where computers learn from data to make predictions or decisions without explicit programming.

Types of Machine Learning:

Supervised Learning: Models learn from labeled data to predict outcomes. ***Examples:*** Classification (spam detection), Regression (house price prediction).

Unsupervised Learning: Models identify patterns in unlabeled data.Examples: Clustering (customer segmentation), Dimensionality Reduction (PCA).

Semi-Supervised Learning: Combines labeled and unlabeled data to improve learning accuracy.

Reinforcement Learning: Models learn by interacting with an environment, optimizing for cumulative rewards.

Examples: Game playing (AlphaGo), Robotics.

Machine learning enables systems to improve their performance over time by learning from data, providing powerful tools for various applications.

Supervised Machine Learning Algorithms

Decision Trees: Models that split data into branches to make predictions based on feature values.

Random Forests: Ensembles of decision trees that improve accuracy and reduce overfitting by averaging multiple trees' predictions.

Support Vector Machines (SVM):Models that find the optimal hyperplane to separate classes in the feature space.

Naive Bayes: Probabilistic models based on Bayes' theorem, assuming feature independence.

Logistic Regression: Models that predict binary outcomes using a logistic function.

Neural Networks: Models inspired by the human brain, consisting of layers of interconnected neurons that learn complex patterns.

These algorithms provide various approaches to solving classification and regression tasks in machine learning.

II. Review of Existing Studies

Numerous studies have addressed predicting purchase intentions in e-commerce using real-time data from online shopping platforms. Key studies include:

S. Ahsainet et al. (2022):Emphasizes the role of AI and machine learning in CRM, proposing a modular e-commerce framework to predict purchase likelihood, enhancing customer engagement and platform operations.

Batta Mahesh et al. (2020): Reviews various machine learning algorithms, highlighting their applications in different domains and the importance of choosing appropriate algorithms for specific problems.

Ramazan Esmeli et al. (2020): Proposes a model using session and log data with utility scoring, finding Decision Trees to be highly effective (97% auROC) for predicting purchase intentions within ongoing sessions.

G. Saranya et al. (2020): Utilizes digital marketing and machine learning (SVM-Linear) to predict customer purchase intentions, focusing on analyzing product ratings and prices through sentiment analysis.

Sujoy Bag et al. (2019):Predicts purchase intentions by analyzing social perception scores and sentiments for durable goods, employing regression analysis to identify significant attributes.

Paulo Rita et al. (2019): Develops an e-service quality model emphasizing website design, security, and fulfilment in predicting consumer behavior and trust.

José Martins et al. (2019): Combines web advertising and flow experience theories to study the impact of smartphone advertisements on online shopping behavior, highlighting the importance of interactive design and brand recognition.

Xiao et al.: Identifies online promotion, content marketing, personalized recommendations, and social reviews as key cues influencing cross-border e-commerce purchases.

Sorim Chung et al. (2018): Explores the impact of touch interfaces on online shopping, finding increased engagement and a preference for hedonic products.

These studies collectively enhance the understanding of factors influencing online purchase intentions, providing insights into effective predictive modeling and customer engagement strategies in e-commerce.

III. Proposed System Architecture

Our proposed prediction model tracks online shopper behavior in real-time, analyzing actions such as product views and cart additions. This behavioral data is processed using the XGBoost algorithm to predict purchase intentions during the current session. Based on predictions, personalized recommendations and offers are made to the shopper. Predictions and outcomes are recorded to continuously update and improve the model, ensuring accuracy by incorporating new shopper behavior data regularly. This adaptive approach helps maintain the model's effectiveness in predicting purchase intentions and enhancing customer engagement.

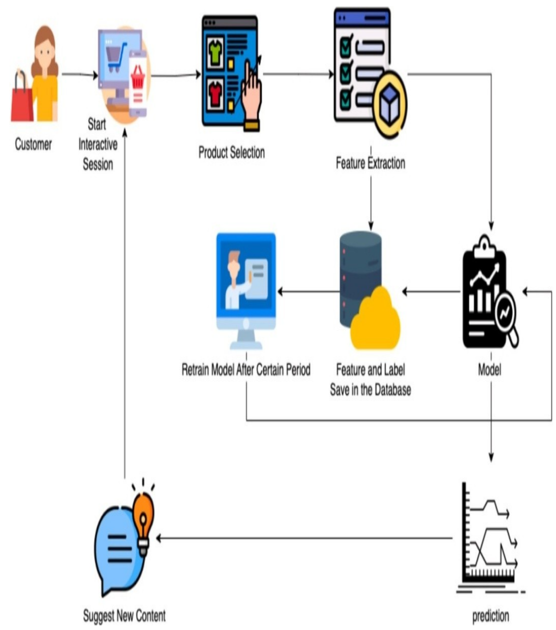


Figure 1: Proposed System Architecture

Training of the Model

We train our prediction model by collecting data from an e-commerce website and performing data pre-processing, including converting categorical data to numerical and scaling. Key features are selected to enhance model performance, and data imbalance is addressed using balancing techniques. Various classifiers are trained and evaluated to identify the best one for predicting purchase intentions. This process ensures the development of an accurate model for understanding and predicting user behavior on e-commerce platforms.

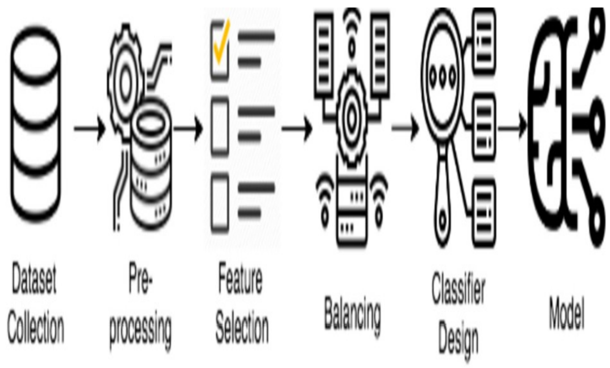


Figure 2: Workflow for training our proposed model.

Dataset Used

In this work, we tackled the task of predicting whether online shoppers will make a purchase or not, which we formulated as a supervised binary classification problem.

The dataset contains various types of features obtained from real-time user interactions on the e-commerce platform. These features include information about the pages visited, duration of visits, product-related interactions, and account management activities. We also considered factors such as user actions, browsing patterns, and client system information (like browser type, location, and operating system) as important features.

Table 1: Numerical Features in the Dataset

Feature Name	Data Type	Feature Description
Administrative	Integer	Number of different pages visited related to the administrative concerns of the website
Administrative Duration	Integer	Total amount of time (in seconds) spent by the visitor on account management related pages
Informational	Integer	Number of different pages visited related to the information of the website and other useful contents of the website
Informational Duration	Integer	Total amount of time (in seconds) spent by the visitor on informational pages
Product Related	Integer	Number of different pages visited related to different products of the website.
Product Related Duration	Integer	Total amount of time (in seconds) spent by the visitor on product related pages
Bounce Rate	Float	Average bounce rate value of the pages visited by the visitor
Exit Rate	Float	Average exit rate value of the pages visited by the visitor
Page Value	Float	Page Value is the average value for a page that a user visited before making a transaction.
Special Day	Float	The "Special Day" feature indicates the closeness of the site visiting time to a specific special day

Table 2: Categorical Features in the Dataset

Categorical Features in the Dataset:

Feature Name	Data Type	Feature Description
Browser	Integer	ID of bowsers from which the session took place.
Region	Integer	ID of Regions from which the session took place.
Traffic Type	Integer	ID of different types of sources from which the users landed on the website.
Visitor Type	String	Visitor type as "New Visitor", "Returning Visitor" and "Other"
Weekend	Boolean	Whether the session was on a weekend or not.
Operating Systems	Integer	Operating system of the visitor
Month	Boolean	Month value of the visit date
Revenue (Target Variable)	Boolean	Whether the user contributed to the revenue by purchasing or not.

Data Preprocessing

We employed two key preprocessing techniques:

One-hot Encoding

Categorical data was converted to numeric form using one-hot encoding, creating binary columns for each category (e.g., "red" becomes [1, 0, 0]).

Standard Scaling

Numeric features were standardized to have a mean of 0 and a standard deviation of 1, ensuring uniformity across features and improving model performance.Class labels were also converted to binary form using label encoding**.**

Feature Selection

We used the chi-squared (χ^2) method to select the most relevant features from our dataset, identifying 9 out of 18 significant features, such as "Administrative", "BounceRates", "ExitRates", and "PageValues". This method ensures simplicity and efficiency, suitable for real-time prediction**.**

Addressing Data Imbalance

To tackle class imbalance, we used two resampling techniques:

Resampling Methods

SMOTE (Synthetic Minority Oversampling Technique): Generates synthetic samples for the minority class.

Random Under sampling: Reduces the number of samples in the majority class.

These methods balanced the training set, improving the model's ability to predict both classes accurately.

Classification Algorithms Comparison

We compared several classification algorithms:

XGBoost: Builds a model iteratively using decision trees, optimizing performance and preventing overfitting.

Decision Tree (DT): Uses entropy for attribute selection, with a maximum tree depth of 5.

Support Vector Machine (SVM): Uses the "rbf" kernel and a regularization parameter (C) value of 7.

Gradient Boosting*:* Sequentially builds models to correct previous errors.

Random Forest: Combines multiple decision trees with a maximum depth of 20 and 100 estimators.

These algorithms were chosen for their effectiveness in predicting online shopping behavior.

Algorithm (XGBoost Classifier)

Initialization:

- Start with a training set D containing pairs of input
- features (χi) and corresponding labels (yi).
- Define a loss function (f(χi), yi).
- Choose a learning rate α.
- Decide on the number of weak learners M.
- ***Iteration:***
- Begin a loop from t = 2 to M.
- For each iteration t:
- Compute the gradients ($\hat{g}$(χi)) and hessians ($\hat{h}$ m(χi)) for each data point i in the training set.
- Update the model using the computed gradients and hessians.
- Move to the next iteration.

Finalization:

After completing all iterations, return the final model $f(\chi)$.

Updating the Model:

Update the model $f(\chi)$ by adding the result of the weak learner for each iteration to the previous model ft−1(χ).

Return the Final Model:

Return the final model $f(\chi)$ obtained after all iterations.

This algorithm essentially iteratively improves a model by adding weak learners and updating the model based on the gradients and hessians computed from the training data. Finally, it returns the optimized model for making prediction

IV. Implementation and Result

We used Python 3.7 and scikit-learn within Jupyter notebooks, splitting the dataset into 70% training and 30% testing. Feature selection, oversampling, and under sampling were applied only to the training data. Experiments began with a basic setup and were expanded to evaluate various components, with results detailed in each section.

Data Visualization

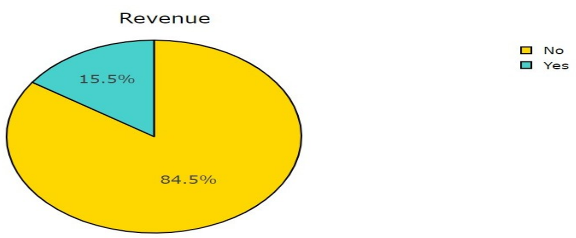


Figure 3: Data set sales ratio visualization

- 84.5% of the transactions did not turn into sales
- 15.5 % of transactions turned into sales
- Highly Imbalanced data set.

Low sales conversion rates are worrisome for businesses and require attention. The data we have includes different details about what customers do on the website. We'll use this information to figure out which factors are most likely to influence customers to make a purchase

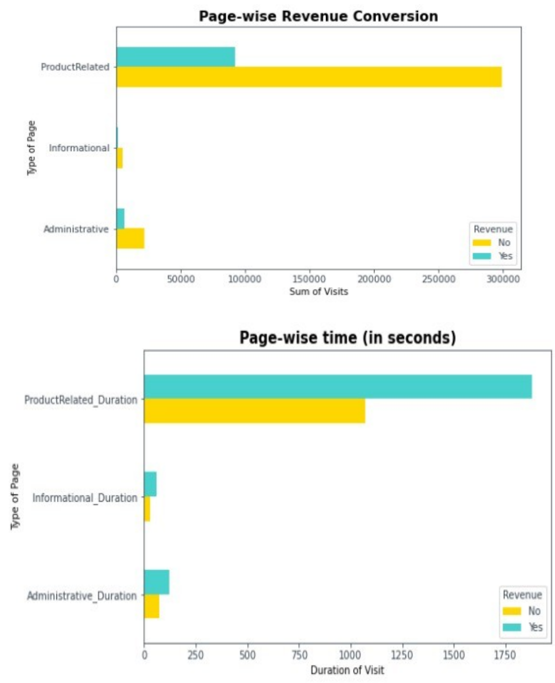
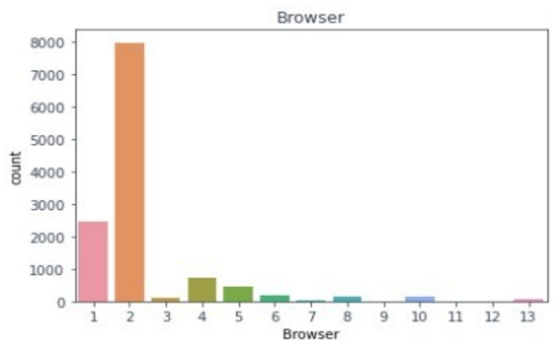
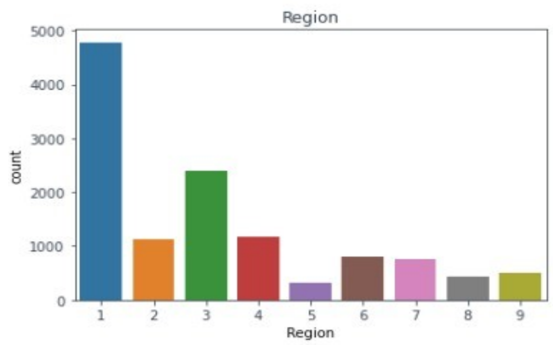
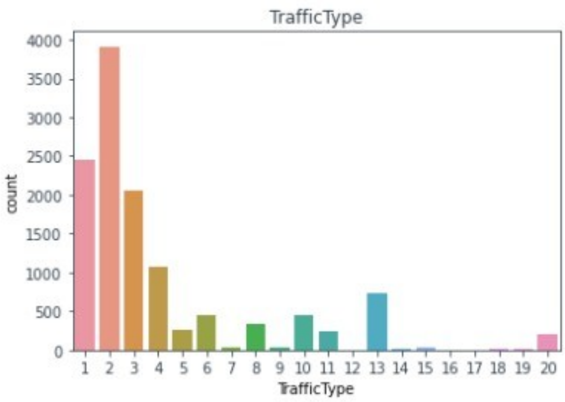


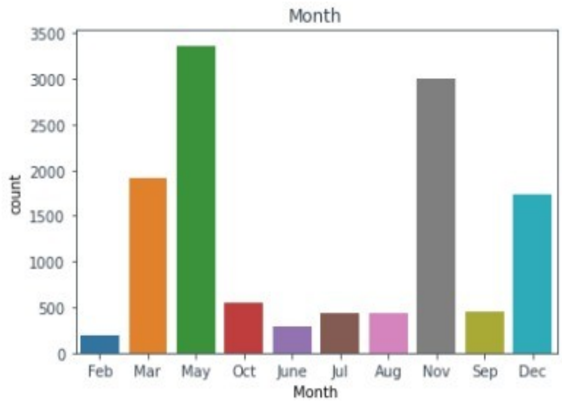
Figure 4: Page-wise Revenue conversion & Time

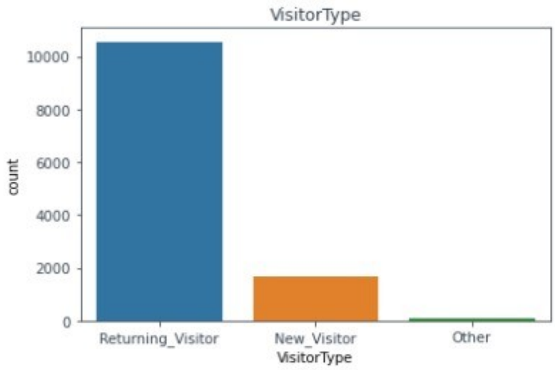
Most visits to the website involve viewing product pages, which also contribute the most to revenue generation. Therefore, these product pages are crucial for the client due to their popularity and revenue impact.

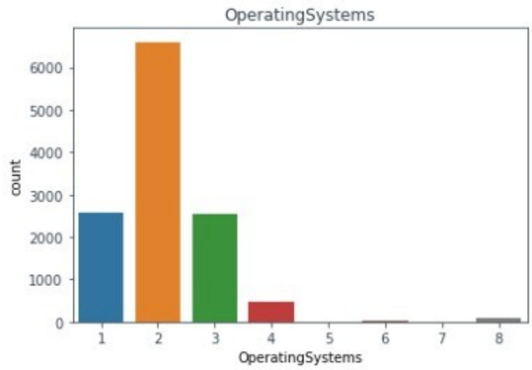


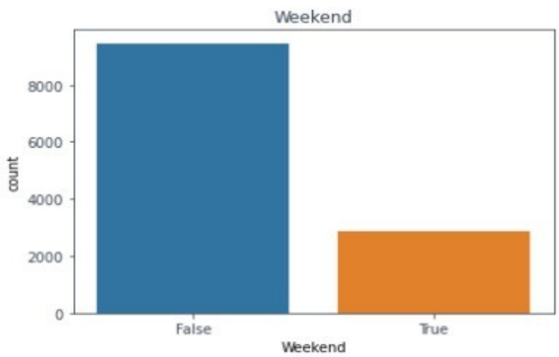












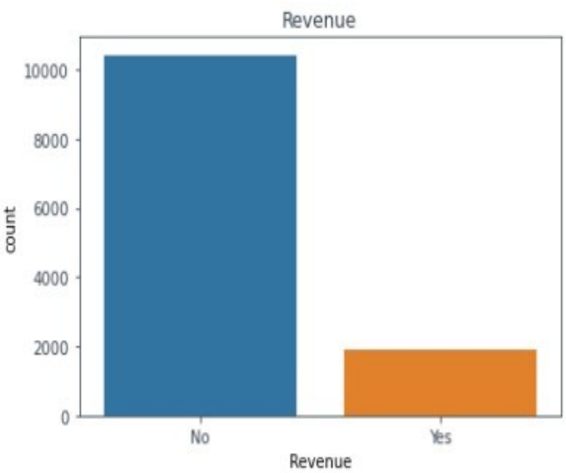


Figure 5: Analysis of Different Features using Counting Plots

Above counting plots shows following findings-
Most of the sessions happened on the month May followed by November.

Most of the sessions happened from systems having OS, Browser No, Traffic Type 2 and region 1.

Visitors are returning visitors.

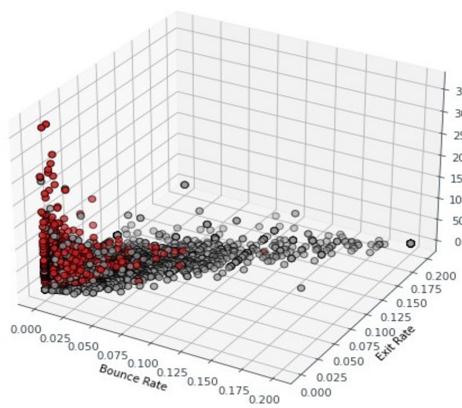


Figure 6: 3D view of Bounce Rate, Exit Rate and Page Value

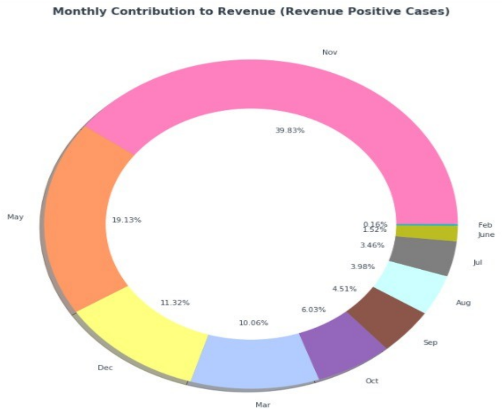


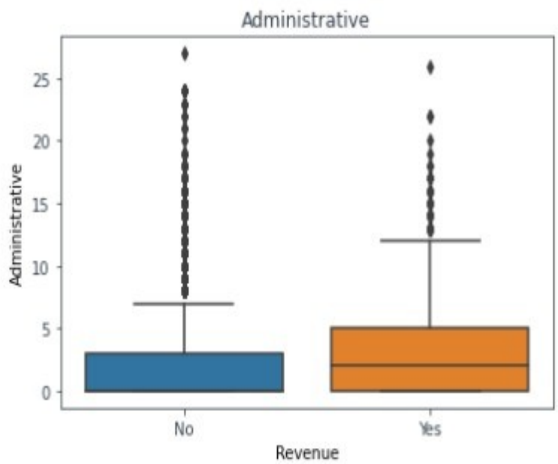
Figure 7: Monthly Contribution to Revenue

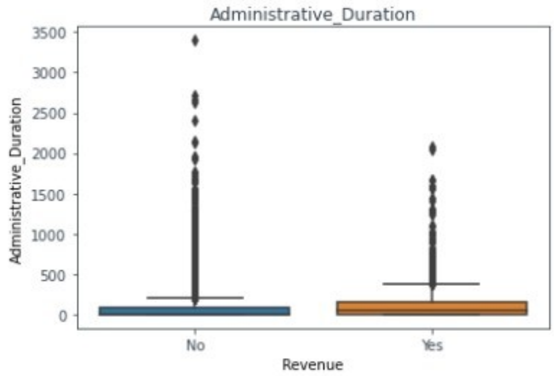
Here, Red points represent customers who did not bring revenue and Black points represent revenue positive customers. From the above visualization, following points are noted:

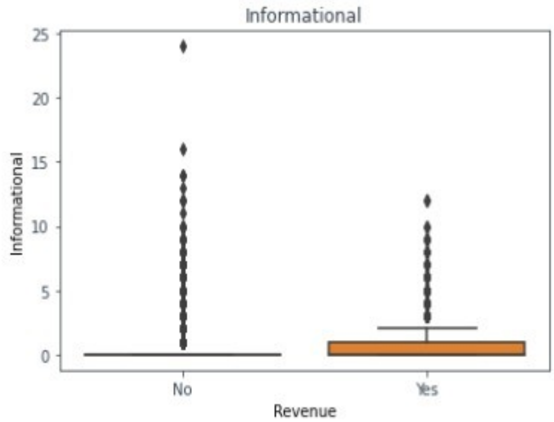
Customers who ended up shopping (Reds) have a relatively lower bounce rate and exit rate than the black points.

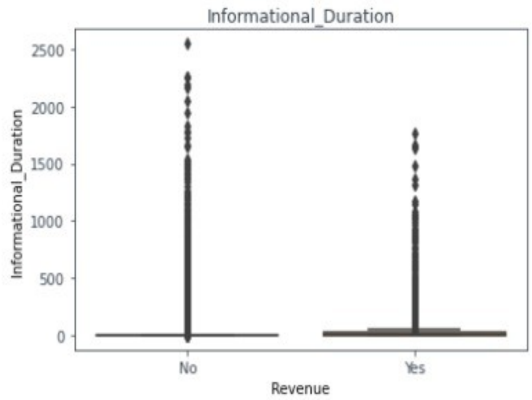
Customers who ended up shopping have a page value on the higher side relative to the negative classes.

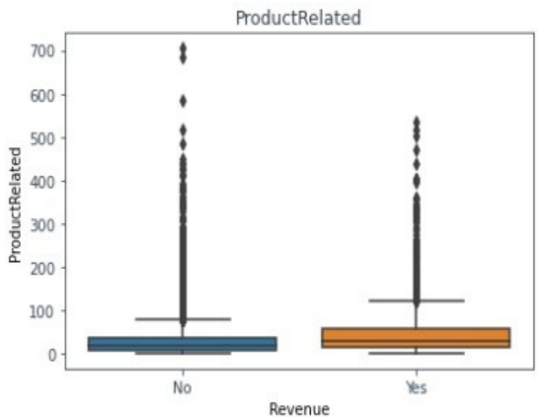
From the above doughnut plot, it can be inferred that November has the most significant contribution to client's revenue followed by May, December and March. While February, June and July have the least contribution. Client needs to analyze what drives the customers during the peak months and implement similar strategies in the other months











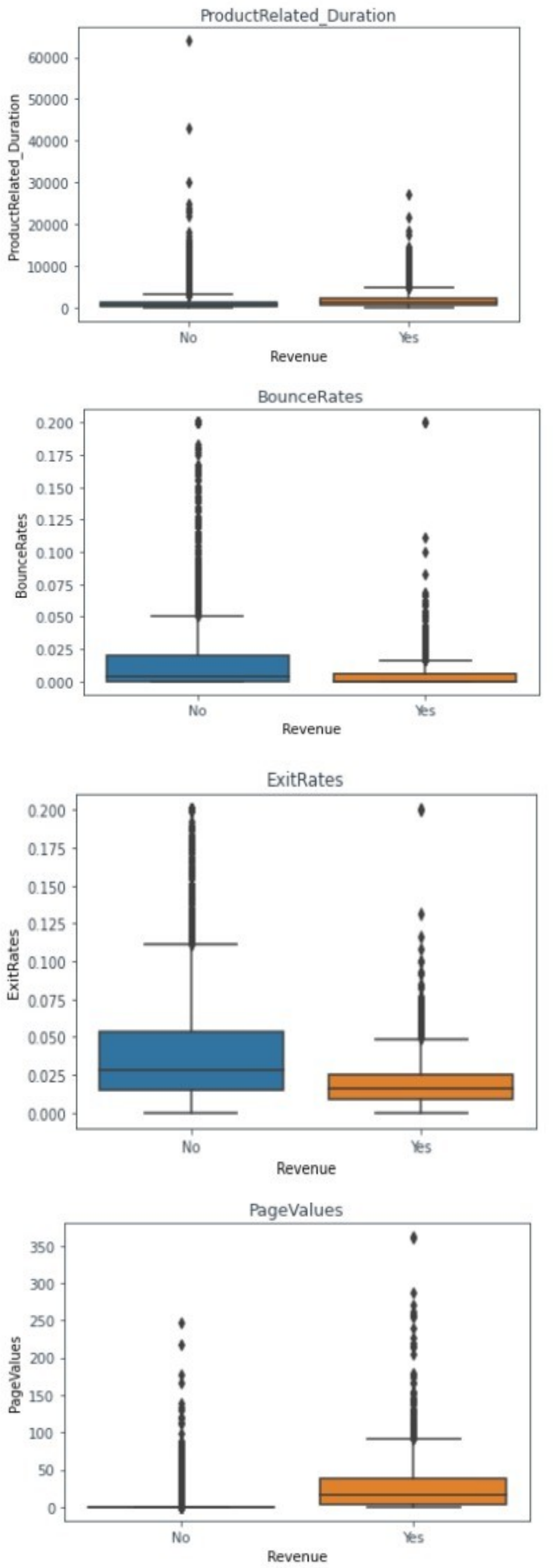


Figure 8:Plotting all numerical variables with target variable

Findings-

Median number of pages visited and time spent on them for administrative purposes on the website is higher for sessions which yielded revenue

It appears that median number of visits to Informational pages and time spent on them is approximately 0.

Median number of product related pages visited and time spent on them is higher for sessions which yielded revenue.

However, there appears to be a lot of outliers where no-revenue sessions had users visiting a lot of pages and spending time on them but not purchasing the product.

Bounce Rates and exit rates of pages were relatively lower for sessions which yielded revenue.

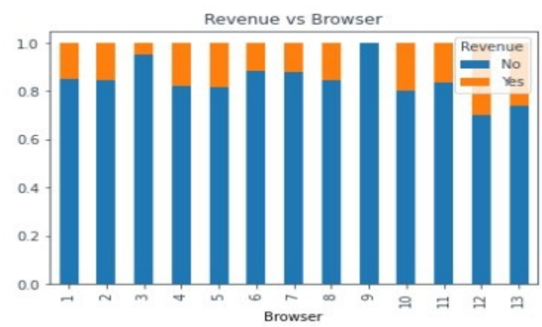
This shows that it is important for websites to have a lower bounce rate and exit rates on their pages.

Similarly Average Page value of a page that user visited before making a transaction is visibly higher in revenue positive sessions

Visually, it appears that closeness to a special day stays at median value of 0 for both revenue and non-revenue cases.

This is because majority of the times, there is no closeness to a festival.

Plotting all categorical variables with target variable



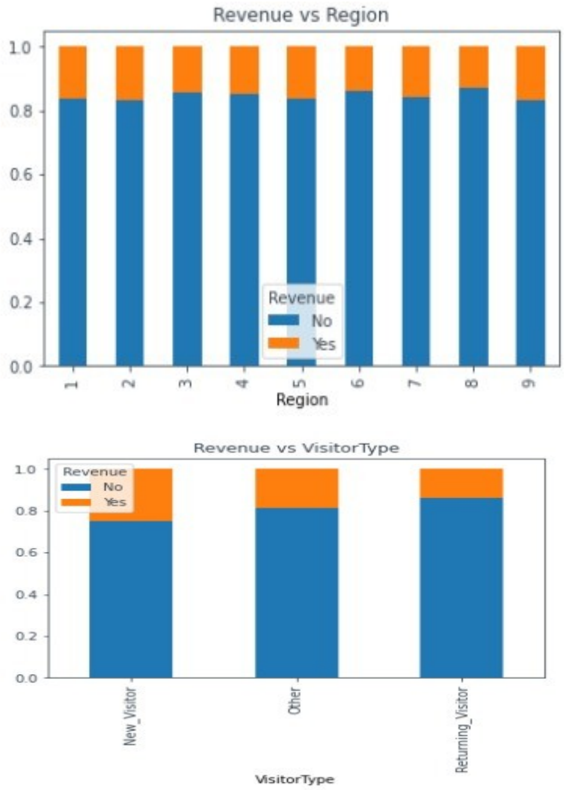


Figure 9:Plotting all categorical variables with target variable

Findings

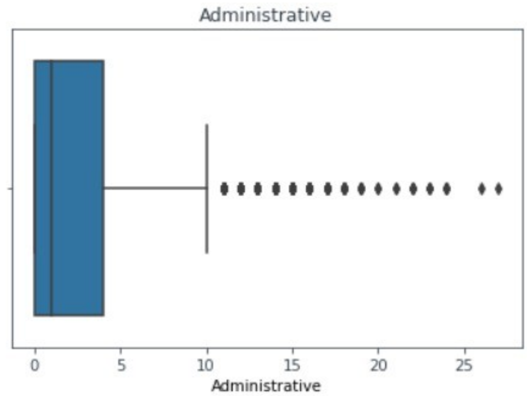
It appears proportion of session which yielded revenue is the highest in November.

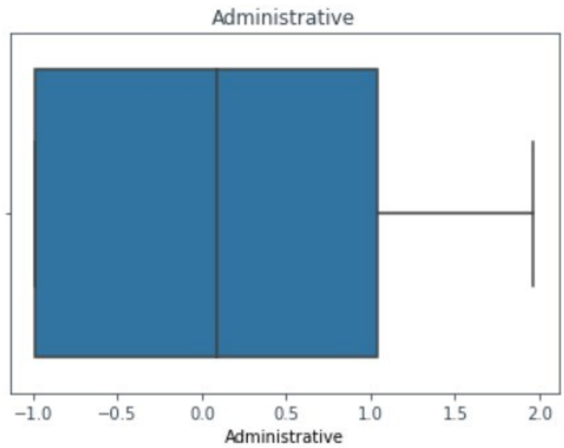
- February has the lowest proportion of sessions which yielded revenue
- Proportion of sessions which yielded revenue is the highest among users having operating system 8
- While operating system 3 yields the lowest proportion of these revenue positive sessions
- Browser 12 has the highest proportion of revenue positive sessions followed by browser 13 while browser 3 has the least contribution
- Region does not appear to play a significant role in determining whether the customer will purchase or not.
- Proportion of revenue positive sessions is the highest among new visitors followed by other visitors.
- Returning visitor sessions have a relatively lower proportion of revenue positive sessions.
- If it's a weekend there are slightly more chances of having a sale! Visually, the difference does not appear to be a very big one.

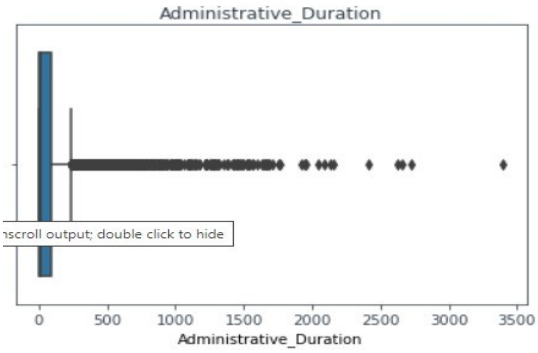
Data Preprocessing by applying Power Transformation

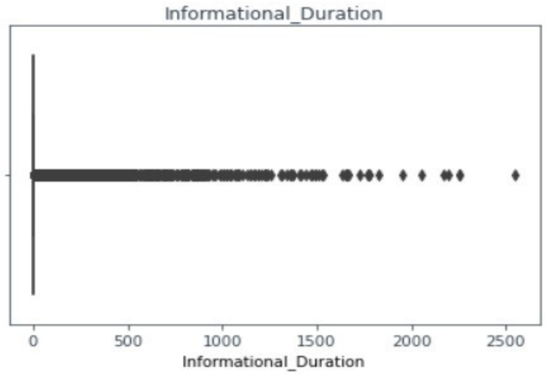
Despite the presence of numerous outliers in the data, they appear to be realistic. Hence, instead of removing these outliers, we will address them by applying a Power Transformation.

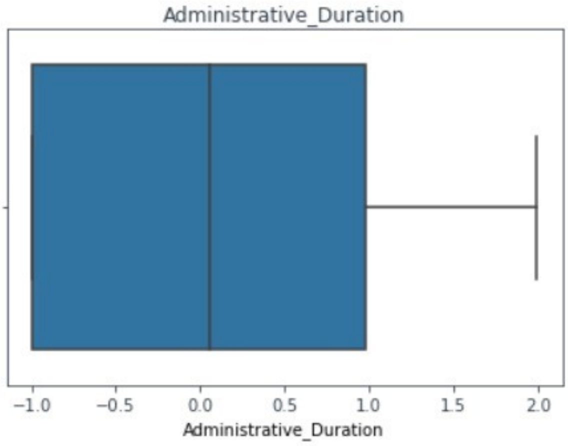
Before Power Transformation After Power Transformation











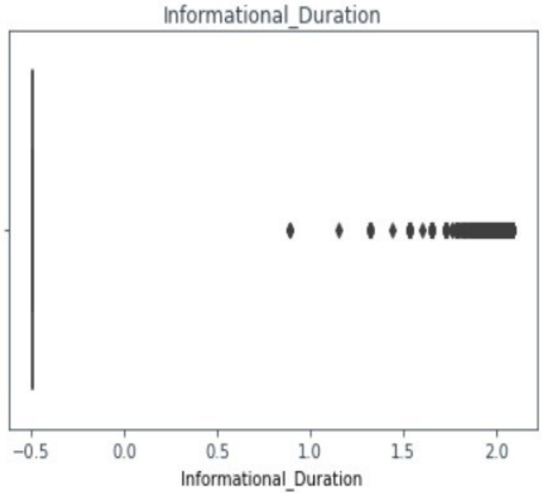
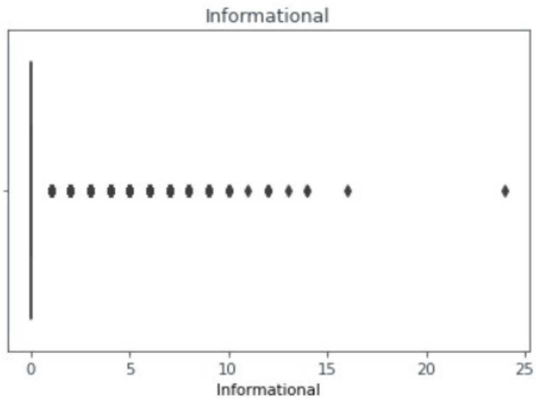
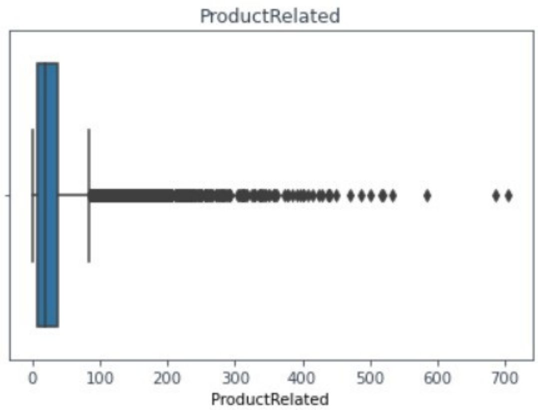
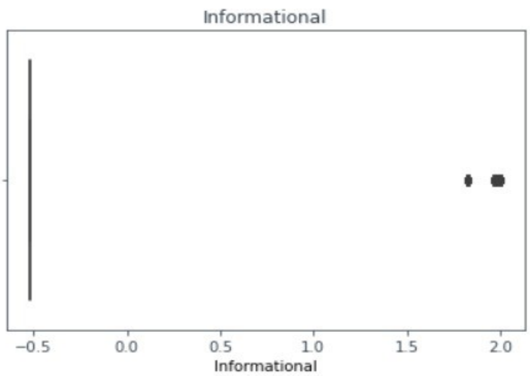
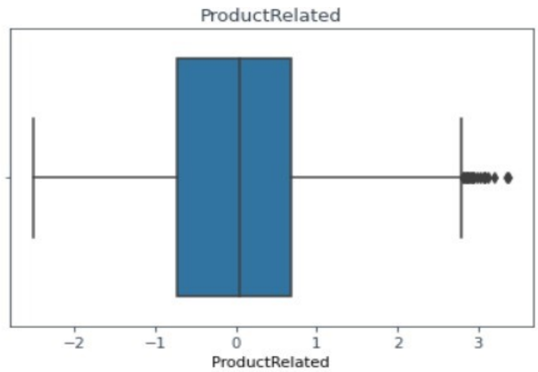


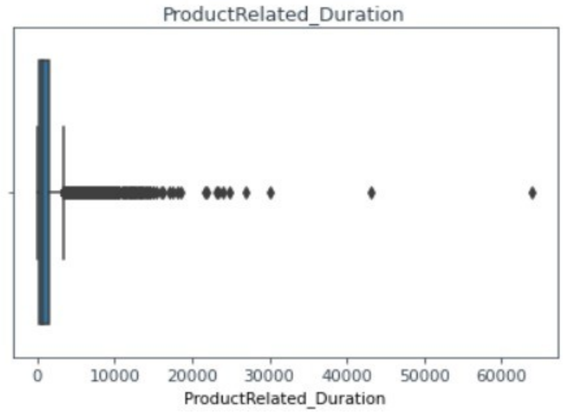
Figure 10:Plotting all numerous variables outliers handling by applying Power Transformation

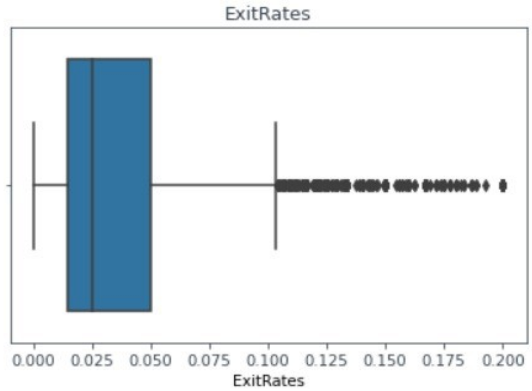


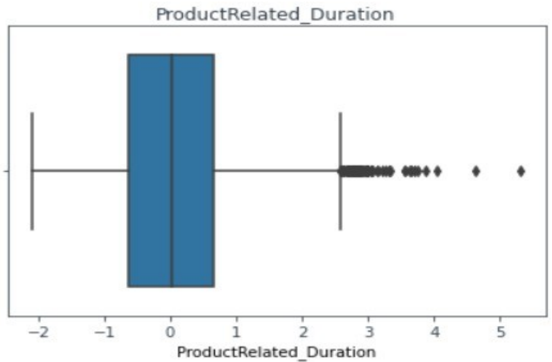


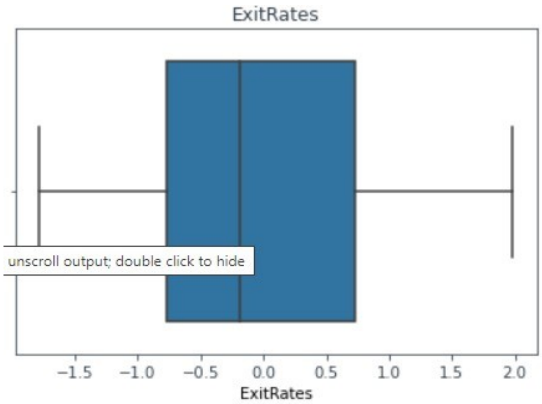


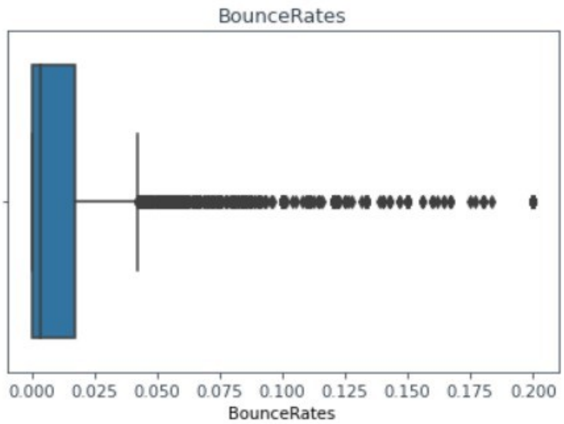


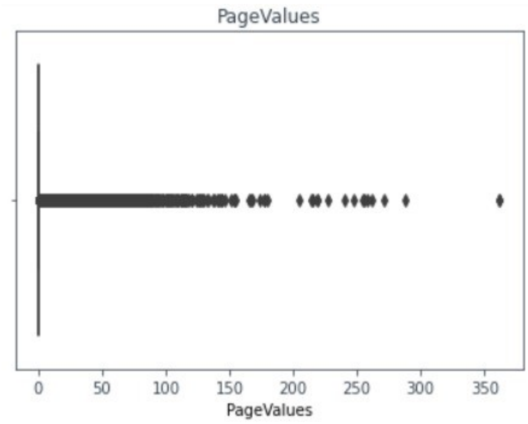

ProductRelated_Duration
0 10000 20000 30000 40000 50000 60000
ProductRelated_Duration

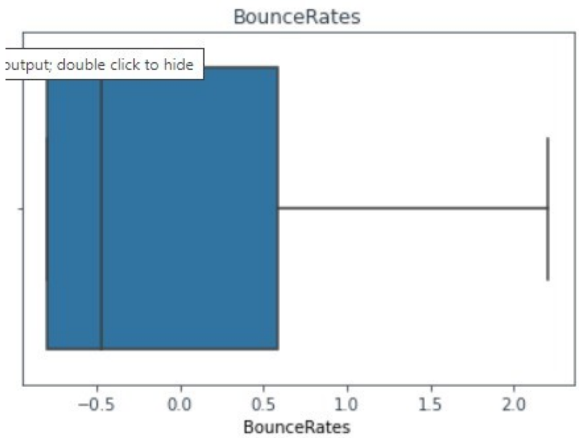

ExitRates
0.000 0.025 0.050 0.075 0.100 0.125 0.150 0.175 0.200
ExitRates

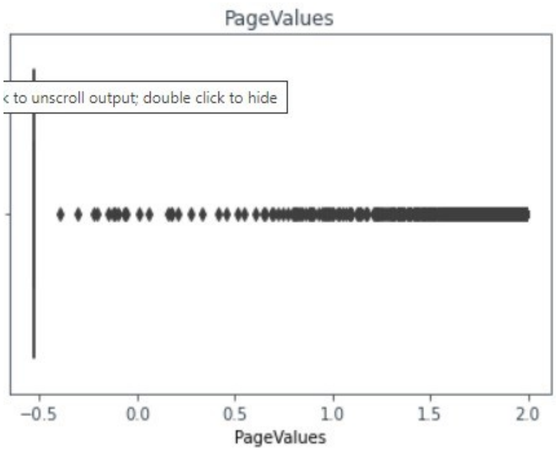

ProductRelated_Duration
−2 −1 0 1 2 3 4 5
ProductRelated_Duration

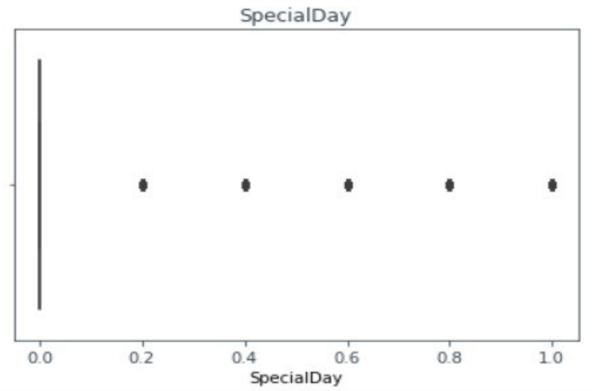

ExitRates
unscroll output; double click to hide
−1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0
ExitRates

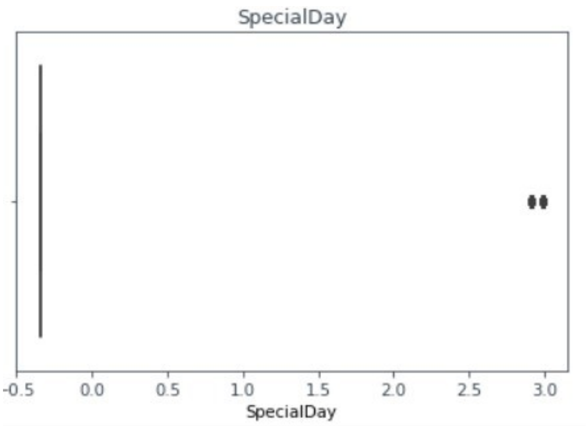

BounceRates
0.000 0.025 0.050 0.075 0.100 0.125 0.150 0.175 0.200
BounceRates


PageValues
0 50 100 150 200 250 300 350
PageValues


BounceRates
output; double click to hide
−0.5 0.0 0.5 1.0 1.5 2.0
BounceRates


PageValues
to unscroll output; double click to hide
−0.5 0.0 0.5 1.0 1.5 2.0
PageValues

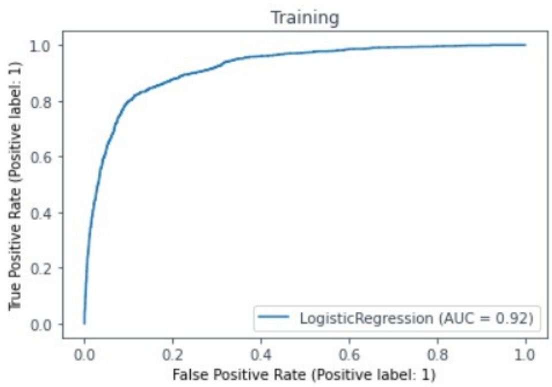




Result of Logistic Regression on Base Model without and with Transformation

Table 3

	Base Model	Base Model - Power Transformed
Train Accuracy	0.884023	0.898969
Test Accuracy	0.890241	0.898351
Train Precision	0.761773	0.707617
Test Precision	0.715789	0.657084
Train Recall	0.398840	0.626541
Test Recall	0.385633	0.604915
Train f1 score	0.523560	0.664615
Test f1 score	0.501229	0.629921
Train ROC AUC Score	0.899392	0.918692
Test ROC AUC Score	0.891306	0.903650



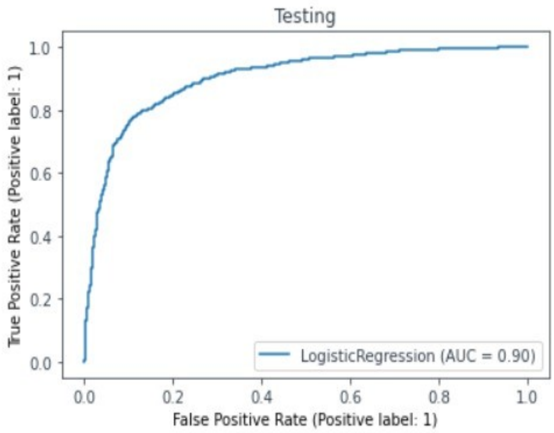
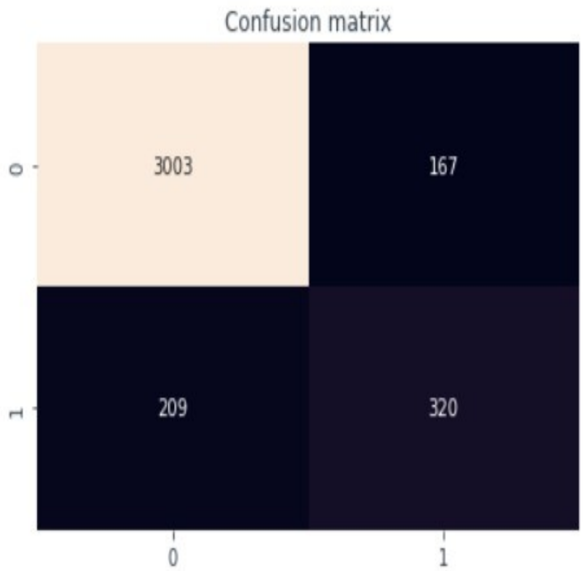


Figure 11:AUC Curve

After Power Transformation Before Power Transformation

All the evaluation metrics, except precision, show improvement when using power transformation. Additionally, the ROC-AUC scores also improve with the power transformation.

Confusion Matrix



In this analysis, 167 sessions that did not result in a sale were incorrectly predicted as revenue-positive. Additionally, 209 sessions that actually generated revenue were wrongly classified as non-revenue sessions. This indicates an area that requires improvement.

Steps to Improve the Model

To improve the model, the following steps will be taken:

- Apply different parametric and non-parametric models.
- Perform hyper-parameter tuning on these models.
- Use ensemble techniques such as Random Forest, XGBoost,Gradient Boosting etc.
- Implement feature selection strategies.
- Use SMOTE analysis to check if performance improves on the test data.

A 5-fold cross-validation strategy will be used to check bias and variance errors, and the scoring parameter will be 'ROC-AUC' because it is a more robust metric. The model that performs best will be evaluated on the test data.

Table 4:

	Model	Bias Error	Variance error
0	Logistic Regression	0.085718	0.004431
1	Tuned KNN	0.076209	0.003316
2	Bagged KNN	0.076331	0.002686
3	Regularized DT	0.078047	0.003577
4	Tuned Random Forest	0.076762	0.002871
5	Gradient Boosting Classifier	0.066663	0.003609
6	XGboost	0.066374	0.003462

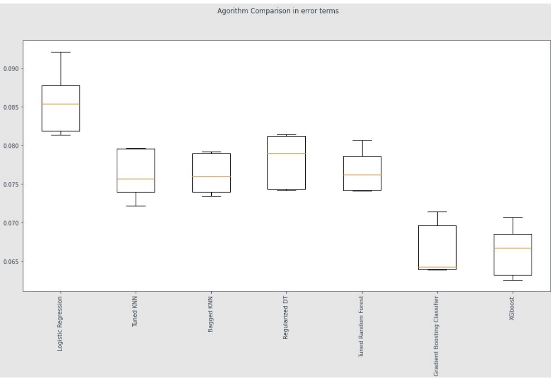


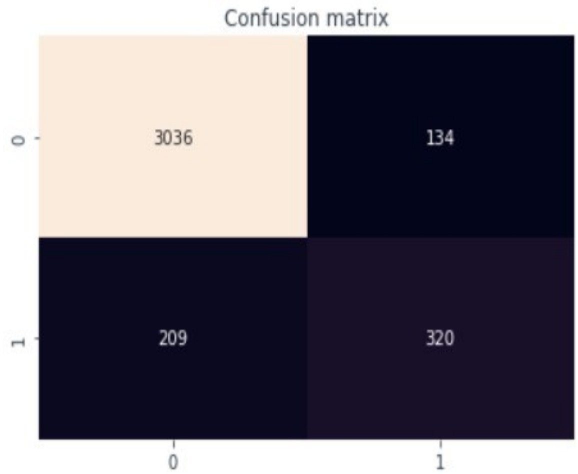
Figure 12: XG Boost performs best with respect to Bias Error and Variance Error

Table 5 :Final Model Test Results without using SMOTE

	Base Model	Base Model - Power Transformed	Final Model without SMOTE
Train Accuracy	0.884023	0.898969	0.916348
Test Accuracy	0.890241	0.898351	0.907272
Train Precision	0.761773	0.707617	0.781491
Test Precision	0.715789	0.657084	0.704846
Train Recall	0.398840	0.626541	0.661349
Test Recall	0.385633	0.604915	0.604915
Train f1 score	0.523560	0.664615	0.716418
Test f1 score	0.501229	0.629921	0.651068
Train ROC AUC Score	0.899392	0.918692	0.950045
Test ROC AUC Score	0.891306	0.903650	0.926179

This model is effective because all the metrics—accuracy, precision, recall, F1 score, and ROC AUC—have significantly improved compared to the base model.

Confusion Matrix



In the base model, 167 sessions were incorrectly predicted as revenue-positive but were actually revenue-negative. This number has now dropped to 135. Additionally, the sessions that were actually generating revenue but not detected decreased slightly from 209 to 203. However, the reduction in false negatives is not very significant compared to the base model.

Final Model Test Results with SMOTE

Confusion Matrix

The number of sessions wrongly predicted as revenue-positive but were actually revenue-negative increased from 167 to 389. However, the number of sessions that actually generated revenue but were not detected as such decreased significantly from 209 to 96. This shows a major improvement from the base model in terms of reducing false negatives.

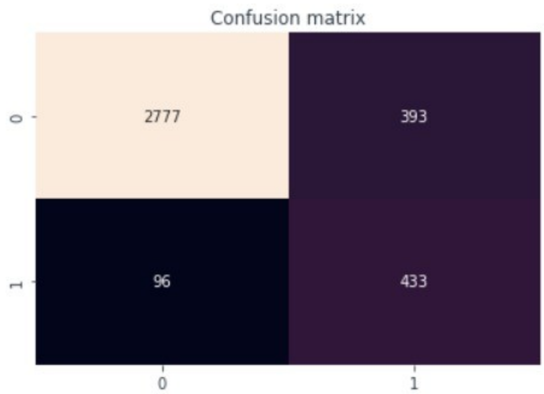


Table 6:

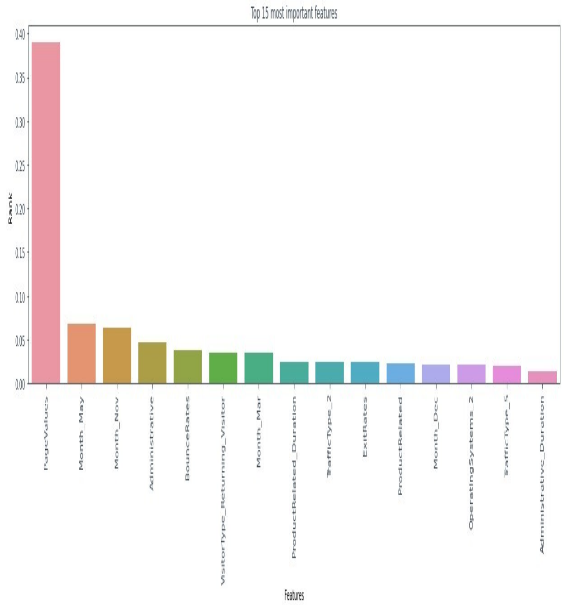
	Base Model	Base Model - Power Transformed	Final Model without SMOTE	Final Model with SMOTE
Train Accuracy	0.884023	0.898969	0.916348	0.897408
Test Accuracy	0.890241	0.898351	0.907272	0.867802
Train Precision	0.761773	0.707617	0.781491	0.883551
Test Precision	0.715789	0.657084	0.704846	0.524213
Train Recall	0.398840	0.626541	0.661349	0.915472
Test Recall	0.385633	0.604915	0.604915	0.818526
Train f1 score	0.523560	0.664615	0.716418	0.899228
Test f1 score	0.501229	0.629921	0.651068	0.639114
Train ROC AUC Score	0.899392	0.918692	0.950045	0.964608
Test ROC AUC Score	0.891306	0.903650	0.926179	0.923725

If the goal of the model is to minimize undetected revenue sessions (False Negatives), even at the cost of incorrectly predicting non-revenue sessions as revenue sessions, then the SMOTE model should be chosen. If the company aims to focus solely on recall value, a SMOTE analysis is recommended. However, the SMOTE model exhibits a high level of overfitting.

For the current analysis, we will proceed with a more generalized model, without using SMOTE.

Table 7: Feature Importance

	Features	Rank
8	PageValues	0.389630
16	Month_May	0.067288
17	Month_Nov	0.063198
0	Administrative	0.046533
6	BounceRates	0.037560
59	VisitorType_Returning_Visitor	0.035257
15	Month_Mar	0.033963
5	ProductRelated_Duration	0.024068
39	TrafficType_2	0.023568
7	ExitRates	0.023553



XGBoost performs very well on this data. If the goal is to reduce the number of missed revenue sessions (false negatives) even if it means mistakenly predicting non-revenue sessions as revenue sessions, then the SMOTE model should be used. If the company wants to prioritize recall, then SMOTE analysis is the way to go. However, the SMOTE model has a high level of overfitting.

V. Summary of Key Findings

XGBoost effectively predicts revenue sessions, with product-related pages being crucial for revenue. SMOTE helps address data imbalance but may cause overfitting. Peak months like November and May are key for marketing. Reducing bounce rates, improving page value, and focusing on administrative pages are important. Operating system and browser choices, particularly Android users on Google Chrome, significantly impact revenue, while region has less influence.

VI. Future Work

Future research should explore additional methods for handling imbalanced datasets, analyse user behaviour on product pages, test marketing strategies during peak times, and improve page value and visitor engagement. Further investigation into the impact of operating systems and browsers on revenue is also recommended.

REFERENCES:

[1] S. AHSAIN and M. AIT KBIR, “Predicting the client’s purchasing intention using Machine Learning models”, E3S Web of Conferences 351, 01070 (2022) ICIES’22.

[2] Batta Mahesh. “Machine Learning Algorithms-A Review”, International Journal of Science and Research (IJSR) ISSN: 2319-7064 ResearchGate Impact Factor (2018): 0.28 | SJIF (2018): 7.426.

[3] Ramazan Esmeli · Mohamed Bader-El-Den· Hassana Abdullahi, “Towards early purchase intention prediction in online session-based retailing systems”, Electronic Markets (2021) 31:697–715 https://doi.org/10.1007/s12525-020-00448-.

[4] G. Saranya, N. Gopinath, G.Geetha, K.Meenakshi and M. Nithya, “Prediction of Customer Purchase Intention Using Linear Support Vector Machine in Digital Marketing”, ICCPET 2020 Journal of Physics: Conference Series 1712 (2020) 012024 IOP Publishing doi:10.1088/1742-6596/1712/1/012024.

[5] Sujoy Bag, Manoj Kumar Tiwari*, Felix T.S. Chan, “Predicting the consumer’s purchase intention of durable goods: An attribute-level analysis”, Article in Journal of Business Research · January 2019 DOI: 10.1016/j.jbusres.2017.11.031.

[6] Paulo Rita, Tiago Oliveira, Almira Farisa, “The impact of e-service quality and customer satisfaction on customer behavior in online shopping” Heliyon 5 (2019) e02690.

[7] José Martinsa,, Catarina Costab, Tiago Oliveirab, Ramiro Gonçalvesa, Frederico Brancoa, “How smartphone advertising influences consumers' purchase intention”, https://www.elsevier.com/locate/jbusre.

[8] Sorim Chung, University of California, Riverside, U.S.A, “Do Touch Screen Users Feel More Engaged?The Impact of Touch Interfaces on Online Shopping”.

[9] S. Ahsain, M. AitKbir, Data Mining and Machine Learning Techniques Applied to Digital Marketing Domain Needs, in Proceedings of the Third International Conference on Smart City Applications SCA, Innovations in Smart Cities Applications 4 (2021).

[10] CO. Sakar, SO. Polat, M. Katircioglu, Y. Kastro, Real-time prediction of online shoppers’ purchasing intention using multilayer perceptron and LSTM recurrent neural net-works, in Neural Computing and Applications, 31, 6893 (2019).

[11] I-Y. Song, K-Y. Whang, T. Korea, “Database Design for Real-World E-Commerce Systems”, in TCDE of the IEEE Computer Society 23, 1 (2000).

[12] M. Mud, R. Iswari, M. Ahsan M, Prediction of Online Shopper’s Purchasing Intention Using Binary Logistic Regression, Decision Tree, and Random Forest, in AIAI no. 583, 43 (2020) .

[13] W. Richert, L. P. Coelho, “Building Machine Learning Systems with Python”, Packt Publishing Ltd., ISBN 978-1-78216-140-0 .

[14] J. M. Keller, M. R. Gray, J. A. Givens Jr., “A Fuzzy K-Nearest Neighbor Algorithm”, IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-15, No. 4, August 1985 .

[15] S. Marsland, Machine learning: an algorithmic perspective. CRC press, 2015.

[16] M. Bkassiny, Y. Li, and S. K. Jayaweera, “A survey on machine learning techniques in cognitive radios,” IEEE Communications Surveys and Tutorials, vol. 15, no. 3, pp. 1136–1159, Oct. 2012.

[17] R. S. Sutton, “Introduction: The Challenge of Reinforcement Learning”, Machine Learning, 8, Page 225-227, Kluwer Academic Publishers, Boston, 1992 .

[18] P. Harrington, “Machine Learning in action”, Manning Publications Co., Shelter Island, New York, 2012.

[19] Ahmed, R.R., Parmar, V., Amin, M.A. (2014). Impact of product packaging on consumer’s buying behavior. European Journal of Scientific Research202X European Journal of Scientific Research,120(2), 1450–216.

[20] Batista, G.E.A.P.A., Prati, R.C., Monard, M.C. (2004). A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD Explorations Newsletter, 6(1), 20–29.

[21] Baumann, A., Haupt, J., Gebert, F., Lessmann, S. (2018).Changing perspectives: Using graph metrics to predict purchase probabilities. Expert Systems with Applications, 94, 137–148.

* * * * *