# FINANCIAL DATA MINING

## PROJECT DESCRIPTION

In this project, we plan to predict real-time online shopper behavior analysis system i.e. whether the user will be purchasing something from an e-commerce site. There is a constant need for e-commerce sites to improve their model based on how much time user is spending while surfing the net and the period at which he is doing it, which causes a need for such model to arise. We have used 5 models for this: Naive Bayes, Logistic Regression, decision trees, random forest, and XGboost.

## PROBLEM

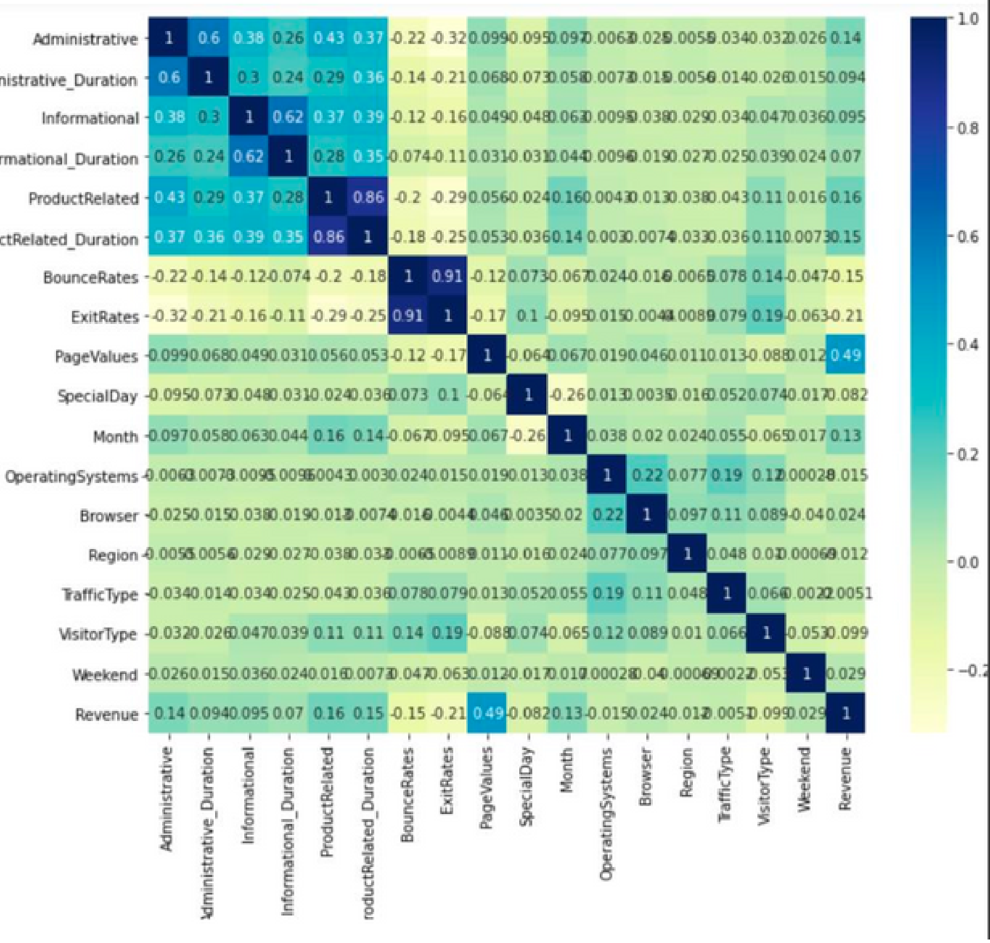
Purchase prediction has an important role for decision making in e-commerce to improve consumer experience, provide personalized recommendations and increase revenue. Many works investigated purchase prediction for session logs by analysis of users' behavior to predict purchase intention after a session has ended. In most cases, e-shoppers prefer to be anonymous while browsing the websites and after a session has ended, identifying users and offering discounts can be challenging. Therefore, after a session ends, predicting purchase  intention may not be useful for the e-commerce strategists.

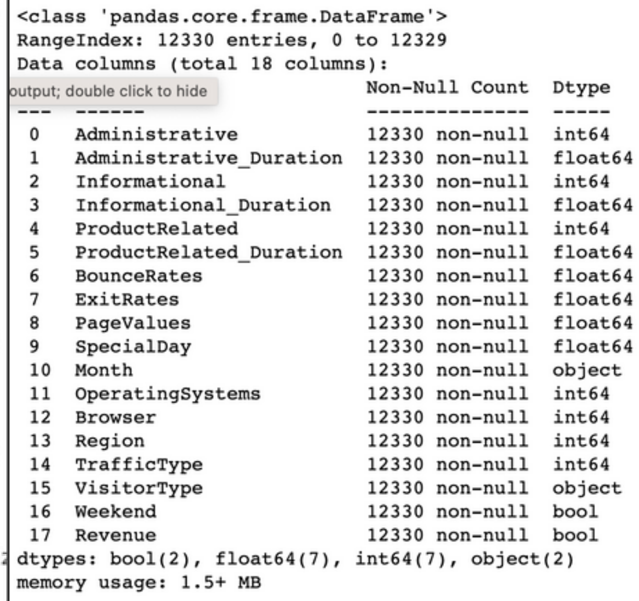
# CUSTOMER INTENT PREDICTION

ANIRUDH GAUR   ROHIT MACHERLA   VARUN GANDIKOTA

## ABOUT THE DATA

This project covers Online shoppers purchase intention dataset: (https://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Intention+Dataset) The dataset consists of feature vectors belonging to 12,330 sessions and was formed in such a way that each session would belong to a different user in a 1-year period to avoid any tendency to a specific campaign, special day, user profile or period. Below is the correlation plot and some more important information to look at glance.

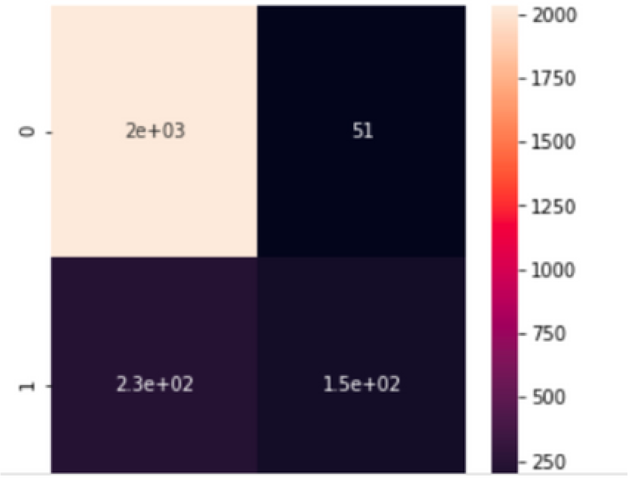




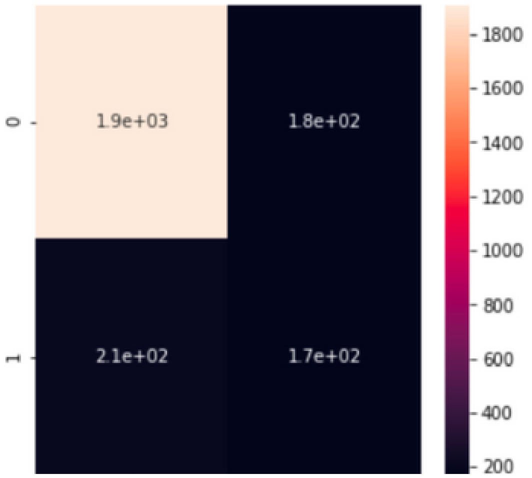
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12330 entries, 0 to 12329
Data columns (total 18 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Administrative           12330 non-null  int64
 1   Administrative_Duration  12330 non-null  float64
 2   Informational            12330 non-null  int64
 3   Informational_Duration   12330 non-null  float64
 4   ProductRelated           12330 non-null  int64
 5   ProductRelated_Duration  12330 non-null  float64
 6   BounceRates              12330 non-null  float64
 7   ExitRates                12330 non-null  float64
 8   PageValues               12330 non-null  float64
 9   SpecialDay               12330 non-null  float64
 10  Month                    12330 non-null  object
 11  OperatingSystems         12330 non-null  int64
 12  Browser                  12330 non-null  int64
 13  Region                   12330 non-null  int64
 14  TrafficType              12330 non-null  int64
 15  VisitorType              12330 non-null  object
 16  Weekend                  12330 non-null  bool
 17  Revenue                  12330 non-null  bool
dtypes: bool(2), float64(7), int64(7), object(2)
memory usage: 1.5+ MB
```

# METHODOLOGY

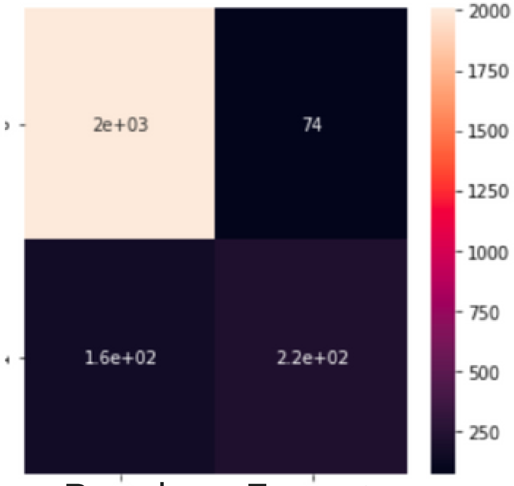
We decided to do an 80-20 split between training and test data set. Parameters such as f1 score, accuracy, misclassification error, etc. would be calculated. Normalization has been done on dataset to change the value of numeric columns to a common scale. Below are the confusion matrix:



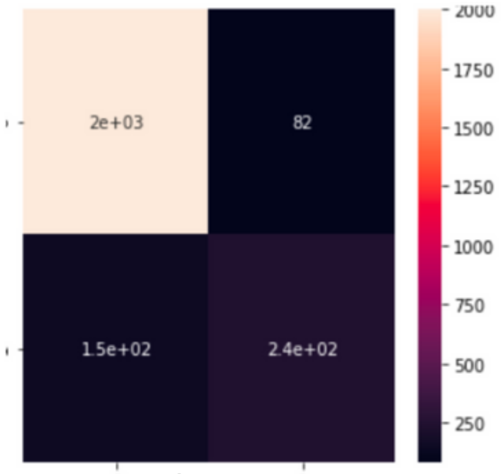
Logistic Regression



Decision Tree

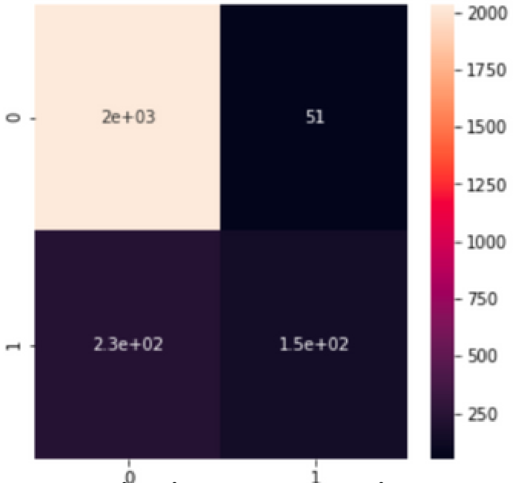


Random Forest

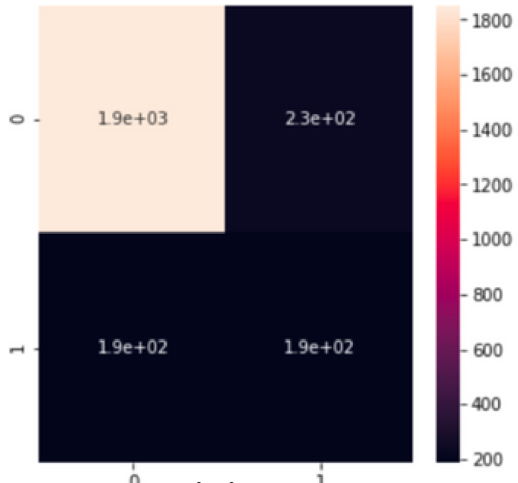


XGBoost

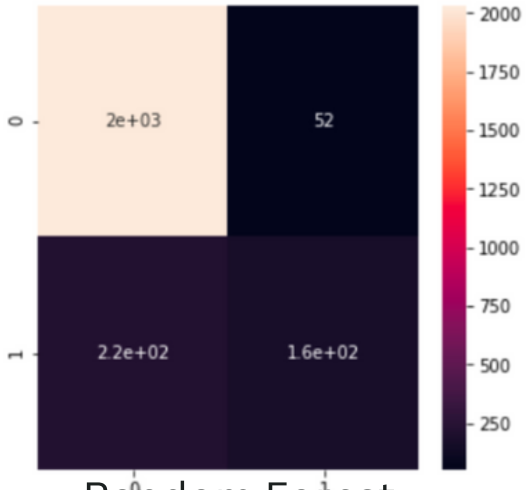
We have also used Principle Component Analysis (PCA) which is a common feature extraction method in data science. Technically, PCA finds the eigenvectors of a covariance matrix with the highest eigenvalues and then uses those to project the data into a new subspace of equal or less dimensions. The confusion matrix for models after PCA are below



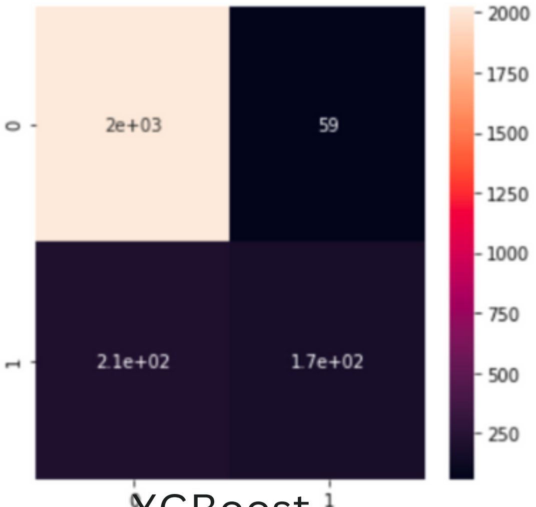
Logistic Regression



Decision Tree



Random Forest



XGBoost

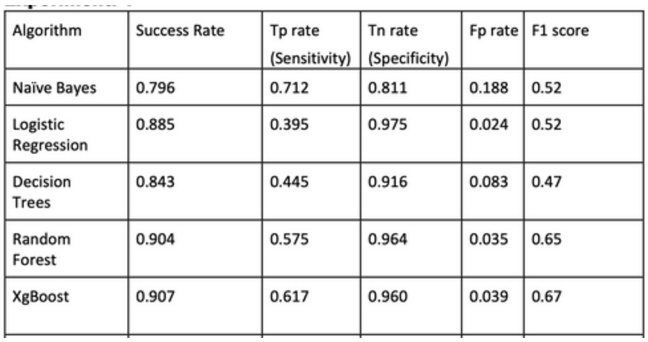
# FINDINGS & EVALUATION

**Accuracy:**Accuracy determines the number of correct predictions over the total number of predictions made by the model. The formula of accuracy is- Accuracy = (TP+TN)/(TP+TN +FP+FN).

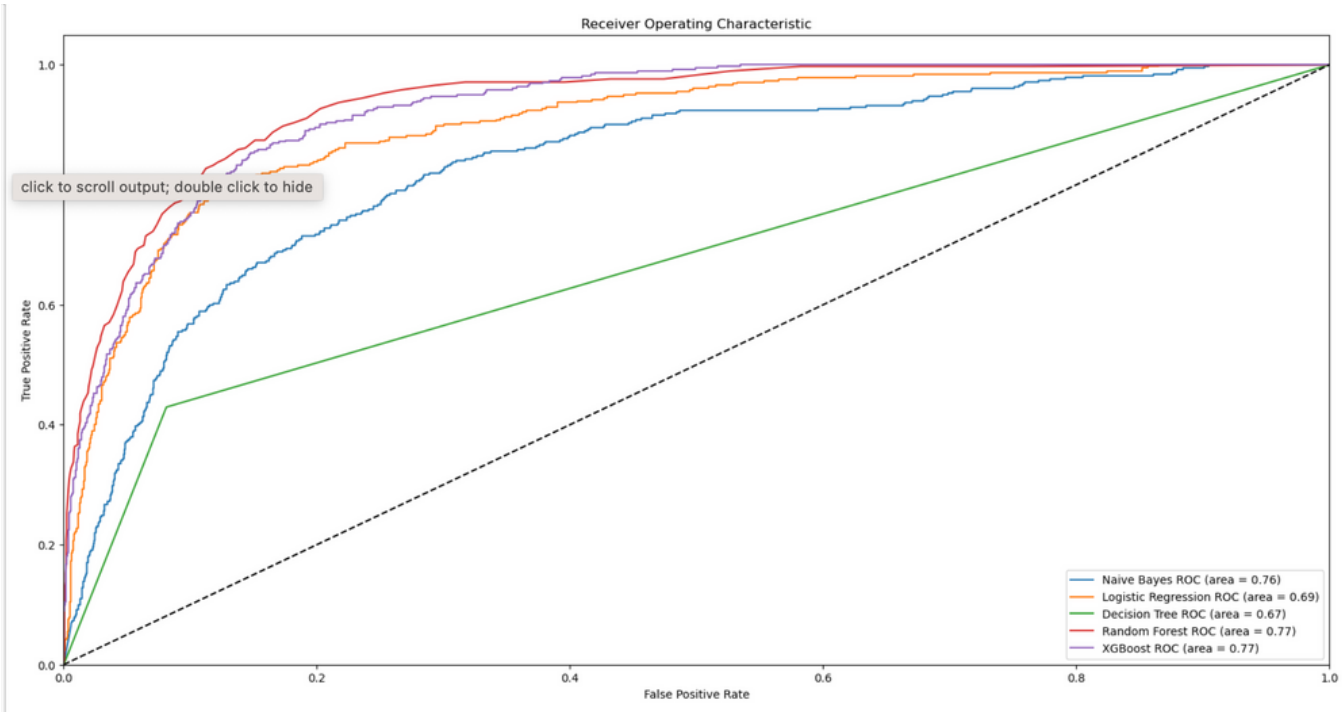
**Sensitivity**: Classifier's performance to spot positive results is related by Sensitivity. Sensitivity is calculated as follows Precision = TP/(TP+FN).

**Specificity**: Classifier's performance to spot negative results is related by Specificity. Specificity is calculated as follows: TN/(TN+FP).

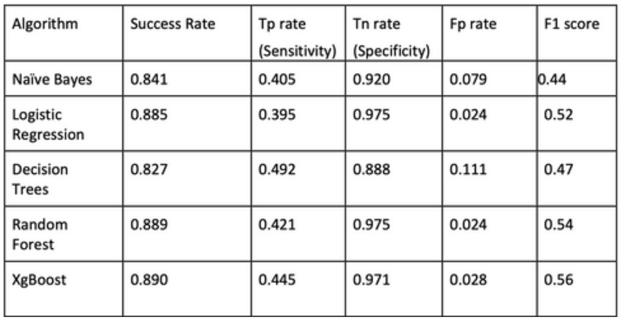
Comparing the 5 models on the basis of success rate, tp rate, tn rate, fp rate and f1 score:



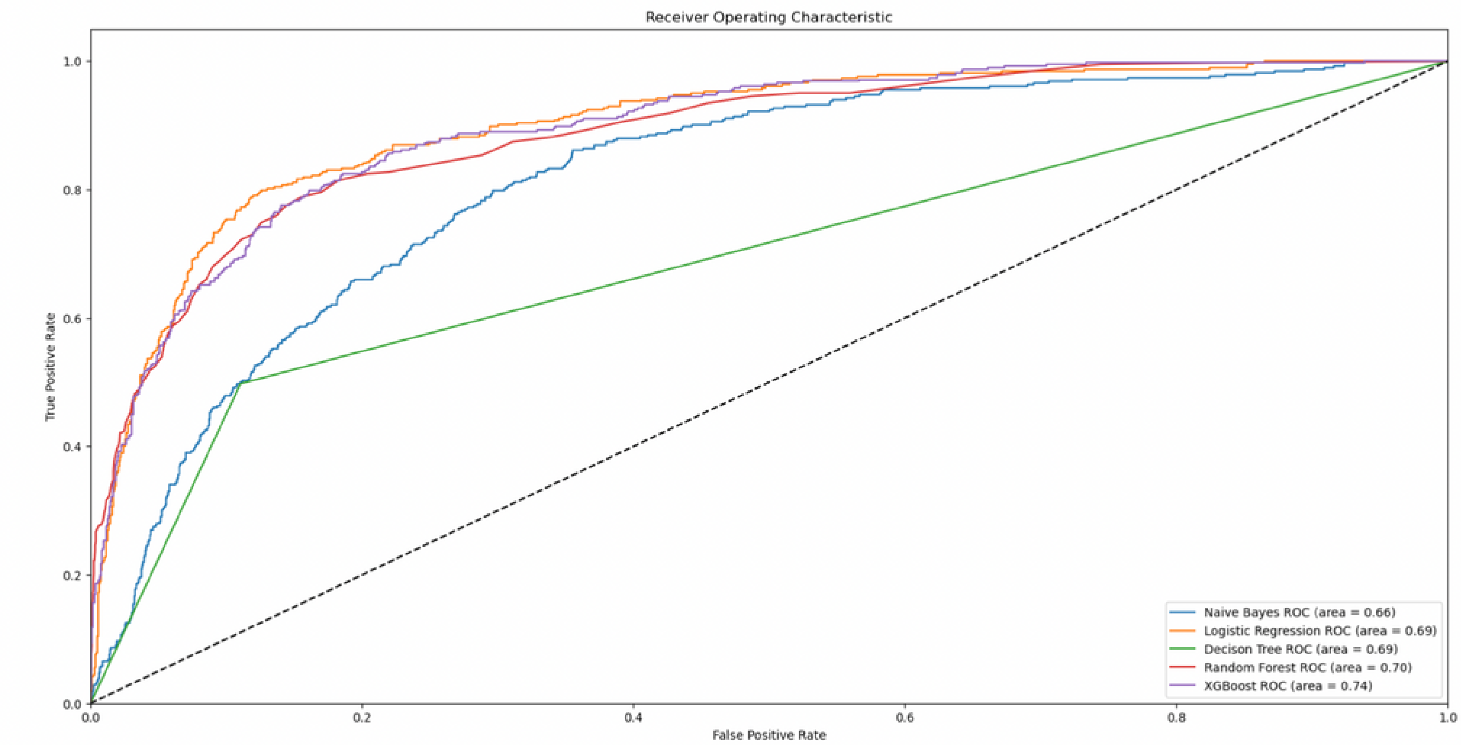
| Algorithm | Success Rate | Tp rate (Sensitivity) | Tn rate (Specificity) | Fp rate | F1 score |
|---|---|---|---|---|---|
| Naïve Bayes | 0.796 | 0.712 | 0.811 | 0.188 | 0.52 |
| Logistic Regression | 0.885 | 0.395 | 0.975 | 0.024 | 0.52 |
| Decision Trees | 0.843 | 0.445 | 0.916 | 0.083 | 0.47 |
| Random Forest | 0.904 | 0.575 | 0.964 | 0.035 | 0.65 |
| XgBoost | 0.907 | 0.617 | 0.960 | 0.039 | 0.67 |



Similar comparison can be done for the 5 models using PCA:



| Algorithm | Success Rate | Tp rate (Sensitivity) | Tn rate (Specificity) | Fp rate | F1 score |
|---|---|---|---|---|---|
| Naïve Bayes | 0.841 | 0.405 | 0.920 | 0.079 | 0.44 |
| Logistic Regression | 0.885 | 0.395 | 0.975 | 0.024 | 0.52 |
| Decision Trees | 0.827 | 0.492 | 0.888 | 0.111 | 0.47 |
| Random Forest | 0.889 | 0.421 | 0.975 | 0.024 | 0.54 |
| XgBoost | 0.890 | 0.445 | 0.971 | 0.028 | 0.56 |



# CONCLUSION

The online shopping customer intention problem is a binary classification task, and we evaluated five models with two techniques: normalization and PCA. Based on metrics such as F1 score and ROC curve, XGBoost outperformed the other models with Random Forest as second the best model. For future work, additional feature extraction techniques like RFE and machine learning models such as neural networks could be explored to further improve the classification performance.