Journal of Informatics and Web Engineering
Vol. 4 No. 3 (October 2025) eISSN: 2821-370X

User Behaviour Prediction in E-Commerce Using Logistic Regression

Wei-Wen Lee[1], Noramiza Hashim[2], Shaymaa Al-Juboori[3*]
[1,2]Faculty of Computing and Informatics, Multimedia University, Jalan Multimedia, 63100 Cyberjaya, Malaysia
[3]School of Computing, Engineering and Mathematics, University of Plymouth, Drake Circus, Plymouth PL4 8AA, United Kingdom
**corresponding author: (shaymaa.al-juboori@plymouth.ac.uk; ORCiD: 0000-0001-5175-736X)*

Abstract - From a psychological perspective, human behaviour reflects underlying thoughts and decision-making patterns, for example, consumer behaviour may correlate with the purchase decisions. In the fast-evolving e-commerce industry, predicting user behaviour is essential for enhancing marketing strategies, improving customer experiences, and increasing sales. However, traditional heuristic (e.g. market basket analysis) approaches to analyse buyer behaviour are often rigid and fail to adapt to complex consumer interactions. This research work develops a predictive model that analyses user behaviour based on data such as historical purchasing patterns and demographic attributes. Based on a review of previous studies, Logistic Regression (LR) is utilized as the primary machine learning algorithm to estimate the likelihood of user performing specific actions including churning and conversion rate. The dataset undergoes preprocessing steps, including data cleaning, feature selection, and normalization, to enhance model accuracy. Evaluation metrics, including accuracy, confusion matrix, precision, recall and F1-Score are used to ensure the model's performance is reliable and effective. Unlike traditional heuristic approaches, this data-driven method offers a scalable and adaptable solution for behaviour prediction. The findings of this research have the potential to revolutionize e-commerce by providing businesses with actionable insights into consumer behaviour. By leveraging predictive analytics, companies can implement targeted marketing campaigns, personalize recommendations, and improve customer retention strategies. Additionally, this study highlights the significance of behavioural modelling in detecting early signs of customer churn, allowing businesses to take proactive measures. Ultimately, this research contributes to the growing field of data-driven decision-making, offering a scalable and adaptable solution for understanding and predicting user behaviour in online shopping environments.

Keywords—Logistic Regression, Machine Learning, User Behaviour Prediction, E-Commerce, Predictive Analytics.

Received: 13 Mar 2025; Accepted: 22 June 2025; Published: 16 October 2025

1. INTRODUCTION

In the rapidly changing online retail, understanding user behaviour is crucial for business to improve customer experience, optimize marketing strategies, and maximize revenue. In fact, customer's personality significantly shapes their choices, interests, and overall consumer behaviour[1]. User behaviour analysis provided many benefits on

predicting sales and recommendations [2], [3], which is essential in market targeting and reaching high-potential customers for business [4]. For example, by observing user behaviour, such as repurchasing or returning product, businesses can gain insights into customer preferences, to evaluate whether a product is in high demand or hoarding. It is beneficial to optimize company inventory e-commerce, avoiding overstocking or understocking, and ensuring that popular items are always available when required. However, with millions of users interacting with online platforms daily, several challenges have come due to the vast amount of information available. This overwhelming influx of information makes it difficult to predict and understand user behaviour accurately. To address this, machine learning (ML) techniques have emerged as a vital tool for behaviour prediction.

In various research, ML proved their ability in predictive analytics, enabling businesses to automate analyses a huge amount of user data, making data-driven decisions with higher accuracy [5], [6]. As the clickstream data captures a user's online actions, reflecting their behaviour, engagement and interaction, it allows ML to derive meaningful insight and contribute that advance personalized marketing, user experience optimization, and strategic decision-making. Among various of ML models, LR is popular in classification assignment due to the simplicity, interpretability, and effectiveness in handling binary outcomes. In the reason of the ability and effectiveness of binary classification, LR is widely used for predicting e-commerce customer behaviour [7]. For instance, churn prediction which are classified in either churn (class 1) or not (class 0). Additionally, another advantage of using LR is this technique not only simply forecasting the result between two cases, but it also provides the probability score of the prediction [8]. The probability score is a continuous value that indicates how likely a specific event occurs. For example, the churn rate and the product returning possibility.

This paper aims to delving deeply into the difference between several ML, particularly LR as the primary ML model due to its interpretability, stable performance, and suitability for binary classification tasks. The primary goal is to assess the performance of the ML model in estimating the purchase behaviour, seeking the correlation of the user behaviour influences the business and e-commerce. LR is applied to predict e-commerce related attributes, including conversion rate and churn probability based on customer browsing behaviour and biographic. The experiment is supported by two different datasets sourced from Kaggle (i.e. "Synthetic e-commerce dataset" and "E-commerce Customer Churn dataset" with appropriate adjustment, such as feature selection and feature scaling are applied. Finally, the result performance is evaluated using key metrics like accuracy, precision, recall, F1-Score, and Area Under Curve (AUC) Charts.

The paper is segmented into four sections, the previous work referred is explained in Section 2 while the Section 3 offered the details of the research methodology. Moreover, the findings and result analysis are recorded in Section 4 and lastly, Section 5 concludes the whole research and suggest the future work of the research.

2. LITERATURE REVIEW

There is a vast body of related work on user behaviour prediction, with various approaches and perspectives aimed at addressing the challenges. The research findings describe that ML like LR, Random Forest (RF), Decision Tree (DT) and others were frequently used in forecasting classification problem, for example, the user behaviour prediction in digital business.

From [9], the researchers decided to build a fusion model, which can handle different dataset types or values, by combining four different single models. The models conducted include DT, LR, Extreme Gradient Boosting (XGBoost), and Support Vector Machine (SVM). The researchers suggested that each model is responsible for different tasks. The DT with C4.5 algorithm was used to predict user behaviour in e-commerce as its ability to process continuous and discrete attribute data, even when their dataset facing the null value issue. XGBoost, with a powerful algorithm combined DT and gradient lifting algorithm, was exploited to predict the commodity purchases behaviour. The e-commerce sales forecasting was completed by using LR and SVM models. Overall, the models' performances were outstanding, particularly for LR model achieved a testing accuracy over 0.9.

[10] proposed a hybrid model built with LR and SVM models, suggesting handling probabilistic classification and maximizing the classification margins. Throughout the experiment, LR was primarily used to estimate purchase behaviour, either occurrence or non-occurrence, while the SVM model maximized the separation margin between classes. Additionally, they chose soft voting techniques as the fusion method in the experiment. Their research was based on a real-world dataset sourced from Alibaba's mobile e-commerce platform. Apparently, the hybrid model achieved better results compared to single model. Similarly, a hybrid model combining XGBoost and LR, fused by

Rank_avg method was built in another article [11]. Rank_avg was introduced as an observation tool on how models rank individual samples relative to others instead of only focusing on exact probability values. Facing the imbalance data distribution, they exploited K-means Clustering to the negative samples. The dataset applied was derived from China's biggest e-commerce platform, Taobao, owned by Alibaba's Group. As a result, the ROC curve of the fusion model is very close to the top-left corner, indicating superior performance in user behaviour prediction.

If a specific word or phrase in an article frequently but rarely appears in other article, then it can be considered as able to perform a good classification. This is the core idea of Term Frequency-Inverse Document Frequency (TF-IDF) [12]. In their paper, they decided to fuse the LR model with a text processing technique, called TF-IDF. They processed Chinese text data using the Jieba tool for segmentation and filtered out stop words. Then, TF-IDF was used to extract important keywords for classification. The output from TF-IDF was vectorized and combined with other user and product features, like demographics and product details, to create the input for LR. Cosine similarity was used to improve feature representation. The model achieved an outperformed accuracy rate of 98% based on the confusion matrix, indicating it is an effective and computationally efficient method for analysing both textual and numerical data, especially for smaller datasets.

[13] aimed to improve the prediction of e-commerce repurchase behaviour by using a hybrid ML model that combined multiple algorithms through soft voting and stacking techniques. To address the issue of dataset imbalance (where positive samples only accounted for 6%), under-sampling was applied to balance the dataset to a 1:1 ratio. Furthermore, the model developed used features from three domains: user features, merchant features, and user-merchant features. Soft voting was employed to combine the predictions from LR, KNN, XGBoost, and RF models. Additionally, a two-layer model was created, with KNN, RF, and XGBoost as base models, and LR as the meta-model. Stacking was used to further enhance the fusion of these models. The results showed that weighted soft voting achieved the highest AUC of 0.6681. To enhance e-commerce applications by capturing detailed used interaction data, [14] developed event listeners with React and Node.js to capture user interaction such as product views, add to cart, and add to wish list. The dataset used was simulated event data from a dummy e-commerce application, with around 800 raw events processed into 300 records for prediction analysis. Collected data was then exploited to predict purchasing patterns by using LR, Naïve Bayes (NB), DT and RF, with RF achieving the most outstanding performance with 98.68% accuracy and 0.98 for both precision and recall.

With millions of users interacting with online platforms daily, information overload becomes a frequent facing problem by each user. Hence, [15] presented an enhanced recommendation algorithm to resolve the challenges, by accurately introduce the recommendations product to the user. In the paper, LR served as the baseline model. It began by engineering user-side features to construct a comprehensive user profile, followed by incorporating product-side features. In addition, cross-features were suggested through an advanced LR model which successfully improve the predictive accuracy. Throughout evaluation metrics like F1-Score, the result showed that Advanced LR model, with added location information and cross feature, outperformed RF and Gradient Boosting Decision Tree (GBDT) models. Moreover, the recent study [16] integrated the Stimulus-Organism-Response (SOR) model and LR model to analyse how negative review and perceived risk affect consumer behaviour and online shopping security. The SOR model helped explore how external factor like negative reviews influence internal states and behavioural outcomes. To analysis the correlation between evaluation and external factors, the study used Data Envelopment Analysis (DEA) model. This approach measured individual efficiencies, identifying key factors such as product pricing and review characteristics that contribute to a secure and favourable shopping experience. The relative efficiencies of various attributes were calculated using a mathematical programming method. LR model was then used to quantify the relationships between the independent variables and dependent variables. The dataset was sourced from Kaggle which included variables such as order details and demographics. In the end of the paper, the author revealed that a causal relationship did exist between perceive risk and consumer negative behaviour based on the result observed.

In addition, [8]'s objective was predicting product return intentions, based on the customer behaviour patterns. The research proposed customer segmentation to identify factors influencing product returns, predict the likelihood of returns, and segment customers into high-risk and low-risk groups. The researchers highlighted the flow of the experiment, start with data preprocessing, data analysis and understanding, model training and evaluation, followed by customer segmentation. In the phase three, the model training and evaluation, seven selected features models, including Naïve Bayes, RF, LR, XGBoost, AdaBoost and K-Nearest Neighbours (KNN) were developed to predict online product returns. The customer segmentation model utilized LR algorithm and classification threshold to segment the customer into groups, effectively reducing the product return rate. As a result, XGBoost performed best among the others, with the accuracy of 76.9%, followed by other models such as LR, Naïve Bayes and AdaBoost

reached the accuracy of 75.6% ,76.7%, and 76.7% respectively. As LR is a strong model in binary class prediction, making it a wise choice for predicting the likelihood of customer churn. [17] introduced the concept of a retention rate as a key indicator of churn likelihood. The retention rate is a continuous value within the range of [0,1], where a lower value signals a higher probability of churn. Using LR, they calculated this retention rate and integrated it into the development of an Electronic Business User Retention Model (EBURM). As a result, a big success accomplished with the model achieved the total predictive accuracy of 93.6%. With similar objective and exploiting same dataset, [18] decided to predict the customer loss with both LR and Gradient Boosting (GB) Models. The authors concluded although GB achieved a better result with 91% of testing accuracy, the performance of LR was also satisfied with the 88% of testing accuracy.

Apart from customer churn prediction, [19] employed LR analysis to explore the purchase decisions. The model was applied to examine how various consumer features, such as customer reviews and ratings, product recommendations, and other relevant factors, influence the purchasing choices. In their study, they introduced several evaluation metrics, including the Omnibus Test, Hosmer and Lemeshow Test, and -2 Log likelihood, among others. The results of these evaluations demonstrated that the LR model was a well-suited fit for the data, showcasing its strong predictive capability in forecasting purchase decisions. The literature on e-commerce adoption[20] highlights various demographic, economic, and technological factors influencing online shopping behaviour, particularly in developing regions. Prior research indicates that younger, higher-educated, and higher-income individuals are more likely to engage in e-commerce, while gender differences persist, with men using online shopping more frequently than women. Digital engagement, including social media usage, online product searches, and internet banking, significantly enhances e-commerce adoption. To empirically validate these relationships, a study applied binary LR using microdata from the Turkish Statistical Institute's Household Information Technologies Survey. The results confirmed that income level, age, education, gender, social media activity, online sales, and financial technology usage were significant predictors of e-commerce adoption. The likelihood of using e-commerce increased with higher income (₺6001 and above) and digital engagement, whereas larger household sizes negatively impacted online shopping.

One of the findings cited from [21] suggested that LR can be an effective tool for both feature extraction and classification purpose. From their research project, LR was first applied to extract the most relevant features from the dataset, ensuring only the features with predictive values were explored for classification. Following feature selection, different ML classifiers, including Linear Support Vector Machine (LSVM), DT, LR and NB were developed and compared to determine the best-performing model. LSVM outperformed with a 96% accuracy rate, while LR was 2% lower to LSVM. Other than that, the [22] study employed binary LR in analysing factors influencing individuals' likelihood of adopting e-commerce. They utilized nationwide survey data covering all 34 provinces of Indonesia and includes 8854 usable samples. LR is used to model the probability of an individual being an e-commerce adopter based on demographic factors, access to digital infrastructure, digital skills, and exposure to harmful content. The results show that individuals with the characteristics of younger, more educated, male, married, and entrepreneurial are more likely to adopt e-commerce in Indonesia. Additionally, access to financial services, mobile internet, and logistics services positively impacts e-commerce adoption, while exposure to harmful online content acts as a deterrent. The study concludes that improving digital literacy, expanding infrastructure, and mitigating perceived risks can enhance e-commerce participation in Indonesia.Moreover, a similar related works from Turkey also exploited binary LR to analyse factors influencing e-commerce adoption across different education levels [23]. The research used microdata from Turkey's Household Information Technologies Usage Survey (2019) and applies LR models separately for individuals with elementary education, high school education, and university degrees. Like the previous paper, the results proved again that higher income, younger age, male gender are more active involving in the digital business. Other features such as social media usage, internet banking, and e-government services positively influence e-commerce adoption. Additionally, individuals in western Turkey are more likely to engage in e-commerce than those in the eastern regions. [24] explored various of ML techniques for sentiment analysis in e-commerce. The researchers applied LR, SVM, NB, and Neural Networks (NN) to classify customer sentiments based on online reviews. Latent Semantic Analysis (LSA) is used for feature extraction, identifying the most frequent words in customer reviews before applying classification algorithms. Among the models tested, LR outperformed all other classifiers, achieving the highest accuracy (91%) and AUC score (0.96). The study highlighted the effectiveness of LR for text-based sentiment analysis, demonstrating its suitability for analysing customer opinions and predicting purchasing behaviour in e-commerce platforms.

Nevertheless, from the study of [25], LR and the Auto-Regressive Moving Average (ARMA) model were used to analyse and predict customer review trends on Amazon. The research focused on quantifying product ratings and text reviews for three products, including laptop, air conditioner, and lamp, and examined how reviews evolve over time.

The LR Equation Model is used to fit a function that models the relationship between review positivity and time, helping to track shifts in customer sentiment. However, recognizing that real-world data is affected by outliers and random fluctuations, the study introduced the ARMA model to smooth predictions and reduce noise. The results indicated that the reputation of the air conditioner and lamp is expected to improve, whereas the laptop's reputation is projected to decline. In the conclusion, the model implemented, Logistic-ARMA effectively enhanced product evaluation trends, offering valuable insights for e-commerce platforms. [16] utilized LR as the primary model to understand and predict consumer behaviour in online shopping, with a particular focus on its relationship with network security. The model is used to quantify the influence of various factors (e.g. negative reviews, perceived risk, delivery speed, etc.) on consumer decision-making. By analysing the purchase decisions, LR helped identify behavioural patterns that contribute to online shopping security risks. The results indicated that factors like perceived risk and negative reviews significantly affect purchasing decisions, leading to delayed consumption, refusal, or opposition to buying. From the research, strengthening consumer perception and trust were proved that able to enhance online shopping security, ultimately reducing fraudulent transactions and improving e-commerce platform safety.

[26] applied LR, NB, SVM, RF and AdaBoost to analyse consumer sentiment in fashion e-commerce. The research utilizes a dataset of 23,485 customer reviews from various e-commerce platforms such as Amazon and Flipkart, focusing on the relationship between ratings, reviews, and product recommendations. In the study, the key findings included LR achieved the highest accuracy of 88.18% for customer reviews sentiment classification and performed best in rating-based sentiment analysis, achieving 80.68% accuracy. Hyperparameter tuning techniques like Grid Search and Random Search were applied to improve model performance. Sentiment analysis using VADER and TextBlob was also conducted, with VADER achieving 76% accuracy and TextBlob 77% accuracy. The study concludes that LR is the most effective model for predicting sentiment from e-commerce reviews, helping businesses understand the consumer behaviour. Lastly, this section is wrapped up by the related work of [27]. In the study, ML techniques, LR alongside with other models were applied to analyse consumer sentiments from Bangla product reviews. The research focuses on classifying sentiments as positive or negative using a dataset collected from Bangladeshi e-commerce platforms and processed with TF-IDF and Trigram features. LR was employed for sentiment classification, alongside other models such as Multinomial Naïve Bayes (MNB), SVM, DT, RF, and Stochastic Gradient Descent (SGD). As a result, LR achieved an accuracy of 87.25%, with strong performance in recall (93.63%) and F1-Score (88.02%). SVM demonstrated superior performance compared to the other classifiers. with the highest accuracy of 90.68% after hyperparameter tuning using RandomizedSearchCV. From their paper, the ML models, particularly SVM and LR, are effective for sentiment analysis in Bangla e-commerce reviews, helping businesses understand consumer opinions and improve services.

Despite extensive prior work exploring research on user behaviour prediction in e-commerce, the reviewed literature demonstrates that LR is a widely adopted and effective model for user behaviour prediction, particularly in binary classification tasks such as purchase intention, churn prediction, e-commerce adoption, and sentiment analysis. Across numerous studies, LR has shown stable and competitive performance, often achieving high accuracy, AUC scores, and interpretability even when compared to more complex models like XGBoost, SVM, or ensemble methods. Its effectiveness is further enhanced when integrated with feature engineering techniques (e.g., TF-IDF, cross-features, clustering) or used as part of hybrid frameworks (e.g., soft voting, stacking). LR's simplicity, generalization ability, and low computational cost make it especially suitable for real-world applications with structured or text-based data. Importantly, despite the popularity of complex models, LR consistently serves either as a strong standalone baseline or as a reliable model in fusion model.

However, there remain significant gaps specifically concerning the role and potential of LR. Firstly, although LR has repeatedly demonstrated strong performance in various e-commerce tasks—ranging from purchase intention and churn prediction to sentiment analysis and e-commerce adoption—it is often embedded within hybrid or ensemble models or used as a baseline. While this confirms its reliability, it also suggests a lack of focused exploration on how LR alone can be optimized and generalized for complex e-commerce environments, particularly under operational constraints.

Moreover, another notable gap in the existing literature is the relatively limited scope of feature usage in many prior studies. While several works have demonstrated strong performance using selected user or product-level features, they often rely on a narrow set of attributes. However, real-world e-commerce environments are inherently complex, involving a much broader data such as interaction sequences (e.g., product views, wish lists), session information, clickstream paths, and promotional exposure. The existing of the features raises questions about LR's generalizability and practical robustness.

Additionally, while many studies have validated LR on large, proprietary datasets (e.g., Alibaba, Taobao), there is a gap in evaluating LR's practicality on smaller or simulated datasets, especially for start-ups or low-resource platforms that may not have access to massive data. Exploring LR in such scenarios will help expand its applicability to a broader range of e-commerce businesses.

Therefore, this study selected LR as the primary ML model not only due to its proven reliability and suitability for binary classification, but also to explore its potential in robustness, interpretability, and adaptability under e-commerce constraints. This includes an emphasis on performance in low-resource settings, model transparency in engineered feature spaces, and ethical implications in behavioural prediction.

3. RESEARCH METHODOLOGY

3.1 Datasets

While implementing the model, there are two sets of data used to predict different types of user behaviour in e-commerce. One of the prototype's model training datasets is a synthetic e-commerce dataset sourced from [28], which is a comprehensive collection that includes transaction, customer, product, and advertising data from a dynamic marketplace. This dataset simulates real-world scenarios, incorporating seasonal effects, regional variations, advertising metrics, and customer purchasing behaviours. Additionally, the dataset contains 14 attribute columns. Table 1 provides a detailed description of the attributes in this dataset.

Table 1. Attributes Description of Dataset 1

Attributed Name	Details	Field Type
Transaction_ID	Unique identifier for each transaction	String
Customer_ID	Unique identifier for each customer	String
Product_ID	Unique identifier for each Product	String
Transaction_Date	The date when the transaction took place	Date
Category	The product category (e.g. Electronics, Clothing)	String
Unit_Sold	The product sold quantity based on the transaction	Integer
Discount_Applied	Percentage discount applied on the product	Float
Revenue	Total earnings from the transaction: Price × Units Sold × (1 - Discount)	Float
Clicks	Number of times the product ad was clicked during the transaction period	Integer
Impression	Number of times the product ad was displayed	Float
Conversion_Rate	Calculated as Clicks / Impressions, representing the ratio of clicks to impressions	Float
Region	The geographical region where the transaction occurred	String
Ad_CTR	Click-through rate (CTR) for the advertisement, representing the effectiveness of the ad campaign.	Float
Ad_CPC	Cost-per-click for the advertisement.	Float
Ad_Spend	Total advertising spends for the product, calculated as Ad_CTR x Ad_CPC x 1000.	Float

The second dataset focuses on predicting churn probability in the e-commerce sector. It is derived from the "E-commerce Customer Churn Dataset" available on Kaggle, sourced from a prominent online e-commerce company.

The dataset comprises 11 columns, including 10 feature variables that capture customer behaviours and characteristics, along with a target variable indicating churn probability. A detailed explanation of these columns is provided in Table 2.

Table 2. Attributes Description of Dataset 2

Attributed Name	Details	Field Type
Tenure	Tenure of a specific customer in the company	Integer
WarehouseToHome	The Distance between customer's home and the warehouse	Integer
NumberOfDeviceRegistered	Total number of devices registered by the customer	Integer
PreferedOrderCat	Customer's most frequently chosen product category in the last month	String
SatisfactionScore	Satisfaction level of customer on service	Integer
MaritalStatus	Marital Status Classification of a customer	String
NumberOfAddress	Number of unique addresses saved by the customer	Integer
Complaint	Indicates if a complaint was made in the past month (1 = Yes, 0 = No)	Integer/ Binary
DaySinceLastOrder	Day since last order by Customer	Integer
CashbackAmount	Average cashback received by the customer in the last month	Integer
Churn	Churn flag (1 = Yes, 0 = No)	Integer/ Binary

3.2 Data Usability

From the datasets that chose to use, two target attributes—conversion rate and customer churn—are serve as the dependent variables in our LR models. The conversion rate refers to the proportion of customer impressions that result in clicks, providing insight into the effectiveness of marketing strategies, advertisements, and campaigns. A high conversion rate indicates successful engagement, which can lead to cost-effective customer acquisition.

On the other hand, customer churn represents the likelihood that a customer will stop engaging with or purchasing from a company. Predicting churn is crucial, as it enables businesses to proactively identify at-risk customers. Instead of implementing broad and potentially misaligned strategies, churn prediction allows companies to tailor interventions, understand the underlying reasons for customer loss, and develop more effective retention strategies. Both user behaviours are key performance indicators for e-commerce platforms and play a vital role in enhancing business decision-making and competitiveness.

Apart from these two attributes, several other variables are well-suited to serve as dependent variables due to their predictive value, specifically, those related to customer demographics, product descriptions, and consumer behaviour. For instance, attributes such as product category, units sold, and whether a discount was applied can offer valuable insights into conversion rates and help businesses understand, learn, and apply their data to practical use cases. Similarly, features like customer tenure, preferred order category, and marital status are useful for analysing customer preferences, behaviours, and shopping patterns. Given these factors, using these two datasets for our experimental analysis is both reasonable and justified.

3.3 Data Cleaning

The initial step involves data cleaning before the dataset can be processed for model training. This crucial stage utilizes an efficient pipeline to prepare and refine the data for the upcoming model training process.

- Step 1: Remove Useless Data
    - To avoid unnecessary complexity or overfitting, the columns which do not have predictive value are being removed.

- In the first dataset, columns including 'Transaction_ID', 'Customer_ID', 'Product_ID', 'Transaction_Date', 'Ad_CPC', and 'Ad_Spend' are removed using .drop() function.
- For the second dataset, no columns need to be dropped, as all columns represent valuable features that contribute meaningfully to the analysis.

- Step 2: Handling Missing Data
  - Use the .info() to identify whether the null value existed.
  - No missing or null values in the first dataset indicate the dataset is well structured. Figure 1 shows the columns details of the Dataset 1.

```
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   Units_Sold        100000 non-null  int64
 1   Discount_Applied  100000 non-null  float64
 2   Revenue           100000 non-null  float64
 3   Clicks            100000 non-null  int64
 4   Impressions       100000 non-null  int64
 5   Conversion_Rate   100000 non-null  float64
 6   Category          100000 non-null  object
 7   Region            100000 non-null  object
 8   Ad_CTR            100000 non-null  float64
dtypes: float64(4), int64(3), object(2)
memory usage: 6.9+ MB
None
```

Figure 1. Column Distribution of Dataset 1

- Figure 2 tells that there are a few missing data appeared in several columns of the Dataset 2, including Tenure, WarehouseToHome, and DaySinceLastOrder. To address this issue, the missing values in numerical attributes are imputed with the mean of the respective column, while missing values in categorical attributes are filled with the mode (most frequent value) of the column. This approach helps ensure that the dataset remains complete for analysis without introducing bias from missing data.

```
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Tenure                    3747 non-null   float64
 1   WarehouseToHome           3772 non-null   float64
 2   NumberOfDeviceRegistered  3941 non-null   int64
 3   PreferedOrderCat          3941 non-null   object
 4   SatisfactionScore         3941 non-null   int64
 5   MaritalStatus             3941 non-null   object
 6   NumberOfAddress           3941 non-null   int64
 7   Complain                  3941 non-null   int64
 8   DaySinceLastOrder         3728 non-null   float64
 9   CashbackAmount            3941 non-null   float64
 10  Churn                     3941 non-null   int64
dtypes: float64(4), int64(5), object(2)
```

Figure 2. Column Distributions of Dataset 2

Figure 3 shows the dataset info after filling the null.

```
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Tenure                   3941 non-null   float64
 1   WarehouseToHome          3941 non-null   float64
 2   NumberOfDeviceRegistered 3941 non-null   int64
 3   PreferedOrderCat         3941 non-null   object
 4   SatisfactionScore        3941 non-null   int64
 5   MaritalStatus            3941 non-null   object
 6   NumberOfAddress          3941 non-null   int64
 7   Complain                 3941 non-null   int64
 8   DaySinceLastOrder        3941 non-null   float64
 9   CashbackAmount           3941 non-null   float64
 10  Churn                    3941 non-null   int64
dtypes: float64(4), int64(5), object(2)
```

Figure 3. Column Distributions of Dataset 2 After Filling Null

- Step 3: Removed Duplicated Data
  - After dropping the duplicate data, the number of the original data and the dataset remain the same, indicating no redundant data in the Dataset 1.
  - Figure 4 proves no duplicated data in the Dataset 1 after checking.

```
Duplicate Rows:
0
Rows number Before removing duplicates:
100000
Rows number After removing duplicates:
100000
```

Figure 4. Output of Removing Duplicated Data on Dataset 1

  - However, in Dataset 2, the duplicate rows problem occurred. Figure 5 shows that there are 671 duplicate rows being removed from the dataset. As a result, 3270 rows of the records left in the dataset.

```
Duplicate Rows:
671
Rows number Before removing duplicates:
3941
Rows number After removing duplicates:
3270
```

Figure 5. Output of Removing Duplicated Data in Dataset 2

3.4 Data Preprocessing

After ensuring the data is clean, the dataset still requires preprocessing to be suitable for the model, as LR is designed to handle categorical data. The data must also be split into training and test sets to ensure proper model evaluation and performance. The details of this phase are shown below:

- Step 1: Convert Columns Data Type to Binary
  - In the first dataset, a threshold is applied to the target variable, Conversion_Rate, to categorize the values into two classes. Specifically, values greater than or equal to 0.2, which corresponds to the mean conversion rate across the dataset, are assigned a label of 1, indicating a higher

conversion rate. Conversely, values below 0.2 are assigned a label of 0, indicating a lower conversion rate. The choice of using the mean as the threshold ensures that the classes are more balanced, as it divides the data around the average conversion rate. The function **get_dummies()** converts the multiclass data, Category into binary class format.

- Figure 6 shows the details of each column of Dataset 1 after pre-processed.

```
Data columns (total 13 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   Units_Sold                100000 non-null  int64
 1   Discount_Applied          100000 non-null  float64
 2   Revenue                   100000 non-null  float64
 3   Clicks                    100000 non-null  int64
 4   Impressions               100000 non-null  int64
 5   Conversion_Rate           100000 non-null  int64
 6   Ad_CTR                    100000 non-null  float64
 7   Category_Clothing         100000 non-null  bool
 8   Category_Electronics      100000 non-null  bool
 9   Category_Home Appliances  100000 non-null  bool
 10  Category_Toys             100000 non-null  bool
 11  Region_Europe             100000 non-null  bool
 12  Region_North America      100000 non-null  bool
dtypes: bool(6), float64(3), int64(4)
memory usage: 5.9 MB
None
```

Figure 6. Columns Distribution of Dataset1 After Preprocessing

- In Dataset 2, the target variable, Churn, is already in binary form, so no further preprocessing is required for this variable. However, for the SatisfactionScore, a threshold is applied where values greater than or equal to 3 are classified as class 1, indicating higher satisfaction, or else, values are classified as class 0, representing lower satisfaction.
- Features such as PreferOrderCat and MaritalStatus are classified into several binary columns by **get_dummies() function.**
- Figure 7 shows the details of each column of Dataset 2 after pre-processed.

```
Data columns (total 16 columns):
 #   Column                              Non-Null Count  Dtype
---  ------                              --------------  -----
 0   Tenure                              3941 non-null   float64
 1   WarehouseToHome                     3941 non-null   float64
 2   NumberOfDeviceRegistered            3941 non-null   int64
 3   SatisfactionScore                   3941 non-null   int32
 4   NumberOfAddress                     3941 non-null   int64
 5   Complain                            3941 non-null   int64
 6   DaySinceLastOrder                   3941 non-null   float64
 7   CashbackAmount                      3941 non-null   float64
 8   Churn                               3941 non-null   int64
 9   PreferedOrderCat_Grocery            3941 non-null   bool
 10  PreferedOrderCat_Laptop & Accessory 3941 non-null   bool
 11  PreferedOrderCat_Mobile             3941 non-null   bool
 12  PreferedOrderCat_Mobile Phone       3941 non-null   bool
 13  PreferedOrderCat_Others             3941 non-null   bool
 14  MaritalStatus_Married               3941 non-null   bool
 15  MaritalStatus_Single                3941 non-null   bool
dtypes: bool(7), float64(4), int32(1), int64(4)
```

Figure 7. Columns Distribution of Dataset 2 After Preprocessing

- Step 2: Train-Test Split and Standardization
  - Dataset 1

    The dataset is split into two subsets using an 80:20 ratio for the training and testing sets with the train_test_split() function. The choice of this ratio is based on the size of the dataset and empirical testing. For large dataset with 100000 rows, ratios of 80:20 or 70:30 often suggested to ensure that the model can learn effectively while still being evaluated on an unseen portion of the data. Initially, a 70:30 split was tested, it resulted in lower performance metrics compared to the 80:20 split when applying *MinMaxScaler t*o normalize the numeric attributes. However, when standardizing the numeric attributes with 70:30 split of train-test set, the result performs better comparing to the previous setting. This outcome suggests that the model benefits from having standardized numeric attributes with larger test sets. Figures 8 and 9 show the row counts of the training and data sets, respectively.

Figure 8. Class Distribution of Training Set in Dataset 1

Figure 9. Class Distribution of Testing Set in Dataset 1

- Dataset 2
  With the experiment of implementing dataset 1, a hypothesis is developed. The setting of 0.3 test size splitting following with standardized numeric features may perform the best in the model. Hence, this setting remained to predict the dataset 2. As a result, the performance is satisfying. However, to ensure the hypothesis is true and still workable in a smaller size of dataset, train-test split with the different rates (i.e. 0.2 and 0.4) and Scaling using MinMaxScaler are utilized, and the result is observed. In conclusion, using 60:40 splitting ratio and scaling the numeric values with a min max scaler did slightly increase training accuracy, but the testing accuracy was dropped, indicating that the hypothesis is reliable. Figures 10 and 11 show the row counts of the training and data sets, respectively.

Figure 10. Class Distribution of Training Set in Dataset 2

Figure 11. Class Distribution of Testing Set in Dataset 2

- Step 3: Data Balancing
  - However, in the train test split outcomes in step 2, the classes are imbalanced in both dataset 1 and 2. To address this, random oversampling is applied to balance the classes. Figure 12 and Figure 13 illustrate the row counts of the training set after balancing for datasets 1 and 2, respectively.

Figure 12. Class Distribution of Balanced Training Set in Dataset 1

Figure 13. Class Distribution of Balanced Training Set in Dataset 2

3.5 Predictive Model

After balancing the classes of the training set, the dataset is almost ready to fit the LR model. However, a few final data preparation steps are required.

Initially, the training and testing datasets are concatenated into a single DataFrame. This approach ensures consistent preprocessing across both datasets. To avoid data leakage and to evaluate the model's ability to generalize to unseen data, the data is then split back into separate training and testing sets after preprocessing.

Next, the numerical feature columns in the testing data are scaled using the MinMaxScaler or StandardScaler, ensuring consistency in data ranges across both the training and testing sets. With these steps complete, the data is fully prepared for modelling.

With preprocessing complete, a LR model is initialized using scikit-learn's LogisticRegression class. The model is trained on the oversampled and pre-processed training data. Model performance is then evaluated on both the training and testing sets using the .score() method, which reports classification accuracy.

3.6 Evaluation Metrics

Assessing the model performance is one of the most important steps in model development. As a classification model, there are various evaluation metrics can be utilized depending on the specific goals of the analysis. Below are some of the key evaluation methods for LR.

3.6.1 Confusion Matrix

confusion matrix provides a detailed summary of prediction results by comparing actual and predicted values. It categorizes outcomes into four classes: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). This matrix is particularly useful for understanding model behaviour in classification tasks.

3.6.2 Accuracy

Accuracy is the ratio of the proportion of correctly predicted observations over the total observations (see Equation (1)). It can be misleading, especially in imbalanced datasets where one class is much more prevalent than the other.

$$\text{Accuracy} = (TP + TN)/(TP + FP + FP + FN) \quad (1)$$

Where:
TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative

3.6.3 Precision

Precision is the ratio of correctly predicted positive observations to the total predicted positives. In Equation (2), it measures the proportion of positive predictions that were correct.

$$\text{Precision} = TP/(TP + FP) \quad (2)$$

Where:
TP: True Positive, FP: False Positive

3.6.4 Recall

Also known as sensitivity, recall measures the ability of the model to correctly identify all relevant positive cases (see Equation (3)). It is crucial when false negatives are costly.

$$\text{Recall} = TP/(TP + FN) \quad (3)$$

Where:
TP: True Positive, FN: False Negative

3.6.5 F1-Score

The F1-Score is the hormonic average of precision and recall as depicted in Equation (4). A higher F1-score indicates that both precision and recall are more balanced, meaning the model is performing well in terms of both minimizing false positives (precision) and false negatives (recall).

$$\text{F1-Score} = 2 \cdot \big((P + R)/(P \cdot R)\big) \quad (4)$$

3.6.6 Area Under Receiver Operating Characteristic Curve (AUC-ROC)

The ROC is the curve of the true positive rate against the false positive. AUC of the ROC measures the area under this curve form range 0 to 1, with higher values indicating better performance.

3.6.7 Area Under the Precision-Recall Curve (AUC-PR)

The AUC-PR measures the area under the precision-recall curve, it plots precision against recall for different threshold values. AUC-PR is essential for evaluating model performance when the positive class is rare.

3.7 Pseudocode

The predictive model employs an advanced LR approach specifically designed to predict user behaviour with different features. This section provides a step-by-step description of the algorithm to clarify the model implementation process.

The implementation begins with loading data from a CSV file, followed by the removal of unnecessary columns. Initial exploratory data analysis (EDA) is conducted in steps 1 through 8 to better understand the dataset. In steps 9 through 14, null values and duplicate rows are identified and addressed to ensure data integrity. Numerical columns with missing values are filled with the mean, while categorical columns are imputed with the most frequent value.

Steps 15 and 16 involve a detailed description of the numerical and categorical columns to investigate their respective features. In step 17, discrete numerical columns are transformed into binary values using a threshold: values equal to or greater than 1 are classified as class 1, while other values are classified as class 0. For categorical features, step 18 utilizes one-hot encoding, such as the get_dummies function, to convert them into a Boolean format.

Algorithm 1: Prediction of User Behaviour using LR

Input: Dataset
Output: User Behaviour Prediction

1. START
2. IMPORT necessary libraries
3. LOAD dataset INTO ‘raw’
4. DISPLAY ‘raw’
5. REMOVE unnecessary columns FROM ‘raw’ INTO ‘df’
6. PRINT shape of ‘df’
7. PRINT columns of ‘df’
8. DISPLAY ‘df’ information
9. CHECK if null value exists:
   FOR numerical columns:
   FILL null by mean value
   FOR category columns:
   FILL null by mode value
   PRINT shape of ‘df’
   PRINT columns of ‘df’
   DISPLAY ‘df’ information
10. DEFINE duplicated rows in ‘df’ AS ‘duplicates’
11. PRINT number of rows of ‘duplicates’
12. DROP duplicates rows
13. DEFINE ‘df’ without duplicate rows as ‘df_no_duplicates’
14. PRINT number of rows of ‘df_no_duplicates’
15. DESCRIBE columns with data types ‘int64’ or ‘float64’
16. DESCRIBE columns with data types ‘object’
17. FOR numeric columns need to convert to binary class:
    DEFINE threshold value
    FOR value >= threshold value:
    SET value AS ‘1’

FOR value < threshold value :
SET value AS ‘0’

18. FOR object columns need to convert to binary class:
    FOR each value in the column:
    CREATE new binary column
    DROP the first multiclass column
    STORE resulting DataFrame with the dummy variables in ‘df1’
19. DISPLAY ‘df1’
20. PRINT shape of ‘df’
21. PRINT columns of ‘df’
22. DISPLAY ‘df’ information
23. DEFINE a copy of ‘df1’ AS ‘df_pp’
24. DEFINE dependent variable from ‘df_pp’ AS ‘X’
25. DEFINE independent variable from’ df_pp’ AS ‘y’
26. SPLIT X AND y into training and test sets (X_train, X_test, y_train, y_test)
27. PRINT ‘X_train’ set size
28. PRINT ‘X_test’ set size
29. PLOT counts of positive and negative classes for ‘y_train’
30. PLOT counts of positive and negative classes for ‘y_test’
31. DISPLAY count plots
32. CREATE ‘scaler’ using MinMaxScaler or StandardScaler
33. DEFINE a copy of ‘X_train’ AS ‘X_train_preprocess’
34. DEFINE columns in ‘X_train’ of data types 'int64' or 'float64' AS ‘num_attributes’
35. SCALE ‘num_attributes’ in ‘X_train_preprocess’ using ‘scaler’
36. DISPLAY ‘X_train_preprocess’
37. CREATE ‘ros’ using RandomOverSampler with random_state set to 0
38. PERFORM oversampling on ‘X_train_preprocess’ and ‘y_train’ using ‘ros’
39. CONVERT ‘X_ovr_samp’ to a DataFrame using columns from ‘X_train_preprocess’
40. PLOT counts of positive and negative classes for ‘y_train’
41. PLOT counts of positive and negative classes for ‘y_test’
42. DISPLAY count plots
43. DEFINE the number of rows in ‘X_ovr_samp’ AS ‘X_train_rows’
44. CONCATENATE ‘X_ovr_samp’ and ‘X_test’ along axis 0 into ‘X_train_test’
45. DEFINE the first ‘X_train_rows’ rows of ‘X_train_test’ AS ‘X_train_onehot’
46. DEFINE the remaining rows of ‘X_train_test’ after ‘X_train_rows’ AS ‘X_test_onehot’
47. DEFINE a copy of ‘X_test_onehot’ AS ‘X_test_preprocess’
48. TRANSFORM ‘num_attributes’ in ‘X_test_preprocess’ using scaler
49. SELECT numeric columns from ‘X_train_onehot’ into ‘X_train_numeric’
50. INITIALIZE LR model with solver='lbfgs' and max_iter=1000
51. TRAIN LR model using ‘X_train_numeric’ and ‘y_ovr_samp’
52. PRINT the training accuracy of the model
53. SELECT numeric columns from ‘X_test_preprocess’ into ‘X_test_numeric’
54. REORDER columns in ‘X_test_numeric’ to match the column order of ‘X_train_numeric’
55. PRINT the testing accuracy of the model with ‘X_test_numeric’ and ‘y_test’
56. MAKE predictions using logreg.predict with ‘X_test_numeric’
57. PRINT classification report comparing ‘y_test’ (true values) and ‘y_pred’ (predicted values)
58. GENERATE confusion matrix using ‘y_test’ (true values) and ‘y_pred’ (predicted values)
59. EXTRACT tn, fp, fn, tp from the confusion matrix
60. PRINT the confusion matrix (conf_mat)
61. PRINT True Positive (tp) number
62. PRINT True Negative (tn) number
63. PRINT False Positive (fp) number
64. PRINT False Negative (fn) number
65. GENERATE normalized confusion matrix using y_test (true values) and y_pred (predicted values) with normalization set to 'true'
66. PLOT heatmap

67. END

Steps 19 to 22 ensure that the dataset is properly pre-processed and ready for model fitting. Steps 23 to 26 outline the process of splitting the dataset into training and test sets, while steps 27 to 31 include visualizations of these sets for better understanding. Step 32 introduces a scaler to normalize numerical attributes in both the training and test datasets, with steps 33 to 36 explaining the scaling process in detail.

To address any class imbalance, steps 37 to 42 describe the application of random oversampling to balance the positive and negative classes. Steps 43 to 51 cover the setup of a LR model, including concatenating the target train set, scaling numerical attributes, and initializing the LR model. The model is then trained, and its training accuracy is printed in step 51.

After training the model, steps 53 to 55 outline the testing procedure. In step 56, predictions on the test set are generated using logreg.predict. Finally, steps 57 to 66 focus on evaluating the model's performance through a classification report, confusion matrix, and heatmap, offering a comprehensive assessment of its effectiveness.

4. RESULTS AND DISCUSSIONS

The outcome performance is evaluated through accuracy, classification report, and heat map in the initial research from developing the predictive model using LR model. Table 3 illustrates the accuracy of the prediction results for the Dataset1 and Dataset2.

Table 3. Accuracy Results of Dataset 1 and 2

Dataset	Training Accuracy	Testing Accuracy
Dataset 1	0.9906582948653438	0.986225
Dataset 2	0.8090551181102363	0.8030431107354185

The training accuracy and testing accuracy achieved by the LR model for Dataset 1 are 0.9907 and 0.9862 respectively, indicating that the model is outperformed in the conversion rate prediction. In the other hand, LR model achieved a training accuracy of 0.8091 and testing accuracy of 0.8030 for the churn prediction in Dataset 2. Although it is slightly lower than dataset 1, the result of 0.8091 and 0.8030 of training and testing accuracies still suggest that the model perform decently.

The results indicate that model has effectively learned the patterns in training data and perform a good generalization to unseen data. Additionally, the close alignment between training and testing accuracy proves that the model is not overfitting, which is a main concern in model implementation.

In Figure 14, the deeper insights into the model's performance are provided. For Class 0, the precision and recall indicate the model's strong ability to correctly identify negative cases (conversion rate is under 0.2) while minimizing false positives. For Class 1, the recall of 1.00 shows that the model successfully captures all positive cases, but slightly lower precision, 0.95 suggests there are some negative cases predicted wrongly as positive. In addition, the macro average (F1-Score = 0.98) and weighted average (F1-Score = 0.99) prove again overall strong performance across both classes.

	precision	recall	f1-score	support
0	1.00	0.98	0.99	30472
1	0.95	1.00	0.97	9528
accuracy			0.99	40000
macro avg	0.97	0.99	0.98	40000
weighted avg	0.99	0.99	0.99	40000

Figure 14. Classification Report for Dataset 1

In Figure 15, the precision and recall in class 0 indicate the model's strong ability to correctly identify negative cases while minimizing false positives. For Class 1, the recall of 0,79 shows that the model successfully captures many true positives, but the low precision with only 0.46 suggests that false positive cases happened constantly.

	precision	recall	f1-score	support
0	0.95	0.81	0.87	981
1	0.46	0.79	0.58	202
accuracy			0.80	1183
macro avg	0.70	0.80	0.73	1183
weighted avg	0.87	0.80	0.82	1183

Figure 15. Classification Report for Dataset 2

The macro-average F1-Score of 0.7 suggests that the model performs well across both classes. The model may slightly favour class 0, but it is acceptable since the churn prediction require to focus much on negative class prediction. Moreover, a weighted-average F1-Score of 0.8 tells that the model is performing better on the more frequent classes. The difference between these two scores implies that the model perform slightly imbalance across classes, with stronger results for class 0 with a larger number of supports.

In Figure 16, the actual number of cases in each class is displayed, and the heatmap illustrates the confusion matrix's rates, where darker colours represent lower values. The model shows a perfect true positive class with a support rate of 1.0 (7146 cases), indicating that all positive cases are predicted correctly, with zero false positives. However, the model exhibits a slight deviation in identifying negative cases. The true negative rate is 0.98, with a false negative rate of 0.018. The specific support numbers for true negatives and false negatives are 22,677 and 7,146, respectively.

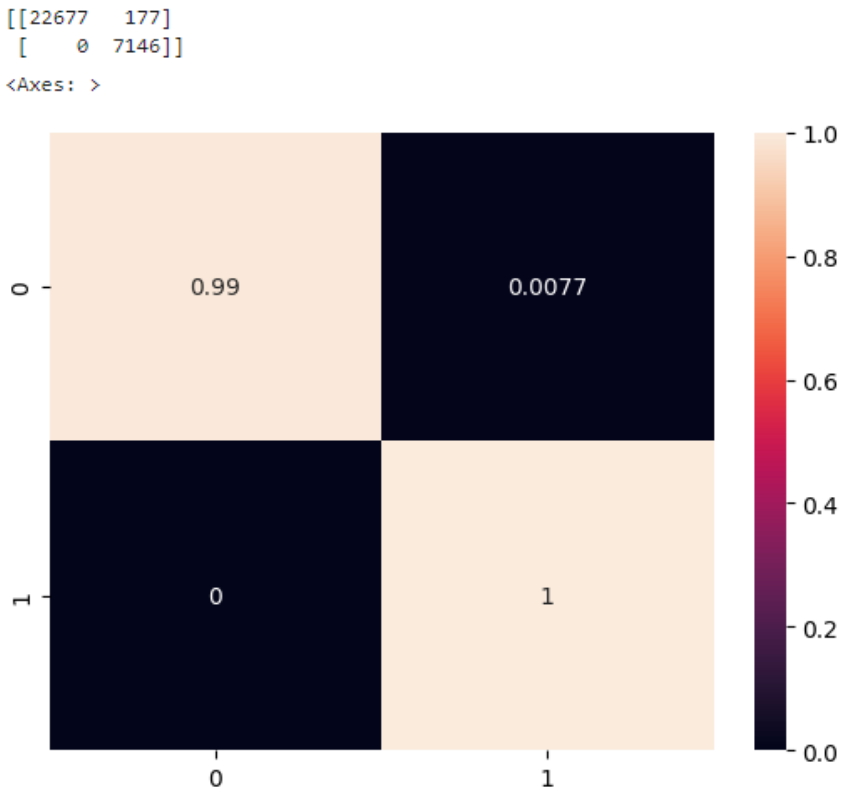

Figure 16. Confusion Matrix and Heatmap for Dataset 1

Based on Figure 17, the rates of 0.79 for true positives and 0.21 for false positives (with 160 correctly predicted positive cases and 42 incorrectly predicted) suggest that the model performs well in predicting positive cases. The true negative rate is the highest among all classes, with a rate of 0.81 and 790 support cases, incorrectly classifying only 191 negative cases as positive. These results indicate that the model performs well as a churn prediction model.

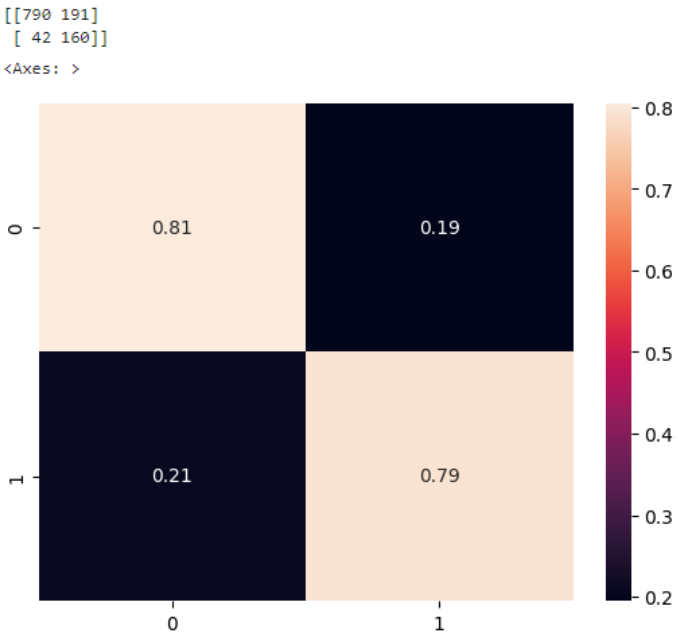

Figure 17. Confusion Matrix and Heatmap for Dataset 2

In Figure 18, the curve hugs the top-left corner, indicating that the model correctly classifies all positives and negatives at every threshold, while AUC-ROC = 1.0 indicates that the perfect discrimination between the positive and negative classes. This is extremely rare in real-world scenarios. However, based on the observations form other evaluation metrics, the overfitting and data leakage problem seem unlikely.

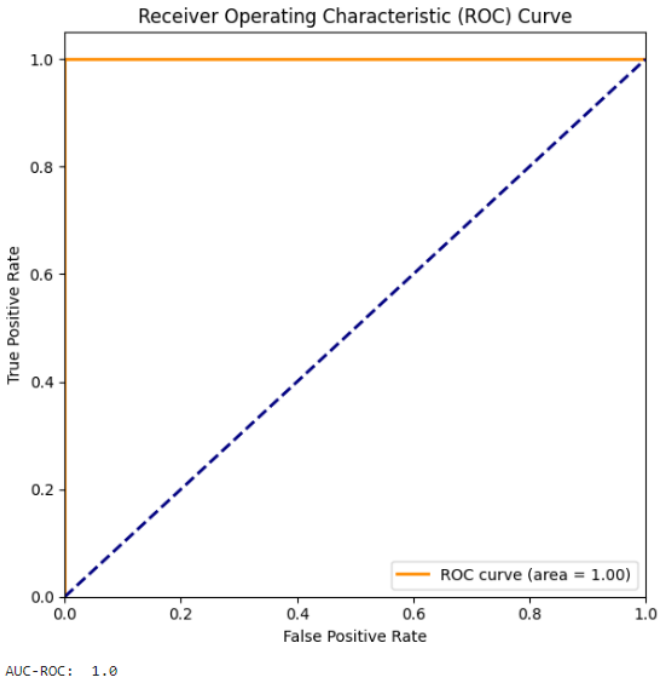

Figure 18. ROC Curve for Dataset 1

In Figure 19, the True Positive Rate (Recall) is plotted against the False Positive Rate, with the model's performance illustrated by the orange curve. The diagonal dashed line represents the baseline of a random classifier, which lacks predictive ability. An AUC-ROC score of 0.8852 indicates that the model has an 88.5% probability of correctly ranking a randomly selected positive instance above a randomly selected negative one, demonstrating strong predictive performance.

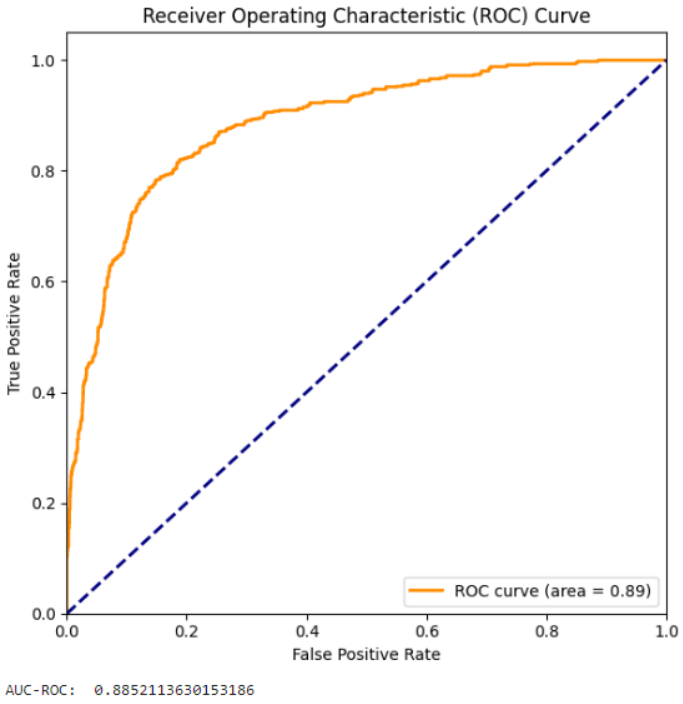

Figure 19. ROC Curve for Dataset 2

In Figure 20, the precision stays at 1.0 across nearly the full range of recall, until a sharp drop at the end—this suggests the model always predicts the correct class until the threshold becomes too relaxed. AUC-PR = 1.0, again indicating perfect performance.

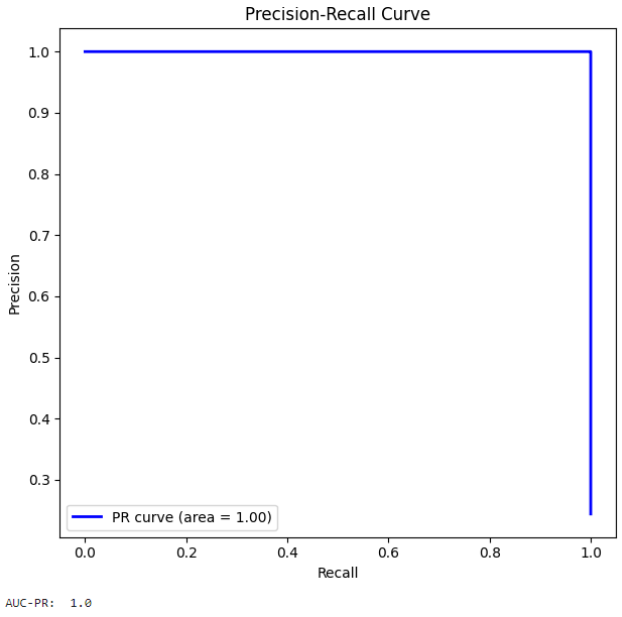

Figure 20. AUC-PR Curve for Dataset 1

Precision is plotted against Recall in the curve in the Figure 21, The AUC-PR score of 0.8859 indicates that the model maintains a good balance between precision and recall across different thresholds, reflecting a strong result.

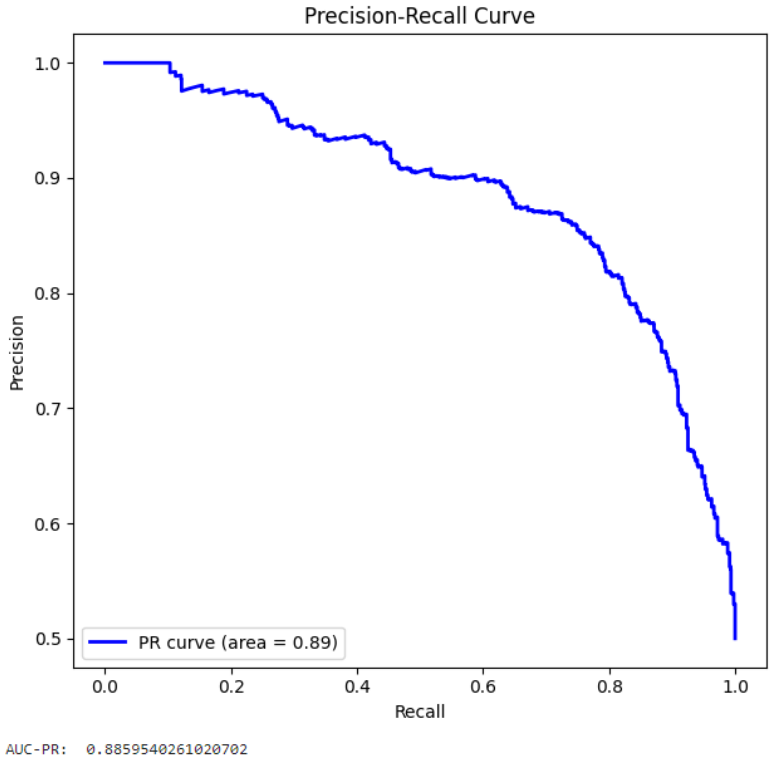

Figure 21. AUC-PR Curve for Dataset 2

The strong performance metrics achieved by the LR model further validate its selection as the primary predictive tool in this study. For Dataset 1, which focuses on conversion rate prediction, the LR model achieved exceptionally high training and testing accuracies, along with macro and weighted F1-Score s nearing 1.0. These results illustrate not only the model's capacity to generalize well to unseen data but also proved again LR's precision in binary classification tasks. Similarly, in Dataset 2 (churn prediction), the LR model maintained stable performance with training and testing accuracies of 0.8091 and 0.8030, respectively. The close alignment between these metrics indicates a low risk of overfitting, a critical consideration in model reliability. The model's performance across key evaluation metrics, including AUC-ROC and AUC-PR, demonstrates its ability to maintain a good balance between precision and recall, especially in imbalanced scenarios. These results affirm that LR is not only computationally efficient and interpretable but also highly capable of delivering consistent and actionable predictions in user behaviour analytics, thus justifying its use as the proposed solution in e-commerce field.

5. CONCLUSION

The implementation of LR to predict user behaviour based on customer demographics and purchase patterns demonstrated strong performance in forecasting customer actions, enabling businesses to make informed decisions about their marketing strategies. Proper evaluation metrics like accuracy, classification report and confusion matrix were used to assess the model's effectiveness. Given the significant success of the LR predictive model implementation, it is important to recognize some limitations, including its potential difficulty in capturing non-linear relationships compared to more complex models. The paper aims to explore further applications of LR as well as other predictive models, including familiar ML such as DT, RF, KNN and SVM, to compare their performance and identify the most effective approach for our data. To achieve better predictive performance, a hybrid model will be implemented by combining multiple predictive models to improve accuracy and generalizability. Additionally, fusion metrics will be further investigated in future work to assess the effectiveness of combining models and optimize their performance.

In conclusion, this study contributes to the growing body of research on predictive modelling in customer behaviour by showcasing the successful application of LR, while also highlighting areas for improvement, such as testing model generalizability across different industries, or across different data structure in e-commerce. Future research will also investigate fusion metrics to optimize model performance and explore the integration of deep learning techniques, such as Natural Language Processing (NLP) to enhance predictive capabilities. By expanding the scope of predictive models, we aim to provide a more comprehensive understanding of customer behaviour and contribute valuable insights for businesses across various sectors.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for the suggestions to improve the paper.

FUNDING STATEMENT

The authors received no funding from any party for the research and publication of this article.

AUTHOR CONTRIBUTIONS

Wei-Wen Lee: Conceptualization, Data Curation, Methodology, Validation, Writing – Original Draft Preparation;
Noramiza Hashim: Project Administration, Writing – Review & Editing;
Shaymaa Al-Juboori: Project Administration, Supervision, Writing – Review & Editing.

CONFLICT OF INTERESTS

No conflict of interests were disclosed.

ETHICS STATEMENTS

Our publication ethics follow The Committee of Publication Ethics (COPE) guideline. https://publicationethics.org/.

REFERENCES

[1] R. Ketipov, V. Angelova, L. Doukovska, and R. Schnalle, "Predicting user behavior in e-commerce using machine learning", *Cybernetics and Information Technologies*, vol. 23, no. 3, p. 89-101, 2023, doi: 10.2478/cait-2023-0026

[2] M. Romzi, N.A. Nabila, S.-C. Haw, W. E. Kong, H. A. Santoso, and G. K. Tong, "Generative AI recommender system in E-commerce," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 14, no. 6, pp. 1823–1835, Dec. 2024, doi: 10.18517/ijaseit.14.6.10509.

[3] T.E. Tong, S.-C. Haw, K.W. Ng, M. Al-Tarawneh, and G.K. Tong, "Performance evaluation on resolution time prediction using machine learning techniques," *International Journal on Informatics Visualization*, vol. 8, no. 2, pp. 583-590, May 2024, doi: 10.62527/joiv.8.2.2305.

[4] X. Zhao and P. Keikhosrokiani, "Sales prediction and product recommendation model through user behavior analytics," *Computers, Materials and Continua*, vol. 70, no. 2, pp. 3855-3874, 2022, doi: 10.32604/cmc.2022.019750.

[5] J. Jayaram, S.-C. Haw, N. Palanichamy, E. Anaam, and S.K. Thillaigovindhan, "A systematic review on effectiveness and contributions of machine learning and deep learning methods in lung cancer diagnosis and classifications," *International Journal of Computing and Digital Systems*, vol. 17, no. 1, pp. 1–12, Jan. 2025, doi: 10.12785/ijcds/1571032811.

[6] Z.-B. Phang, S.-C. Haw, T.-E. Tai, and K.-W. Ng, "Interactive data visualization to optimize decision-making process," in *International Symposium on Parallel Computing and Distributed Systems (PCDS)*, IEEE, pp. 1–6, Sep. 2024, doi: 10.1109/PCDS61776.2024.10743427.

[7] P. A. Sunarya, U. Rahardja, S. C. Chen, Y. M. Lic, and M. Hardini, "Deciphering digital social dynamics: A comparative study of logistic regression and random forest in predicting e-commerce customer behavior," *Journal of Applied Data Sciences*, vol. 5, no. 1, pp. 100-113, 2024, doi: 10.47738/jads.v5i1.155.

[8] N. Sharma, "Analyzing customer behavior patterns & predicting online product return intentions: a data mining approach," Master's Thesis, Faculty of Communication and Environment, Rhine-Waal University of Applied Sciences, Kleve, Germany, 2024.

[9] C.J. Liu, T.S. Huang, P.T. Ho, J.C. Huang, and C.T. Hsieh, "Machine learning-based E-commerce platform repurchase customer prediction model," *PLoS One*, vol. 15, no. 12, 2020, doi: 10.1371/journal.pone.0243105.

[10] X. Hu, Y. Yang, S. Zhu and L. Chen, "Research on a Hybrid Prediction Model for Purchase Behavior Based on Logistic Regression and Support Vector Machine," *2020 3rd International Conference on Artificial Intelligence and Big Data (ICAIBD)*, Chengdu, China, 2020, pp. 200-204, doi: 10.1109/ICAIBD49809.2020.9137484.

[11] X. Chen, S. Ding, Y. Xiang, and L. Liu, "Research on prediction of online purchasing behavior based on hybrid model," in *Journal of Physics: Conference Series*, 2021, vol. 1827, no. 1, 2021, doi: 10.1088/1742-6596/1827/1/012075.

[12] S. Xiao and W. Tong, "Prediction of user consumption behavior data based on the combined model of TF-IDF and logistic regression," in *Journal of Physics: Conference Series*, vol. 1757, no. 1, 2021, doi: 10.1088/1742-6596/1757/1/012089.

[13] Y. Jiang, "Research on prediction of E-commerce repurchase behavior based on multiple fusion models," *Applied and Computational Engineering*, vol. 2, no. 1, pp. 868-882, 2023, doi: 10.54254/2755-2721/2/20220555.

[14] P. Rajapaksha, and D. Asanka, "Recommendations to increase the customer interaction of E-commerce applications with web usage mining," in *Proceedings of IEEE InC4 2023 - 2023 IEEE International Conference on Contemporary Computing and Communications*, pp. 1-6, 2023, doi: 10.1109/InC457730.2023.10263131.

[15] J. Guo, "Research on mobile E-commerce recommendation algorithms based on logistic regression improved model features," *Academic Journal of Engineering and Technology Science*, vol. 7, no. 5, 2024, doi: 10.25236/AJETS.2024.070514.

[16] L. Feng, "Online shopping consumer perception analysis and future network security service technology using logistic regression model," *PeerJ Computer Science*, vol. 10, 2024, doi: 10.7717/peerj-cs.1777.

[17] Y. Qiu, and C. Li, "Research on E-commerce user churn prediction based on logistic regression," *2017 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Chengdu, China, pp. 87-91, 2017, doi: 10.1109/ITNEC.2017.8284914.

[18] A.R. Lubis, S. Prayudani, Julham, O. Nugroho, Y.Y. Lase, and M. Lubis, "Comparison of model in predicting customer churn based on users' habits on e-commerce", *2022 5th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, pp. 300-305, 2022, https://doi.org/10.1109/isriti56927.2022.10052834.

[19] C. Sunita, "Artificial Intelligence in E-commerce: Exploring the purchase decisions through logistic regression analysis," *Quing: International Journal of Commerce and Management*, vol. 3, no. 3, pp. 301-309, 2023, doi: 10.54368/qijcm.3.3.0008.

[20] O. Alkan and S. Unver, "Determination of factors that affect use of E-Commerce in Eastern Turkey through categorical data analysis," *Toros University FEASS Journal of Social Sciences,* 8(Special Issue), pp. 22-36, 2021.

[21] V. Malik, R. Mittal, and S.V. Singh, "EPR-ML: E-commerce product recommendation using NLP and machine learning algorithm," in *Proceedings of 5th International Conference on Contemporary Computing and Informatics (IC3I)*, pp. 1778-1783, 2022, doi: 10.1109/IC3I56241.2022.10073224.

[22] K. Ariansyah, E.R.E. Sirait, B.A. Nugroho, and M. Suryanegara, "Drivers of and barriers to E-commerce adoption in Indonesia: Individuals' perspectives and the implications," *Telecommunications Policy*, vol. 45, no. 8, 2021, doi: 10.1016/j.telpol.2021.102219.

[23] S. Unver, and O. Alkan, "Determinants of E-commerce use at different educational levels: Empirical evidence from Turkey E-commerce use at different educational levels," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 3, 2021, doi: 10.14569/IJACSA.2021.0120305.

[24] S.A. Alquhtani, and A. Muniasamy, "Analytics in Support of E-Commerce Systems Using Machine Learning," *2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, Prague, Czech Republic, pp. 1-5, 2022, doi: 10.1109/ICECET55527.2022.9872592.

[25] Y. Li, J. Shi, F. Cao, and A. Cui, "Product Reviews Analysis of E-commerce Platform Based on Logistic-ARMA Model," *2021 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, Shenyang, China, pp. 714-717, 2021, doi: 10.1109/ICPICS52425.2021.9524238.

[26] P. Kathuria, P. Sethi, and R. Negi, "Sentiment analysis on e-commerce reviews and ratings using ml & nlp models to understand consumer behavior", *2022 International Conference on Recent Trends in Microelectronics, Automation, Computing and Communications Systems (ICMACC)*, 2022. https://doi.org/10.1109/icmacc54824.2022.10093674

[27] S. Zulfiker, A. Chowdhury, D. Roy, S. Datta, and S. Momen, "Bangla E-Commerce Sentiment Analysis Using Machine Learning Approach," *2022 4th International Conference on Sustainable Technologies for Industry 4.0 (STI)*, Dhaka, Bangladesh, pp. 1-5, 2022, doi: 10.1109/STI56238.2022.10103350.

[28] I.A. Shah, "Comprehensive synthetic e-commerce dataset," Kaggle. Accessed: Dec. 12, 2024. [Online]. Available: https://www.kaggle.com/datasets/imranalishahh/comprehensive-synthetic-e-commerce-dataset.

BIOGRAPHIES OF AUTHORS

	**Wei-Wen Lee** is a student pursuing her bachelor's degree at Multimedia University, majoring in Computer Science with a specialization in Data Science. Her research interests primarily focus on Data Analysis, Machine Learning and Deep Learning, with a secondary interest in software engineering, including database management and website development. She can be contacted at email: 1211103858@student.mmu.edu.my (student email) or leevivian12345@gmail.com (personal email).
	**Noramiza Hashim** is a lecturer at the Multimedia University in data science and multimedia technology. Her research interests lie in the fields of image and video processing, computer vision, and machine learning. She has published in various peer-reviewed journals and conferences, reviewed several academic publications, and served on the program committee for academic conferences. She can be contacted at email: noramiza.hashim@mmu.edu.my.
	**Shaymaa Al-Juboori** is a lecturer at the University of Plymouth, specialising Machine Learning, and its applications in healthcare. Her research focuses on using ML techniques, to predict dementia from brain imaging data and EEG. She has published in peer-reviewed journals and conferences and serves as a reviewer for various academic publications. She has actively collaborated on research projects in AI and cybersecurity. She can be contacted at email: shaymaa.al-juboori@plymouth.ac.uk.