Predicting Online Shopper's Purchasing Decisions based on Behaviour

22 Fall Group 16

Tianchen Tang
20935548
ttangai@connect.ust.hk

Yiyun Zhong
20578821
yzhongap@connect.ust.hk

Sai Zhang
20932168
szhangei@connect.ust.hk

Yifan Zuo
20876522
yzuoah@connect.ust.hk

Abstract

Online shopping has emerged worldwide in the last decades. According to the visual capitalist, global eCommerce sales have continued to grow in the previous few years (Visual Caplist 2021). So, we would like to see if we can use the power of machine learning to predict the purchase decisions of different users to help online shopping businesses maximise their revenue more effectively. In this report, the goal is to determine the model most suited for the workload. Meanwhile, we also analysed which features each model thinks are the most important, which can help the development team to determine the behaviour data they need to collect. We found a dataset from the UCI Machine learning repository with over 12,000 records in 1 year. There are 18 behaviour variables; 10 of them are numeric, and 8 of them are categorical. This report focused on four classifiers: logistic regression done by Willian, MLPClassifier done by Evan, Random Forest and K-nearest Neighbour done by Jerry. But only the first three models will involve in the feature importance analysis. Finally, we vote for the best model by F-score and accuracy. Their accuracy score is all around 90%. The best model is MLPClassifier, which achieves a 90.1% accuracy score and 80.0% f1 score. Regarding feature ranking, all models agree "page values" is the most significant feature. However, the other top 5 features are inconclusive.

1. Dataset

Our dataset is on online shoppers' behaviour, retrieved from the UCI Machine Learning Library. It consists of feature vectors of 12,330 sessions, each belonging to a different user in a 1-year period to avoid the tendency to a particular period (Sakar et al., 2018). About 84.5% of the samples are negative, which means that the visitor did not end up purchasing the product. There are ten numerical variables and eight categorical variables. Their descriptions are presented below. Because there is no index for *TrafficType*, *OperatingSystem*, Region and Brower, these four categorical variables are not used in our analysis and, therefore, are removed from the table.

Variable Name	Description
Administrative	The number of administrative pages that the user visited.
Administrative_ Duration	The amount of time spent on administrative pages.
Informational	The number of informational pages that the user visited.
Informational_ Duration	The amount of time spent on informational pages.
ProductRelated	The number of product-related pages that the user visited
ProductRelated_ Duration	The amount of time spent on product-related pages.
BounceRates	The percentage of visitors who enter the website through that page and exit without triggering additional tasks.
ExitRates	The percentage of pageviews on the website that ends at that specific page.
PageValues	The average value of the page is averaged over the value of the

	target page and/or the completion of an eCommerce.
SpecialDay	This value represents the closeness of the browsing date to particular days or holidays.
Month	Contains the month the pageview occurred in string form.
VisitorType	A string representing whether a visitor is New Visitor, Returning Visitor, or Other.
Weekend	A boolean represents whether the session is on the weekend.
Revenue	A boolean represents whether or not the user completed the purchase.

Table 1. Variable Description

2. Exploratory Data Analysis

In the EDA section, our purpose is to identify explanatory variables that might have greater importance in the classification task, explore whether there is any interesting relationship between variables, and better understand our data in the context of helping businesses to maximise revenue through predicting online shopper's purchasing decision based on their behaviours.

Firstly, we explored the distribution of individual variables in the dataset to see whether they differ for customers who eventually purchased the product and those who did not. For numerical variables, our primary tools are numerical summaries and box plots. To make our analysis more structured, we divided our numerical variables into three groups.

The first group is the number of pages visited and duration, which focuses on the type of content and consumption. This group includes variables *Administrative*, *Informational*, *ProductRelated*, and the corresponding time that users spend on them, which are *Administrative Duration*, *Informational Duration*, and *ProductRelated Duration*. Because we are curious about what type of content the visitors, in general, are interested in, we first conducted an aggregated analysis on our group 1 variables. The five-number summary shows that for visitors in general, product-related pages are probably the type that interests the most. It matches our expectations since the purpose of the visit is primarily shopping. And even if a visitor does not end up buying, he/she might still spend time getting information about a specific product and will, therefore, browse through product-related pages. Administrative pages have the second highest mean and median, which might be explained by the fact that visitors might need to fill out relevant account information before they can start to shop. Notice that the 75% quartile for informational pages is still 0, which indicates that most visitors do not look at pages about the Web site, communication, and address information of the shopping site at all. This tells us that most visitors might not find it relevant. The duration measures also show similar trends.

Index	Administrative	Informational	ProductRelated
count	12330	12330	12330
mean	2.31517	0.503569	31.7315
std	3.32178	1.27016	44.4755
min	0	0	0
25%	0	0	7
50%	1	0	18
75%	4	0	38
max	27	24	705

Table 2. Numerical Summary of Page Variables

Index	Administrative Duration	Informational Duration	ProductRelated Duration
count	12330	12330	12330
mean	80.8186	34.4724	1194.75
std	176.779	140.749	1913.67
min	0	0	0
25%	0	0	184.137
50%	7.5	0	598.937
75%	93.2562	0	1464.16
max	3398.75	2549.38	63973.5

Table 3. Numerical Summary of Page Duration Variables

The analysis above yields insight into customers' behaviours. However, for the purpose of maximising revenue for businesses, the core of our research is whether the distribution of these variables can help us distinguish those who are more likely to make purchases from the whole and design more targeted content for them to finalise the transaction. Therefore, a segregated analysis is in order. Below are the boxplots of variables in group 1 grouped by our revenue label, which has the value True for those who made the final purchase and False for those who did not.

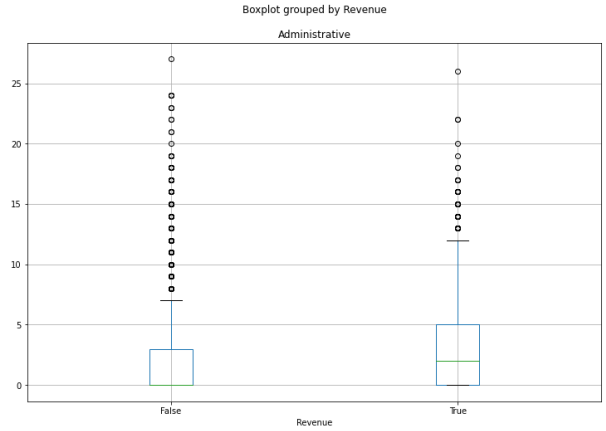

Figure 1. Administrative grouped by revenue

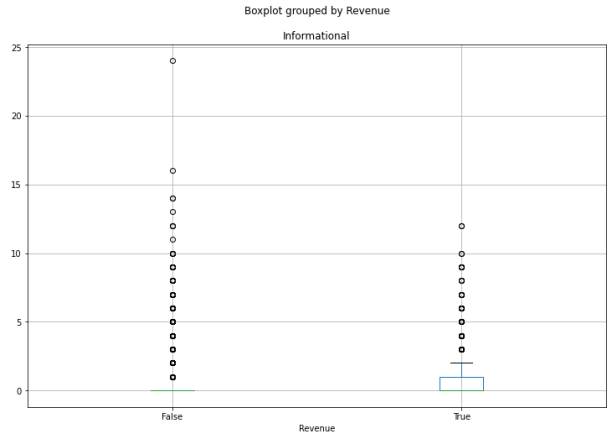

Figure 2. Informational grouped by revenue

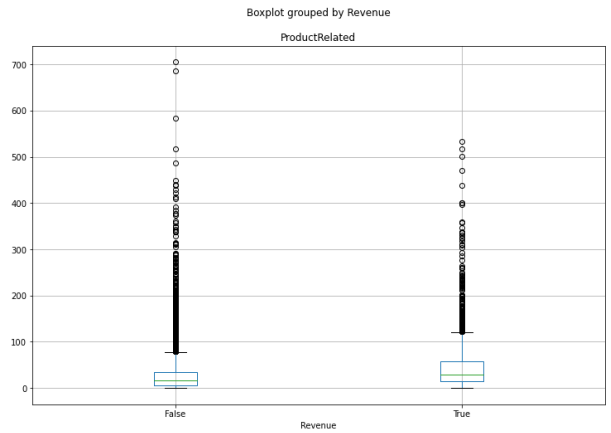

Figure 3. ProductRelated grouped by revenue

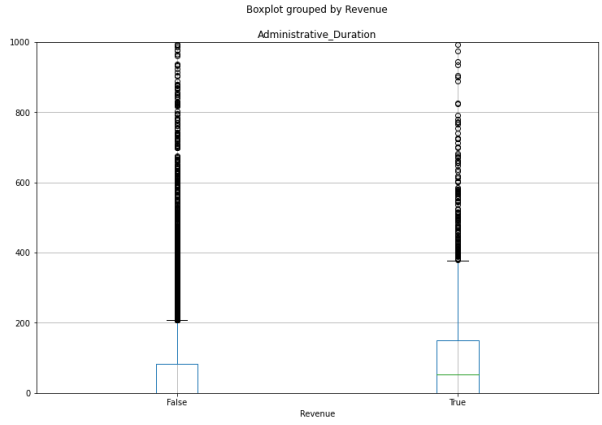

Figure 4. Administrative_Duration grouped by revenue

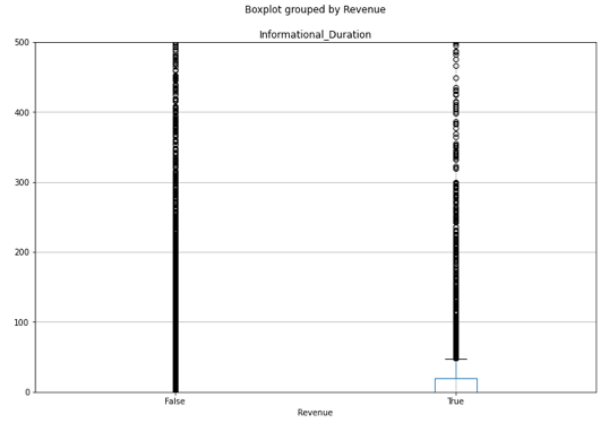

Figure 5. Informational_Duration grouped by revenue

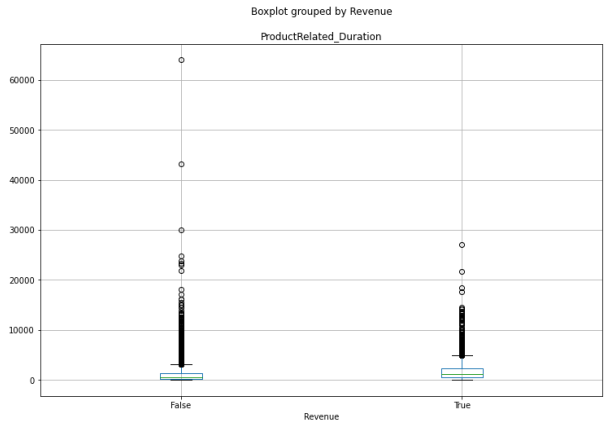

Figure 6. ProductRelated_Duration grouped by revenue

Among these six variables, it seems that *Information* and *Information Duration* are suitable distinguishing variables when, in fact, most of the visitors do not visit informational pages, which is initially surprising. It is likely that the minority that visits the informational pages are the ones who made the purchase. Reflecting on what

informational pages include, we think this situation can be explained by the fact that those who want to finalise the transaction might care more about the website's credentials and the corresponding policy regarding return or exchange, which distinguishes them from customers who are just "window shopping". We initially found these boxplots surprising. The difference in distribution seems plausible.

The second group of numerical variables focuses on the quality of the content. It includes variables *Bounce Rates*, *Exit Rates*, and *Page Values*. From the boxplots, we can see that customers who end up buying tend to have lower bounce rates and exit rates and do tend to read through more pages than those who do not. Additionally, *Page Values* seems to be a strong distinguishing variable as the interquartile range for the True class does not overlap with that for the False class at all.

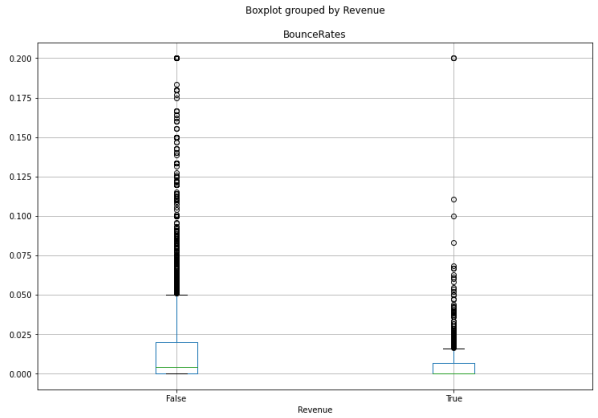

Figure 7. BounceRates grouped by revenue

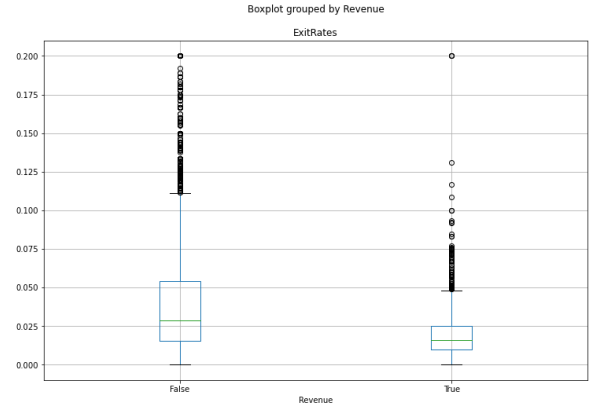

Figure 8. ExitRates grouped by revenue

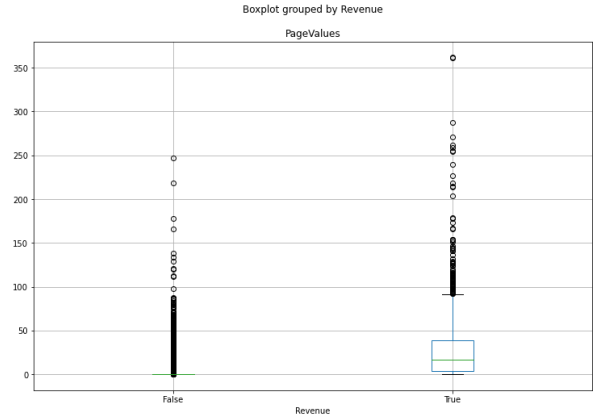

Figure 9. PageValues grouped by revenue

The last numerical variable is *Special Day*. After running a five-number summary, we found that the 25%, 50%, and 75% quartile for both groups of visitors are 0, which means that most of the customers visit the site on a date that is not close to any of the special days. This corresponds with our expectations since there are not many special occasions in a year.

For categorical variables, *Weekend*, *Visitor Type*, and *Month*, we use bar graphs to check whether the distributions differ. The chart on the left shows the percentage of visitors who eventually made the purchase grouped by month, and the variation appears to be quite significant. For November, we can see that the ratio is over 25%, whereas the percentage for February is below 5%. This is reasonable since e-commerce can also be considered seasonal. In November, businesses prepare for Black Friday by planning discounts and activities to boost sales, so the number of successful transactions might be significantly greater. On the other hand, a month like March might have lower sales since it is in the middle of Spring and does not have any traditions for shopping and gifting. On the other hand, the percentage of purchases made on the weekend is about 2.5% lower than those made during the weekdays. The difference might be too small to be helpful in predicting a shopper's purchasing decision, but we cannot conclude so until we construct a model. Finally, the percentage of the purchase grouped by visitor type differs, with new visitors ranking the highest, followed by other visitors and then returning visitors. Compared to new visitors, the purpose of visiting the site for returning customers might not be to make a purchase. Instead, it could be checking the previous orders, membership benefits, and so on, which might explain the lower percentage.

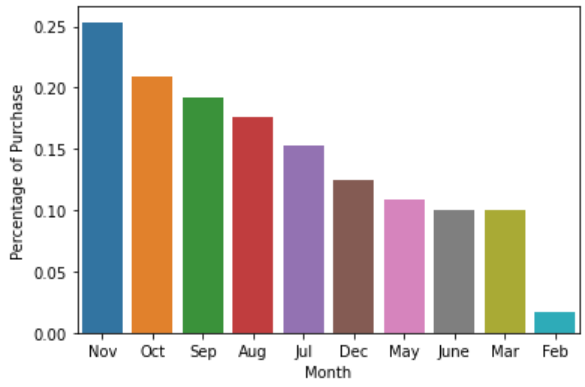

Figure 10. Percentage of Purchase by Month

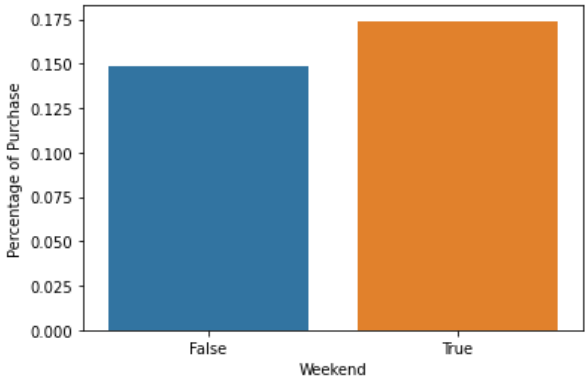

Figure 11. Percentage of Purchase by Weekend

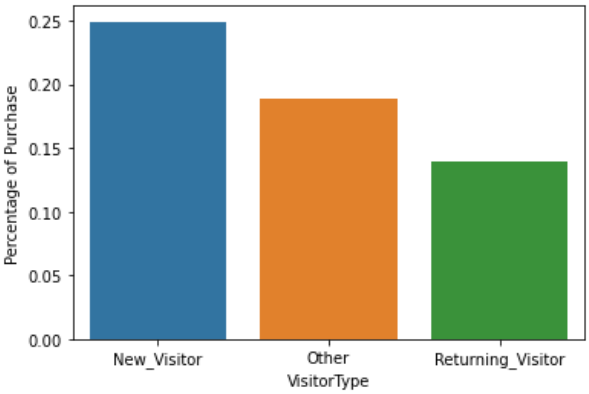

Figure 12. Percentage of Purchase by VisitorType

Before we enter the data pre-processing phase, it is necessary to examine the correlation between variables. Because one of the models that we are planning to construct is logistic regression, we hope to check whether we might encounter multicollinearity, which refers to the statistical phenomenon in which several independent variables in a model are correlated. Although it does not affect the regression estimates, it can make determining the effect of each explanatory variable on the label difficult when interpretability is one of the strong advantages of a regression model (Hayes, 2022).

Additionally, we hope to explore whether there are interesting relationships that can yield insight into our research question. According to the correlation matrix, there are two pairs of variables that are highly correlated, which are *Product Related* and *Product Related Duration*, and *Bounce Rates* and *Exit Rates*. For the first pair, it is not surprising since both of them measure visitors' interest in product-related pages. The second pair is far more interesting. Bounce rate and exit rate are both metrics for e-commerce on Google Analytics and are often confused with one another. For all pageviews to the page, the exit rate is the percentage that was the last in the session. For all sessions that start with the page, the bounce rate is the percentage that was the only one of the sessions (Google). A high exit rate does not necessarily equate to a high bounce rate. Furthermore, these two metrics are designed for different purposes. Bounce rate is often associated with user satisfaction, whereas exit rate usually signifies problems with the conversion funnel (Bounce rate vs exit rate: What's the difference? 2022). Therefore, although these two variables are highly correlated, it might be hasty to consider one of them as redundant as they provide different insights for e-commerce. To address the problem of multicollinearity, we are going to use a regularised logistic regression model.

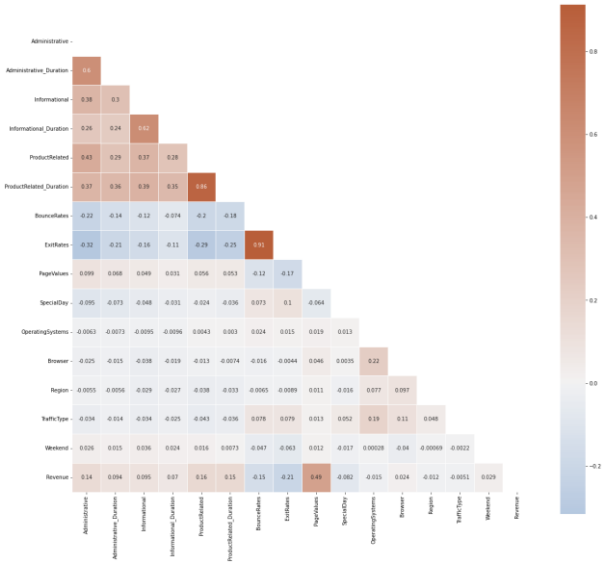

Figure 13. Correlation Matrix

3. Data Pre-processing

One problem we encountered was the lack of an index for four categorical variables, including *Region*, *Operating System*, *Traffic Type*, and *Browser*. Because we would not be able to conduct helpful analysis without knowing what the numerical values represent, we decided to drop them from our model. Additionally, we found that our dataset did not contain data in January and April and was not able to find another source other than the UCI Machine Learning Library. However, considering that these two months might not be representatives of a specific type and other months like November obviously stand out from our EDA, we decided to keep the month variable in our dataset.

We split our dataset into the training set and test set, with the test size being 0.3. Based on the histogram of the numerical variables, we can see that many of our variables are quite right-skewed, which is reasonable since customers might visit the site for a variety of purposes and, therefore, can behave very differently. To tackle the problem of skewness, we applied a log transformation to variables with a value of skewness larger than 0.5 in both the training and test set.

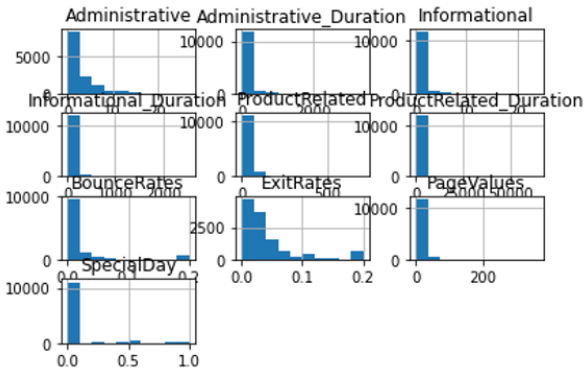

Figure 14. Distribution of Numerical Variables

To take a glance at whether the feature importance corresponds to our EDA, we used the chi-square filter method. We applied a min-max scaler to fit it in, and the result indicates that *Page Values*, *Product Related Duration*, *Product Related*, *Informational Duration*, and *Administrative Duration* as the top 5 features. Among these features, the score for *Page Values* stands out, which matches our finding in the boxplot.

Before we entered the model constructing phase, we transformed the categorical variables into dummies and then applied a standard scaler to both the training and test set because variables on different scales might not contribute equally to the model and might lead to bias (*How and why to standardise your data: A python tutorial, 2020*).

4. K-Nearest Neighbour & Random Forest

4.1 Simple KNN

For K-Nearest Neighbours model training, we use three kinds of models.

First, we use the simple KNN model to train our dataset. We run the for loop code to find the best K value. As a result, here is the relationship between the accuracy score and the k value.

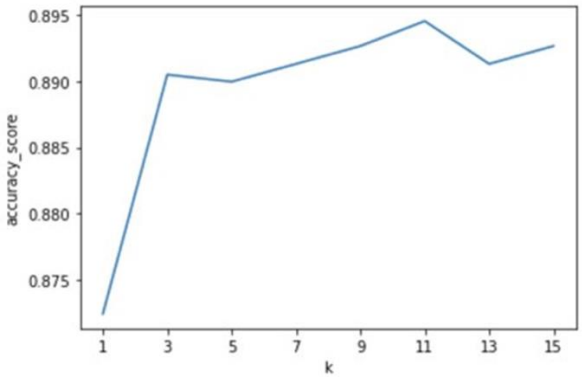

Figure 15. Accuracy VS k value (Simple KNN)

As shown in this diagram, when the k value is equal to 11, we have the best accuracy, 89.5%, and the F1 score is 59.11%.

4.2 KNN with Cross-Validation

To make this KNN model more generalised, we use the 10-fold cross-validation method for the KNN model. Figure 16 shows the relationship between the accuracy score and the k value. As we can see, when the k value is equal to 13, we have the best accuracy, 89.3%, and the F1 score is 59.11%.

Figure 16. Cross-Validation

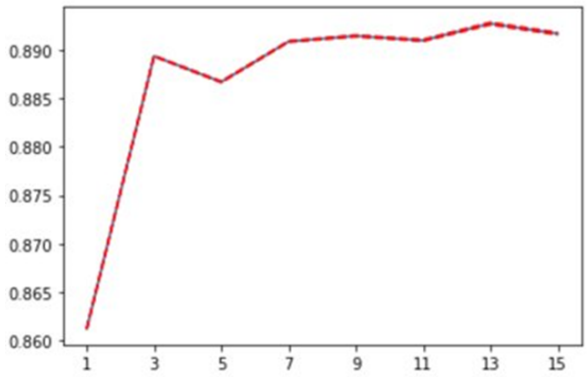

Figure 17. Accuracy vs K value

Although the accuracy is a little bit lower than the simple KNN model, the generalisation ability of this model has improved a lot.

4.3 Grid Search

To better consider more parameters, such as the weight and distance in the KNN method. We use 10-fold cross-validation and grid search for KNN to train again. By grid search, we have these best parameter choices: k is equal to 14, and this model decides to give more weight to the nearest neighbour data. And finally, the accuracy is 89.4%, and the F1 score is 59.11%.

```
test accuracy:  0.8964585022979183
best score: 0.894451150165229
best model: KNeighborsClassifier(n_neighbors=14, p=3, weights='distance')
F1 score: 0.5911431513903194
```

Figure 18. Grid Search Results

4.4 Random Forest

However, the KNN model is not easy to analyse each parameter's importance. Therefore, we use the random forest structure to do the analysis. For the result, the accuracy is equal to 90.02%, and the F1 score is 62.31%. The performance is better than the KNN model.

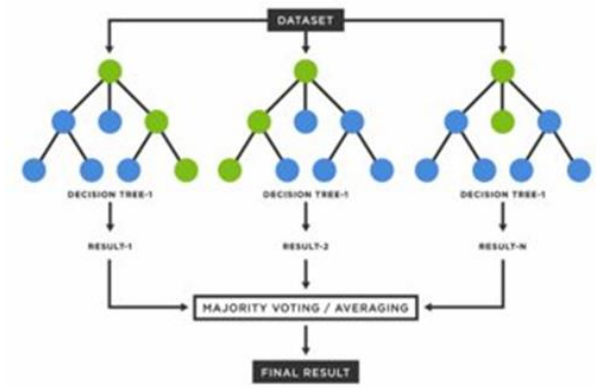

Figure 19. Random Forest Structure

4.5 Evaluation

For the random forest model evaluation, we have the "Feature Importance" parameter. Feature importance is computed as the mean and standard deviation of accumulation of the impurity decrease within each tree. And the top 5 features are: *Page Values*, *Product Related Duration*, *Bounce Rates*, *Exit Rates* and *Product Related*. And we notice that the parameter Page Values is much more important than other parameters, which is consistent with the assessment results of other models.

```
1) PageValues                 0.431736
2) ProductRelated_Duration    0.102532
3) BounceRates                0.086050
4) ExitRates                  0.083915
5) ProductRelated             0.068055
```

Figure 20. Top 5 important features

5. Logistic Regression

5.1 Building Model

Before we can train the model, we should only keep one feature in a highly correlated feature group for the vanilla implementation of the logistic regression model. However, sklearn is smart enough to apply regularisation automatically, penalising repeated information ("Sklearn documentation", 2022). We just need to fit the training data into the labels using pre-processed data produced by the previous steps. We increased the maximum number of iterations to 4096 to avoid convergence warnings.

5.2 Evaluation

For this model, we used a confusion matrix to see how each accuracy class was distributed. There are 3010 true positives, 117 false positives, 255 false negatives and 317 true negatives.

		Actual Values	
		Positive	**Negative**
Predicted Values	**Positive**	3010	117
	Negative	255	317

Table 4. Confusion Matrix of Logistic Regression

The accuracy score of the model is relatively high, which is 89.8%. However, if we use a harsher metric like the f1 score, the number will drop to 62.7% since the model is not good at predicting negative classes correctly, which we can also tell from the confusion matrix in Table 1 above.

5-fold cross-validation to see if the model overfitted the dataset.

#	Accuracy Score
1	0.90040533
2	0.89397451
3	0.88528389
4	0.9032445
5	0.89049826

Table 5. 5-Fold Result of Logistic Regression

All accuracy scores are around 90% which indicates no overfitting issue.

5.3 Feature Ranking

In this section, let us look at how each feature contributes to the prediction. There is a coefficient variable of the logistics regression in sklearn. However, since the logistic regression uses the logit function maps a linear regression model to a classifier. The raw values of the model coefficient are not convenient to interpret. We need to get the odd ratio by the following formula (George Choueiry n.d.):

$$odd = e^{coefficient}$$

Odds are the probability of events occurring divided by the probability of events not occurring.

#	Feature	Odd
1	PageValues	4.18
2	Month_Nov	1.46
3	ProductRelated	1.3
4	VisitorType_New_Visitor	1.15
5	Informational	1.12

Table 6. Top 5 Features of Logistic Regression

The most significant feature is page values, which are the number of pages a user visits before completing the transaction. Its odd is over 4.18, which means if the user visits one more page, the odds of a user will purchase an item increase by 318%. The value is four times more than most other features in the result. The second feature is the Month of November, which means if a user visits the website in November, the odds of the user will purchase an item will increase by 46%. It is easier to understand in the context of America since black Friday is in November, and many people are preparing supplies for Christmas. The third feature is Product Related. If the number of different types of pages increases by 1, the odds of the user will increase by 30%. The fourth feature is the new visitor type. If a user is a new visitor, the odds of the user making a purchase will increase by 15%. The last feature is Informational, which is the time user spends on an informational page. So, if a user spends more time on informational pages, the odds of a successful purchase will increase by 12%.

6. Neural Network

6.2 Parameter Tuning

Solver: Adam

Solver	lbfgs	sgd	adam
Accuracy	88.77%	89.1%	90.1%

Table 7. Solvers accuracy comparison

We tried all three solvers. We found that Adam achieves the best performance, which is 90.1% since it combines the adaptive learning rate and the moments.

Hidden layer size: (6, 3):

Number of perceptrons: (we fix the second layer to be 3)

#perceptrons	3	6	10	20	30
Accuracy	88.12%	**89.1%**	89.07%	88.14%	87.51%

Table 8. Number of perceptron accuracy comparison

We see that through the experiments, less than three perceptron layers drop the accuracy compared to six perceptrons since too few perceptrons cannot represent the function of the dataset, then the model is underfitting the data. Larger than six perceptrons also drop the accuracy compared to six perceptrons because too many perceptrons overfit the data.

Number of layers: (we fix the number of the perceptron to be 6)

#layers	1	2	3	5	10
Accuracy	88.65%	**89.1%**	88.94%	87.56%	87.01%

Table 9. Number of layers accuracy comparison

In the experiments, the result shows that the best performance accuracy, which is 89.1%, is provided by the two layers network. Too less number of layers decrease the accuracy since it cannot represent a complex model, and too many layers also decrease the accuracy, which is due to the difficulty of gradients propagating back to the lower layer.

We use Relu as our activation function since Relu is commonly used in the hidden layer as the activation function. We set Max iteration to be 1000 since our model cannot converge within the default 200 iterations, so we set max iteration to be 1000 in order to let the model to converge. And we set Alpha to be 0.0001 because we use the default value for the strength of the L2 regularisation term. The tuned hyperparameters with the model could achieve 90.1% accuracy and 80.0% f-score.

6.3 Permutation Feature Importance

Permutation feature importance measures the increase in the prediction error of the model after we permuted the feature's values, which breaks the relationship between the feature and the true outcome.

6.3.1 Theory:

A feature is considered significant if shuffling its values increases the model error because, in this case, the model relied on the feature for the prediction. A feature is considered non-significant if shuffling its values leaves the model error unchanged because, in this case, the model ignored the feature for the prediction.

6.3.2 Algorithm:

The permutation feature importance algorithm based on Fisher, Rudin, and Dominici (2018):

Input: Trained model $\hat{f}$, feature matrix X, target vector y, error measure $L(y, \hat{f})$.

1. Estimate the original model error $e_{orig} = L(y, \hat{f}(X))$ (e.g. mean squared error)
2. For each feature $j \in \{1, \ldots, p\}$ do:
   - Generate feature matrix X_{perm} by permuting feature j in the data X. This breaks the association between feature j and true outcome y.
   - Estimate error $e_{perm} = L(Y, \hat{f}(X_{perm}))$ based on the predictions of the permuted data.
   - Calculate permutation feature importance as quotient $FI_j = e_{perm}/e_{orig}$ or difference $FI_j = e_{perm} - e_{orig}$
3. Sort features by descending FI.

Figure 21. Permutation Feature Importance Algorithm

6.3.3 Result:

The Original model error is 0.09813. Then we do transformation, which means how much percentage of each feature error exceeds the original error. The calculation formula is provided below: features_error = (features_error - original_error) / original_error

Each feature error after the transformation:

PageValues	0.607713
Month_Nov	0.119284
Informational_Duration	0.108264
Month_Dec	0.105510
Administrative_Duration	0.099725
SpecialDay	0.068320
ExitRates	0.067769
Administrative	0.067493
Month_May	0.065289
Month_Jul	0.063085
Month_Oct	0.060331
VisitorType_New_Visitor	0.059504
ProductRelated	0.058953
ProductRelated_Duration	0.055372
Month_June	0.053994
Month_Sep	0.051240
Month_Mar	0.050689
BounceRates	0.050413
Informational	0.050138
Weekend	0.048209
Month_Aug	0.047658
VisitorType_Returning_Visitor	0.046832
VisitorType_Other	0.046556
Month_Feb	0.038017

Figure 22. Top 5 Important Features for MLPClassifier with Permutation Feature Importance

The result shows that the most significant feature is page value. This feature error exceeds the original error by 60.7%. This means that the page value has a really big impact on the result of the transaction. The top five significant features are page value, the month of November, information duration, the month of December and administrative duration.

7. Conclusion

During our whole project, we have two main parts: exploratory data analysis and model training.

For the EDA part, we analyse the type of content and the corresponding duration; analyse the quality of the content; do the category visualisation; analyse the correlation between variables; analyse Bounce Rates vs Exit Rates. For the model training part, we achieve Logistic Regression; use MLPClassifier; realise three kinds of KNN algorithms, and finally use random forest structure. Here is our model comparison diagram.

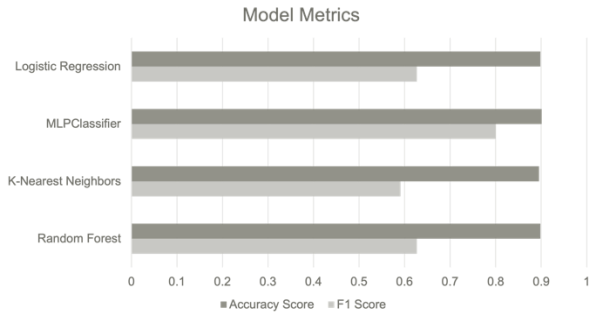

Figure 23. Model Metrics

All these models have similar accuracy scores, around 90%. However, the F1 score of MLPClassifier is much higher than other models. This means this model is more robust and more suitable for this problem.

#	Logistic Regression	MLPClassifer	Random Forest
1	**Page Value**	**Page Values**	**Page Values**
2	**November**	**November**	Product Related Duration
3	**Product Related**	Informational Duration	Bounce Rates
4	New Visitor	December	Exit Rates
5	Informational	Administrative Duration	**Product Related**

Table 10. Feature Importance Comparison

Here is one interesting parameter: page values. As shown in the table, the feature is always the most important feature among all the models.

According to this feature's definition: page values are the average value for a web page that a user visited before completing an e-commerce transaction. So combined with real life, when people are about to buy a certain product, they tend to read this product web page carefully. This common sense fits our result and fits this interesting parameter.

To sum up, we implemented the EDA and data pre-processing tasks thoroughly and carefully. We trained different models and compare their accuracy scores and F1 scores to find the best model. For each model, we have the top 5 most important parameters, and the page value feature is always the most important feature among all the models. This result is important and tells us about the most important factor behind our shopping purchase dataset.

8. Further Discussion

So far, we have discovered which model performs the best among the candidate models and what important factors affect the online shopper's decision in different models. However, the pageValue is the only common feature that all models consider significant. We could discover why the different models with different methods give different feature rankings. Furthermore, which one is closest to the true feature ranking and why the others are not close to that?

9. Acknowledgement

All members of the team actively participated in this project throughout the term. Christina is responsible for dataset, EDA and data pre-processing. William is responsible for logistics regression and report formatting. Evan is responsible for the MLPClassifer and Abstract with William. Finally, Jerry is responsible for the KNN, Random Forest and conclusion.

Reference

Carmen Ang. (July 5, 2021). *Timeline: Key Events in the History of Online Shopping*. Visual Caplist. Retrieved from https://www.visualcapitalist.com/sp/history-of-online-shopping/

Bounce rate vs exit rate: What's the difference? CXL. (2022, September 12). Retrieved November 21, 2022, from https://cxl.com/guides/bounce-rate/bounce-rate-vs-exit-rate/#h-what-is-the-difference-between-bounce-rate-and-exit-rate

Google. (n.d.). *Exit rate vs Bounce Rate - analytics help*. Google. Retrieved November 21, 2022,

from https://support.google.com/analytics/answer/2525491?hl=en&ref_topic=6156780

Hayes, A. (2022, September 27). *Multicollinearity*. Investopedia. Retrieved November 21, 2022, from https://www.investopedia.com/terms/m/multicollinearity.asp

How and why to standardise your data: A python tutorial. (n.d.). Retrieved November 21, 2022, from https://towardsdatascience.com/how-and-why-to-standardize-your-data-996926c2c832

Sakar, C. O., Polat, S. O., Katircioglu, M., & Kastro, Y. (2018, May 9*). Real-time prediction of online shoppers' purchasing intention using Multilayer Perceptron and LSTM recurrent neural networks - neural computing and applications*. SpringerLink. Retrieved November 26, 2022, from https://link.springer.com/article/10.1007/s00521-018-3523-0#citeas

Sklearn documentation for logistic regression. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

George Choueiry. *Interpret Logistic Regression Coefficients*. (n.d.). Retrieved from https://quantifyinghealth.com/interpret-logistic-regression-coefficients/

Interpretable Machine Learning: Permutation Feature Importance. (2022, November 12). Retrieved from https://christophm.github.io/interpretable-ml-book/feature-importance.html