ONLINE SHOPPERS PURCHASE INTENTION

Team I

Ingrid Lee

Lu Ho

Jack Lau

ABOUT THE DATASET

- **Online shopping** data
- Aim to identify **if a user will purchase or not**
- Consist of **18 features** collected through browser & website information
- Total **12,330 samples** (i.e. website sessions)
- Formed so that each session would belong to a different user in a **1-year period**

BUSINESS QUESTIONS

Conversion Performance

EDA

Identify the source of revenue from different aspects (E.g. customer, traffic type, region, seasonality)

Future Customer Prediction

Predictive Model

Revenue forecast based on web sessions

Part 1 Exploratory Data Analysis

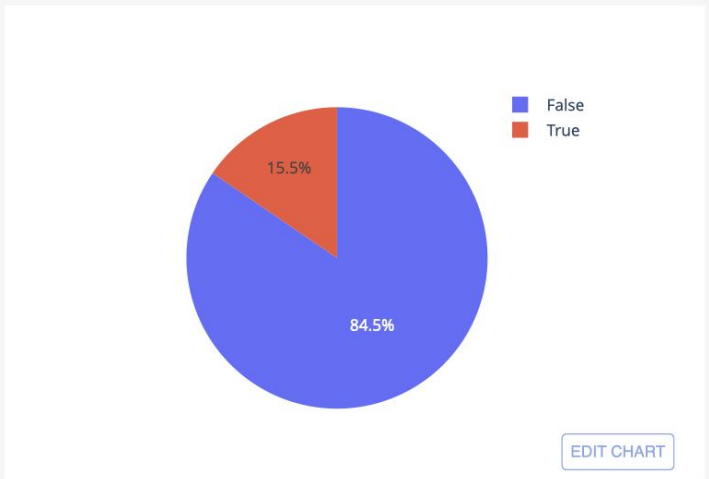
Get more understanding on the Online Business

REVENUE

Type: Bool

True: Purchase | False: Did not purchase

Sessions with Purchase Count



TARGET VARIABLE

Binary Classification Problem

A high conversion rate at 15.5%

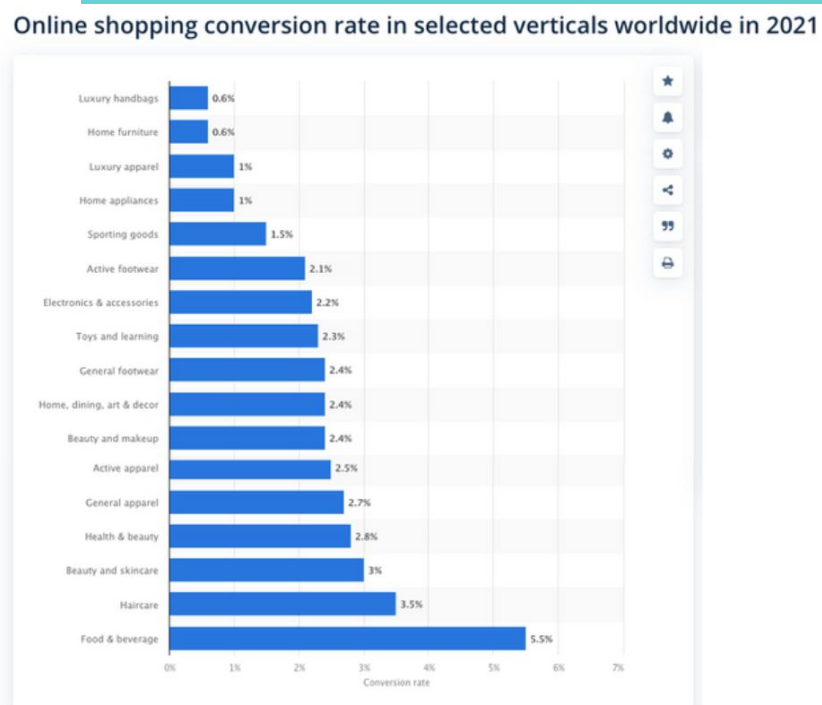
Assuming the dataset provides full data for the given period, 15.5% is very high versus market benchmark at 2.9-3.2% during 2018

Source: Statista - Conversion rate of online shoppers worldwide 2018-2020

Fact check on conversion rate across verticals

Concerning the vertical of the dataset is unknown, we have cross-checked if certain verticals would outperform others.

From the research, highest CR% is from F&B (5.5%). Therefore, we conclude the **CR% performance of given dataset is outstanding.**



Source: Statista - Conversion rate of online shoppers by verticals 2021

FEATURE VARIABLES

Categorical & Numerical

Session-based Variables

Page

1. **Page Type (3 kinds)**
   - **Administrative / Informational/ Product Related**

   No. of pages of corresponding type that the user visited.
   - **Administrative / Informational/ Product Related Duration**

   Amount of time spent in corresponding category of pages.
2. **Bounce**
   - **Bounce Rate:** % visitors who enter the website through that page and exit without triggering any additional tasks.
   - **Exit Rate:** % pageviews on the website that end at that specific page
3. **Page Value**
   - Avg. value of the page averaged over the value of the target page and/or the completion of an eCommerce transaction.

Device

- Operating Systems
- Browser

Date

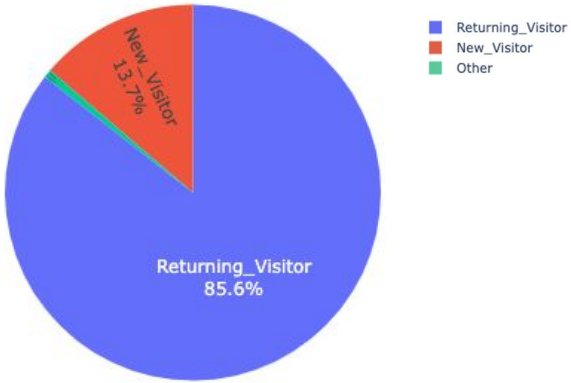
- Month
- Weekend
- Special Day

Others

- Traffic Type (Traffic Source)
- Visitor Type (New / Returning)
- Region

Visitors Type Analysis

All Sessions



Purchased Sessions

1 out of 3

New Visitors have purchased

1 out of 6

Returning Visitors have purchased

Key Takeaways

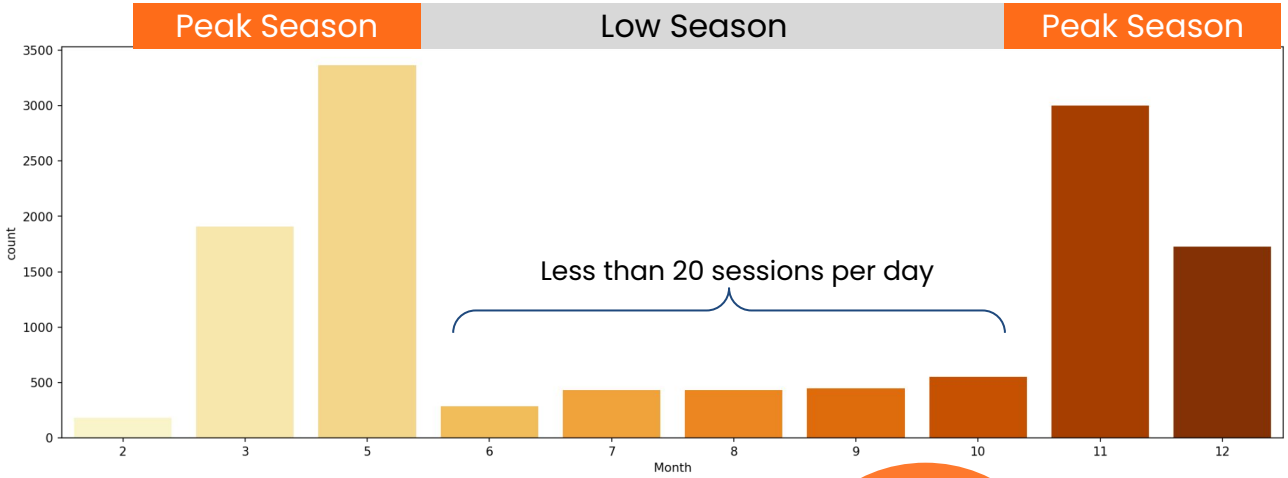
1. **Returning visitors** accounts for most of our traffic and transaction source.
2. It indicates that the business has **high customer loyalty & retention.**
3. New visitors are **likely to purchase** or of **high quality**.

Seasonality Analysis

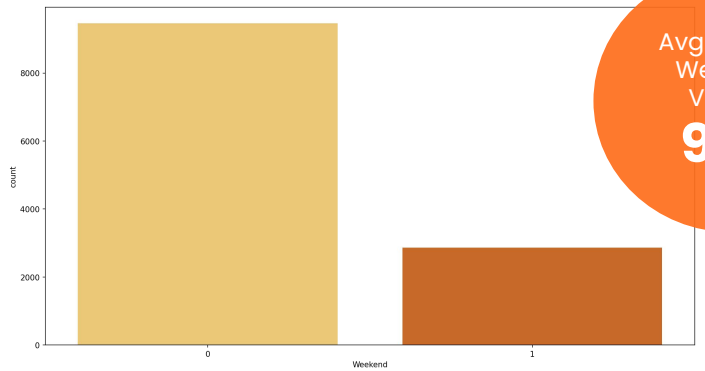
Any factors influencing their visiting in time?

Monthly Traffic Performance

Avg.Monthly Session:
1,233



Weekday VS Weekend



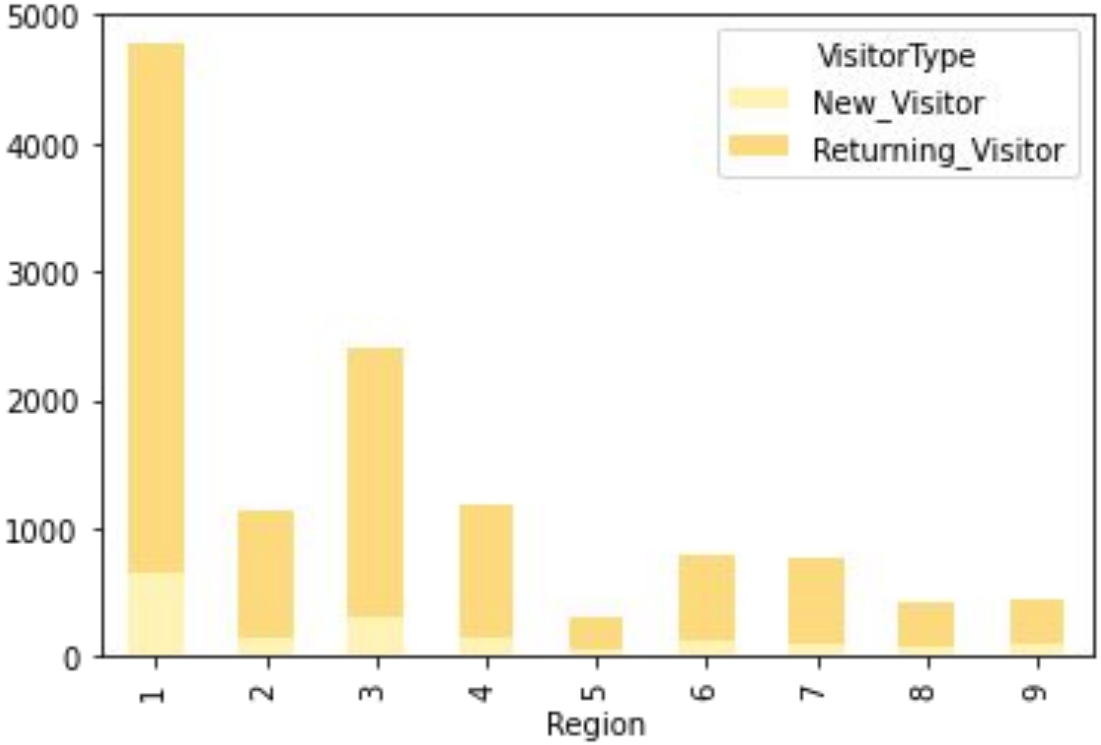
Avg.Monthly WeekDay Visitors
946

Avg.Monthly Weekend Visitors
287

*Data of April is missing in the raw data

Traffic Source Analysis (Region)

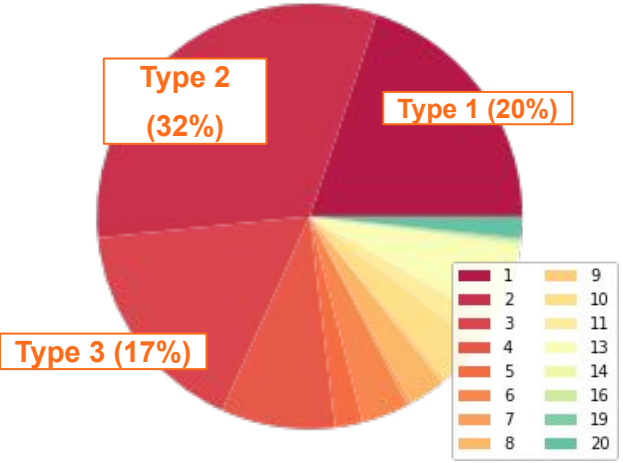
Where do visitors come from?

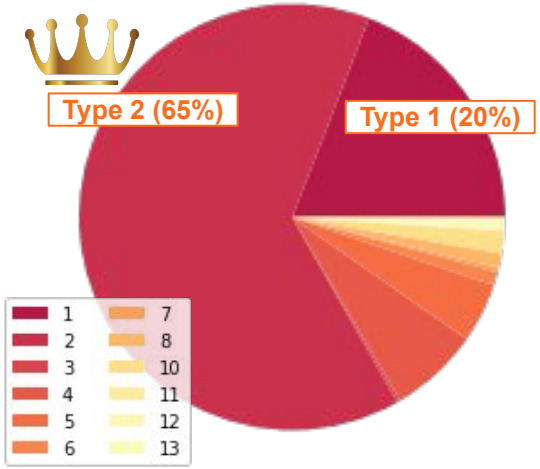


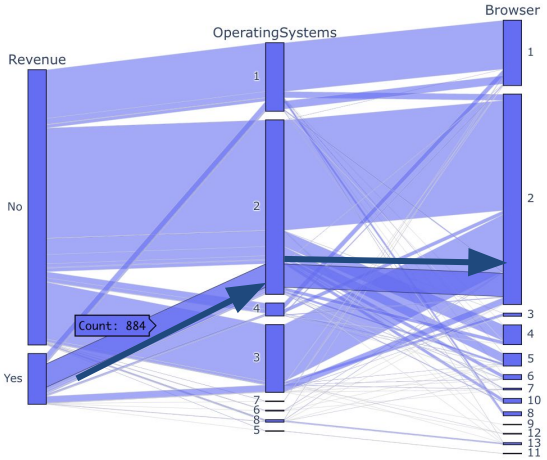
Region 1 & 3
drive the highest traffic, especially from returning visitors.

Traffic Source Analysis (Traffic Type & Browsers)

Where do visitors come from?







Traffic Type 1-3
drives the highest traffic and conversion among all channels.

Browser Type 2
drives the highest traffic and conversion.

OS 2 + Browser 2 Combination
drives the highest traffic and conversion.

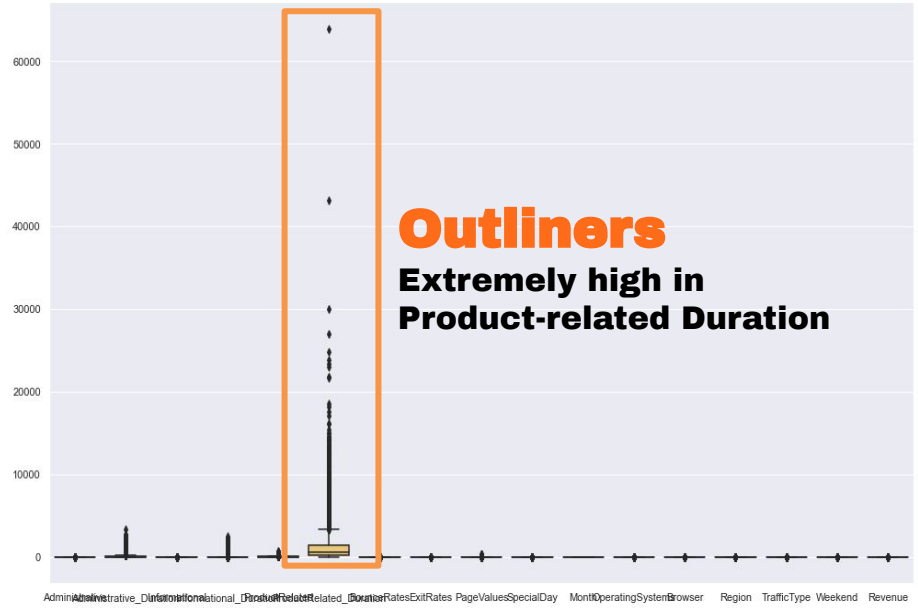
Page Analysis

21.8 mins

Avg. Session Duration

34.6

Avg Page/Session

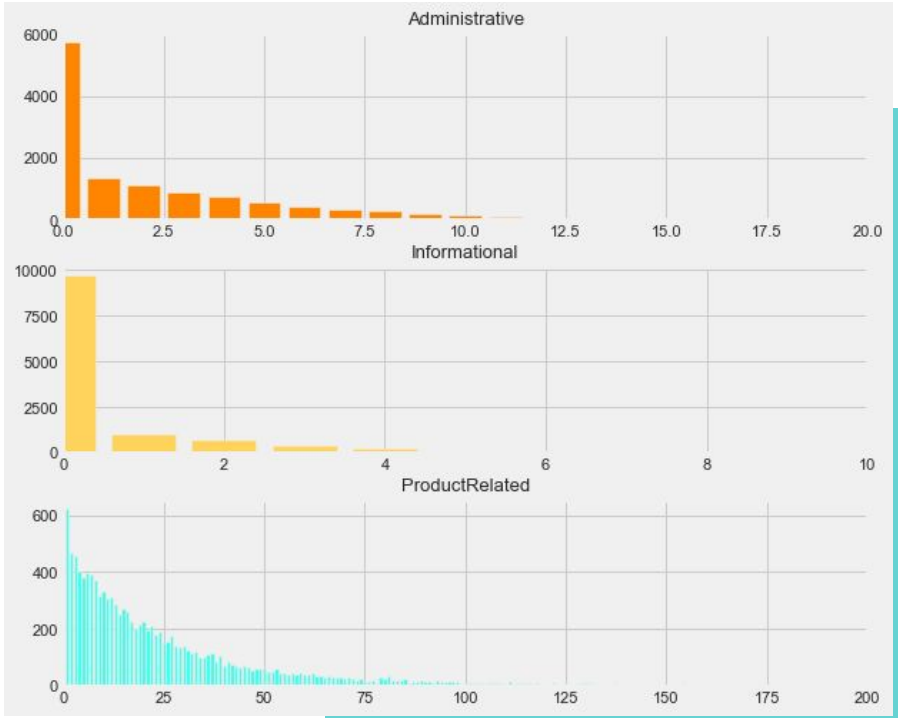


Outliners

Extremely high in Product-related Duration

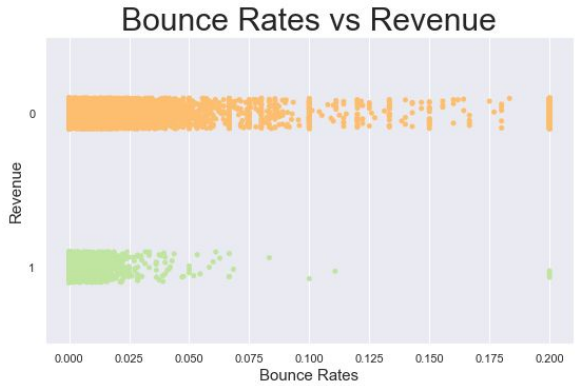
No. of pages visited

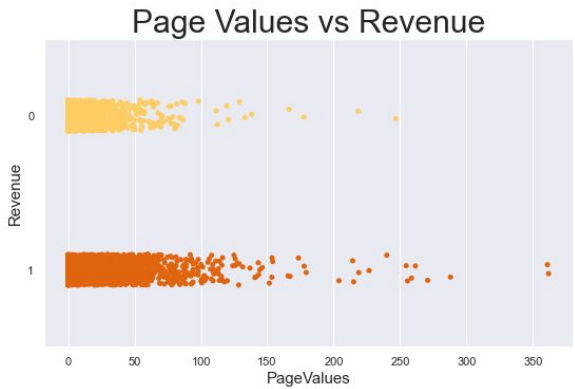
Product Related > Admin and Info pages



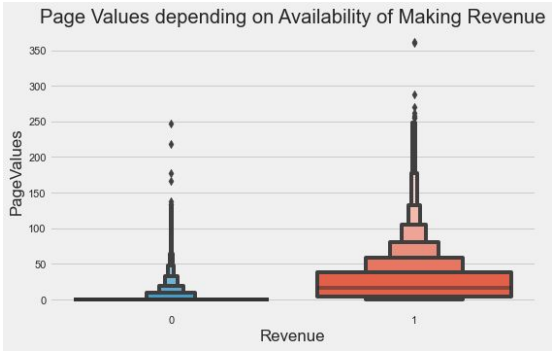
Page Analysis

Exit and Bounce Rates VS Revenue





Page Values VS Revenue



CONCERNS

Business Concern

Rely heavily on return visitors to generate sales revenue

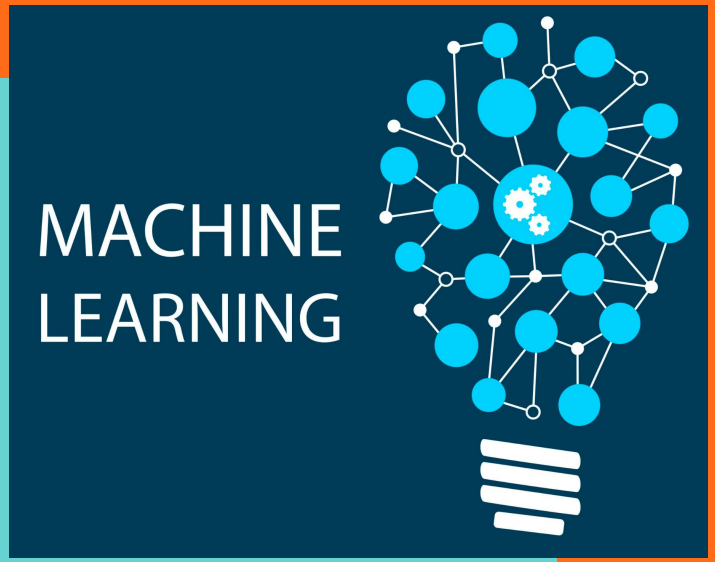
Centralized traffic types to drive website traffic

Modelling Concern

Weak correlation between page per session & session duration VS sales

High no. of outliers in page duration

Part 2
Machine Learning



Build Predictive Model on Revenue

Model Building Process

Data Preprocessing

- Missing Value
- Check Dtype
- Remove Outliers
- Encoding

Before Modelling

- PCA
- Feature Importance

Modelling

- Train Test Split
- Model Comparison

Fine Tune Model

- Sampling
- Model Comparison
- Hypertuning

Final Model

- Output
- Prediction
- Evaluation

Missing Values & Class Distribution

```
In [8]: df.isnull().sum()

Out[8]: Administrative             0
        Administrative_Duration    0
        Informational              0
        Informational_Duration     0
        ProductRelated             0
        ProductRelated_Duration    0
        BounceRates                0
        ExitRates                  0
        PageValues                 0
        SpecialDay                 0
        Month                      0
        OperatingSystems           0
        Browser                    0
        Region                     0
        TrafficType                0
        VisitorType                0
        Weekend                    0
        Revenue                    0
        dtype: int64
```

No missing value from the dataset

```
In [18]: df['Revenue'].value_counts()

Out[18]: 0    10422
         1     1908
         Name: Revenue, dtype: int64

In [91]: df['Revenue'].value_counts()/len(df)*100

Out[91]: 0    84.525547
         1    15.474453
         Name: Revenue, dtype: float64
```

Expected outcome: ‘0’ or ‘1’

Highly imbalanced data

Data Cleaning

```
<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
```

```
df['Revenue'] = df['Revenue'].astype(int) #clean data type: bool to int
```

```
df['Weekend'] = df['Weekend'].astype(int) #clean data type: bool to int
```

Update Boolean dtype to integers

	Administrative	Administrative_Duration	Informational	Informational_Duration	ProductRelated	ProductRelated_Duration
count	12330.000000	12330.000000	12330.000000	12330.000000	12330.000000	12330.000000
mean	2.315166	80.818611	0.503569	34.472398	31.731468	1194.746220
std	3.321784	176.779107	1.270156	140.749294	44.475503	1913.669288
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000	0.000000	7.000000	184.137500
50%	1.000000	7.500000	0.000000	0.000000	18.000000	598.936905
75%	4.000000	93.256250	0.000000	0.000000	38.000000	1464.157214
max	27.000000	3398.750000	24.000000	2549.375000	705.000000	63973.522230

	Administrative	Administrative_Duration	Informational	Informational_Duration	ProductRelated	ProductRelated_Duration
count	10467.000000	10467.000000	10467.000000	10467.000000	10467.000000	10467.000000
mean	1.604662	36.259147	0.337346	20.959281	21.961116	741.160616
std	2.442930	56.192433	0.976302	102.611044	22.490491	771.708313
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	0.000000	0.000000	6.000000	144.500000
50%	0.000000	0.000000	0.000000	0.000000	15.000000	471.400000
75%	3.000000	58.033333	0.000000	0.000000	30.000000	1101.568333
max	19.000000	233.083333	16.000000	2252.033333	223.000000	3382.287999

Remove Admin Duration & Product-related Duration Outliers

Encoding

Method 1: Convert nominal variables by Map()

```
 #   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  int64
 17  Revenue                  12330 non-null  int64
dtypes: float64(7), int64(9), object(2)
```

Both 'Month' and 'Visitor Type' are nominal data.

Total 18 columns

```
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Administrative           10467 non-null  int64
 1   Administrative_Duration  10467 non-null  float64
 2   Informational            10467 non-null  int64
 3   Informational_Duration   10467 non-null  float64
 4   ProductRelated           10467 non-null  int64
 5   ProductRelated_Duration  10467 non-null  float64
 6   BounceRates              10467 non-null  float64
 7   ExitRates                10467 non-null  float64
 8   PageValues               10467 non-null  float64
 9   SpecialDay               10467 non-null  float64
 10  Month                    10467 non-null  int64
 11  OperatingSystems         10467 non-null  int64
 12  Browser                  10467 non-null  int64
 13  Region                   10467 non-null  int64
 14  TrafficType              10467 non-null  int64
 15  VisitorType              10467 non-null  int64
 16  Weekend                  10467 non-null  int64
 17  Revenue                  10467 non-null  int64
```

```
df['Month'].map({'Feb':2, 'Mar':3,'May':5, 'June':6, 'Jul':7, 'Aug':8, 'Sep':9, 'Oct':10, 'Nov':11, 'Dec':12})
df['VisitorType'].map({'Returning_Visitor':2, 'New_Visitor':1, 'Other':0})
```

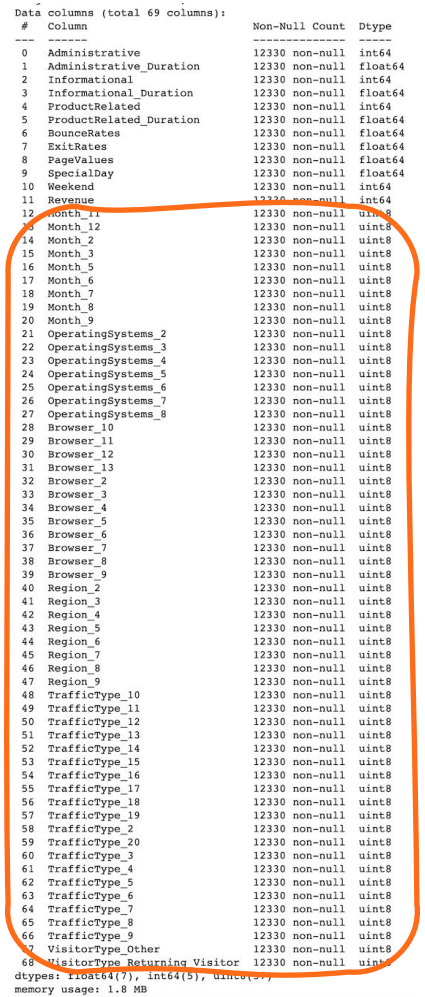
Encoding

Method 2: Dummies Encoding

#	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	int64
17	Revenue	12330 non-null	int64

dtypes: float64(7), int64(9), object(2)

Transforms the categorical variables into a set of binary variables through dummy encoding.



Total 69 columns

Train Test & Split

80% Train, 20% Test

```
X = df.drop(columns='Revenue', axis=1)
y = df['Revenue']
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,train_size=0.8,random_state=1)
print("Input Training:",X_train.shape)
print("Input Test:",X_test.shape)
print("Output Training:",y_train.shape)
print("Output Test:",y_test.shape)
```

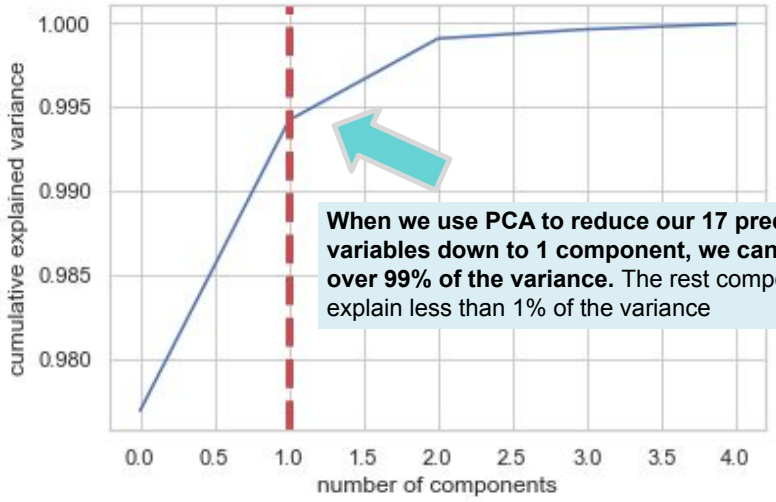
```
Input Training: (8373, 17)
Input Test: (2094, 17)
Output Training: (8373,)
Output Test: (2094,)
```

Principal Component Analysis (PCA)

- See if dimension reduction is necessary
  - Helps speed up training model
  - PCA(n_components=5)
  - Result: PC-1 = 99%

Insights

- More useful when the datasets have huge number of columns
- Next Step: No dimension reduction



When we use PCA to reduce our 17 predicting variables down to 1 component, we can explain over 99% of the variance. The rest components explain less than 1% of the variance

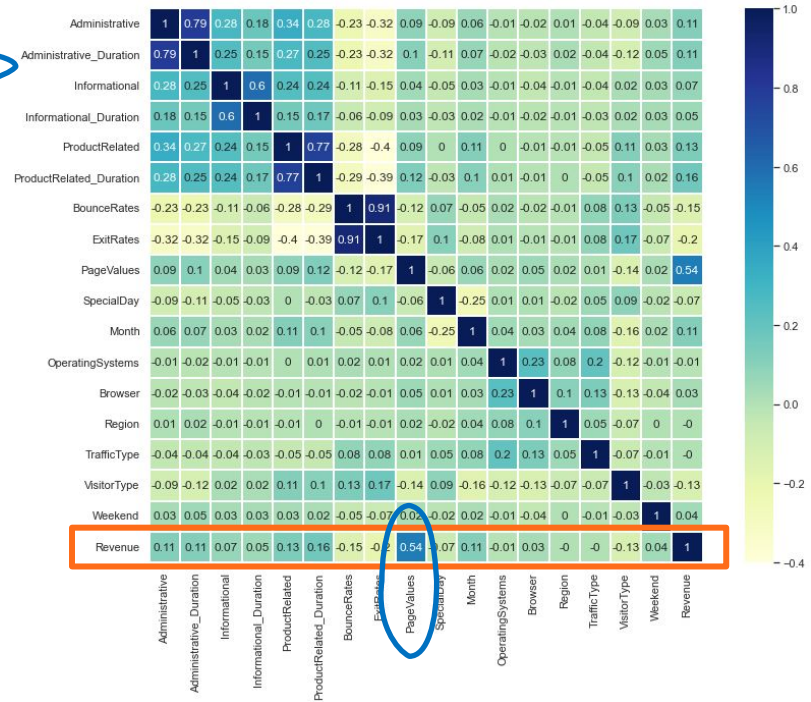
pca_test.explained_variance_ratio_
array([9.76892057e-01, 1.72962520e-02, 4.86249650e-03, 5.49929960e-04, 3.32771367e-04])

Feature Importance

- Understand the relative importance of each feature when making a prediction
- Fitting an XGBClassifier
- Page Values is the key component

	Feature	Importance
8	PageValues	0.354704
10	Month	0.086715
15	VisitorType	0.063573
6	BounceRates	0.060466
0	Administrative	0.049524
7	ExitRates	0.040472
2	Informational	0.035623
4	ProductRelated	0.035499
1	Administrative_Duration	0.035209
5	ProductRelated_Duration	0.034034
16	Weekend	0.031666
14	TrafficType	0.030828
9	SpecialDay	0.030021
3	Informational_Duration	0.029965
12	Browser	0.028707
13	Region	0.026991
11	OperatingSystems	0.026004

Summary of Calculated Feature Importance Scores



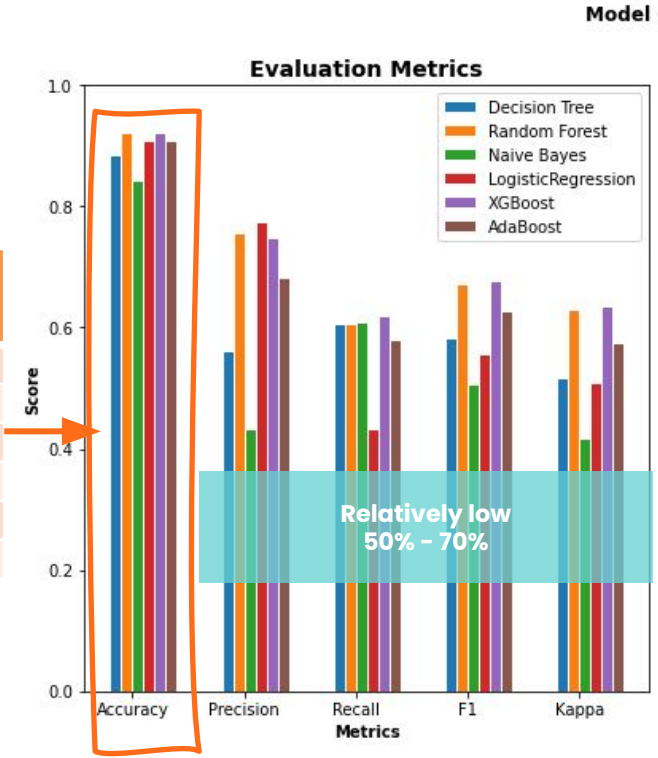
Correlation Heatmap

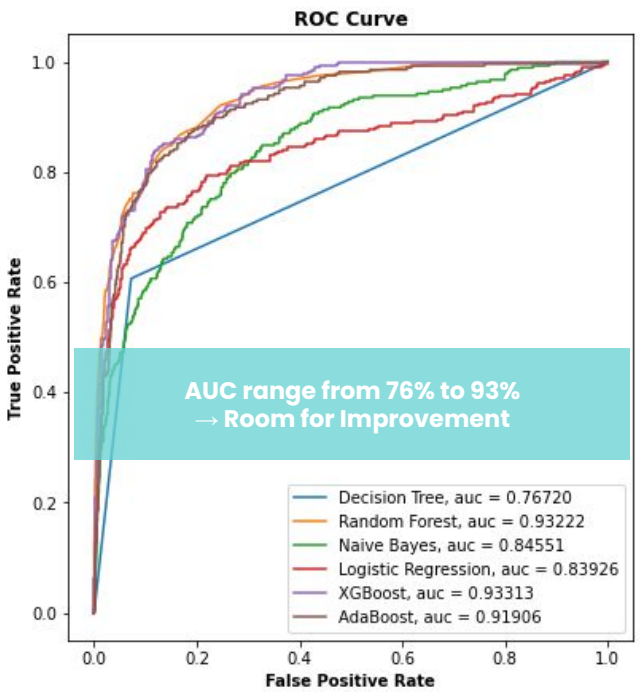
Classification Models

- Decision Trees
- Random Forest
- Naive Bayes
- Logistic Regression
- XGBoost
- AdaBoost

Initial Results - Scores

Classification Model	Accuracy Score
Decision Tree	0.8854
Random Forest	0.9221
Naïve Bayes	0.8429
Logistic Regression	0.9083
XGBoost	**0.9222**
AdaBoost	0.9088





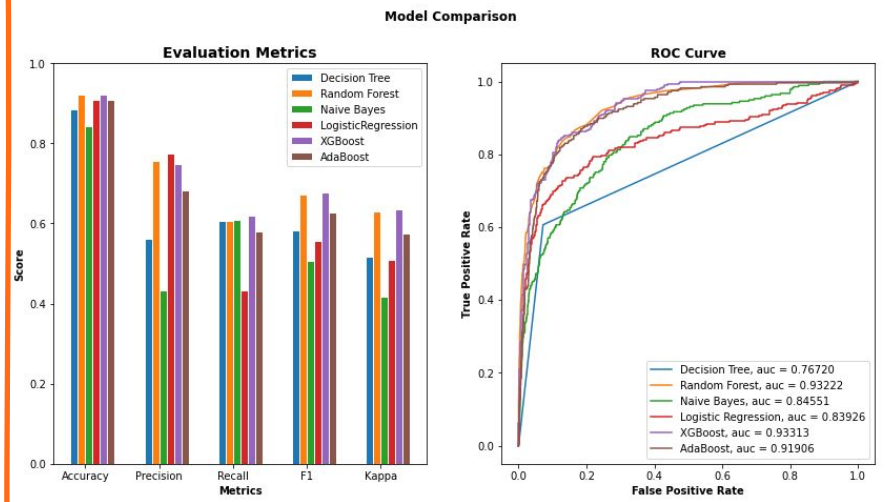
Initial Results – Confusion Matrix

TN	**FP**
FN	**TP**

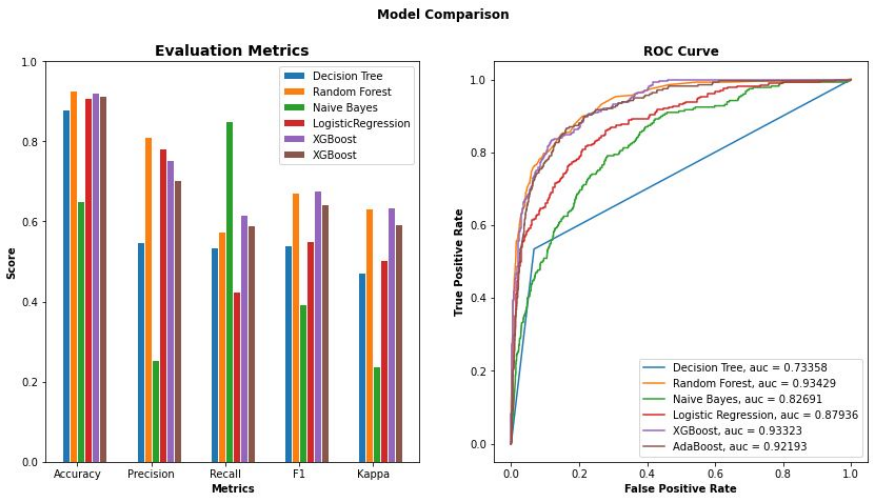
Initial Results

(Compare with Dummy Encoding)

Method 1: Convert nominal variables by Map()



Method 2: Dummies Encoding



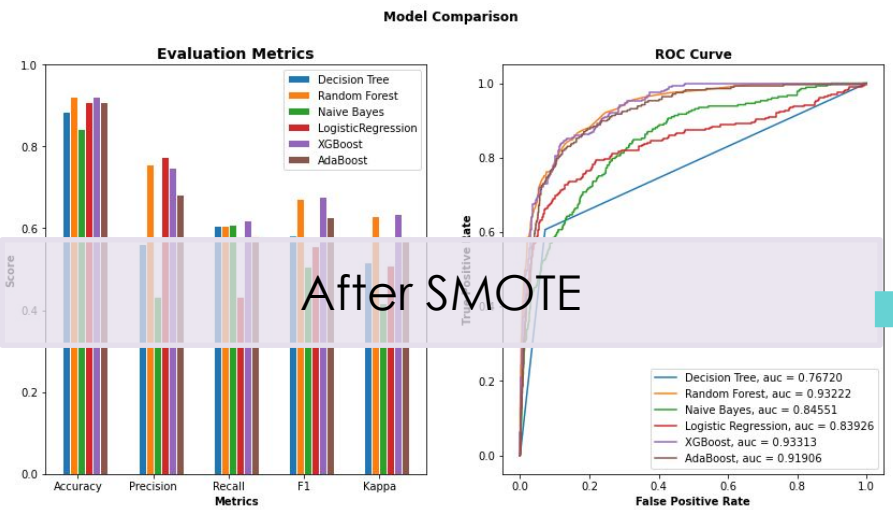
- More fluctuation on results
- Difference is not significant
- Take method 1

SMOTE

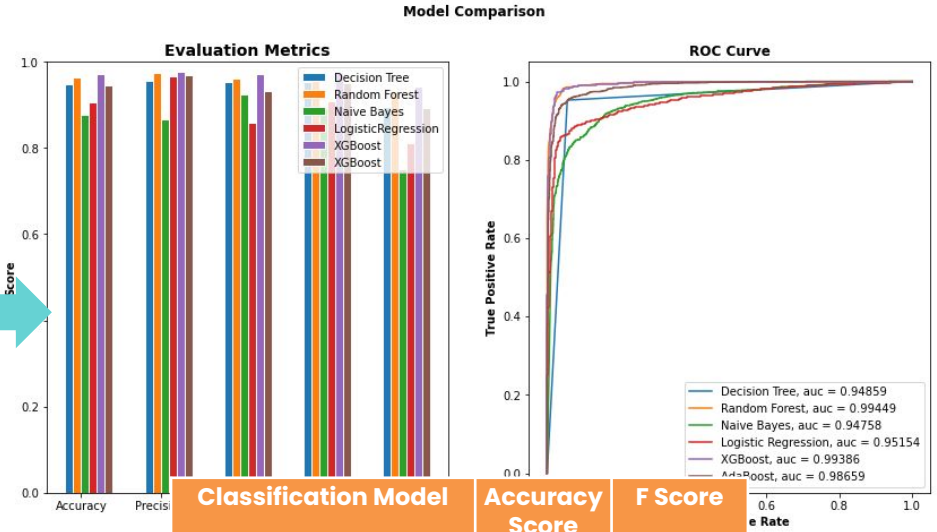
- Transform imbalanced data (Purchase vs No Purchase)
- The most commonly used method
- Method: Oversampling the minority class

```
In [48]: from imblearn.combine import SMOTEENN

In [49]: sm = SMOTEENN()
         X_resampled1, y_resampled1 = sm.fit_resample(X,y)

In [50]: X_train, X_test, y_train, y_test=train_test_split(X_resampled1, y_resampled1,train_size=0.8,random_state=1)
```

Results After SMOTE



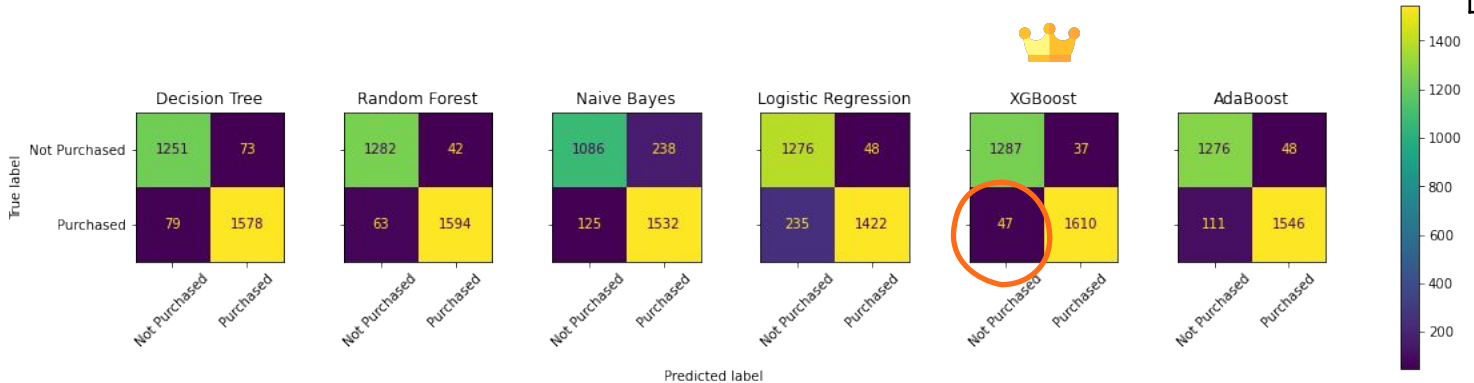
After SMOTE



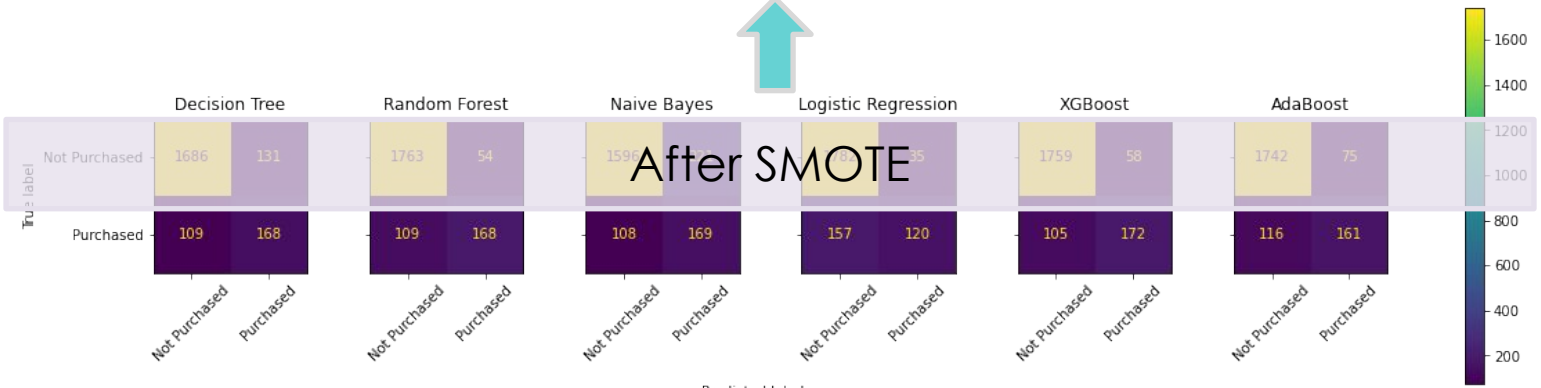
Classification Model	Accuracy Score	F Score
Decision Tree	0.9490	0.9540
Random Forest	0.9647	0.9681
Naïve Bayes	0.8782	0.8940
Logistic Regression	0.9050	0.9094
XGBoost	**0.9718**	**0.9745**
AdaBoost	0.9466	0.9510

Results After SMOTE

TN	FP
FN	TP



After SMOTE



Hypertuning with Randomized SearchCV

```
## Hyper Parameter Optimization

params={"learning_rate"    : [0.05, 0.10, 0.15, 0.20, 0.25, 0.30 ] ,
 "max_depth"        : [ 3, 4, 5, 6, 8, 10, 12, 15],
 "min_child_weight" : [ 1, 3, 5, 7 ],
 "gamma"            : [ 0.0, 0.1, 0.2 , 0.3, 0.4 ],
 "colsample_bytree" : [ 0.3, 0.4, 0.5 , 0.7 ] }
```

```
random_search=RandomizedSearchCV(classifier_smote_hpo,param_distributions=params,
                                 n_iter=5,scoring='roc_auc',n_jobs=-1,cv=20,verbose=3)

from datetime import datetime

start_time = timer(None)
random_search.fit(X_resampled1, y_resampled1)
timer(start_time)
```

- Hypertuning XGBoost
- Fitting 20 folds for each of 5 candidates
- Total 100 fits

Results of Best Estimator & Parameters

```
In [84]: random_search.best_estimator_

Out[84]: XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
                       colsample_bynode=1, colsample_bytree=0.5,
                       enable_categorical=False, gamma=0.1, gpu_id=-1,
                       importance_type=None, interaction_constraints='',
                       learning_rate=0.25, max_delta_step=0, max_depth=15,
                       min_child_weight=1, missing=nan, monotone_constraints='()',
                       n_estimators=100, n_jobs=8, num_parallel_tree=1, predictor='auto',
                       random_state=0, reg_alpha=0, reg_lambda=1, scale_pos_weight=1,
                       subsample=1, tree_method='exact', validate_parameters=1,
                       verbosity=None)

In [85]: random_search.best_params_

Out[85]: {'min_child_weight': 1,
          'max_depth': 15,
          'learning_rate': 0.25,
          'gamma': 0.1,
          'colsample_bytree': 0.5}
```

Cross Validating Score

```
[0.913085   0.91786055 0.9417383  0.90639924 0.90830946 0.9025788
 0.8739255  0.90630975 0.91108987 0.90917782]
```

Final Result & Prediction

XGBoost

BEFORE

Accuracy: 0.9718215363971822

Confusion Matrix:
[[1287 37]
[47 1610]]

AFTER

Accuracy: 0.9721569942972157

Confusion Matrix:
[[1282 42]
[41 1616]]

Accuracy score has slightly improved and False Negative reduced.

Final Classification Report

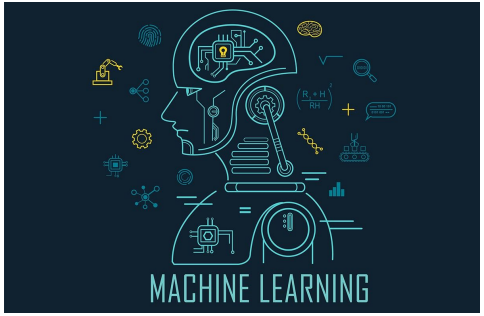
```
Confusion Matrix:
[[1282   42]
 [  41 1616]]
Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.97      0.97      1324
           1       0.97      0.98      0.97      1657

    accuracy                           0.97      2981
   macro avg       0.97      0.97      0.97      2981
weighted avg       0.97      0.97      0.97      2981

Accuracy: 0.9721569942972157
```

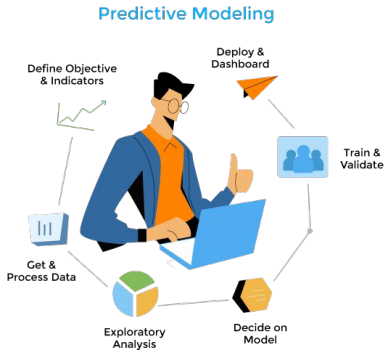
Prediction

```
In [91]: print(y_pred_new[0:10])
         [1 1 1 1 1 0 0 0 1 1]

In [92]: print(y_test[0:10])
         11055    1
         7328     1
         7219     1
         11698    1
         11490    1
         5321     0
         5012     0
         6530     0
         10152    1
         7939     1
         Name: Revenue, dtype: int64
```

Model Evaluation & Conclusion

- A high accuracy and precision for the Model built
- Pros for using XGBoost:
  - Less feature engineer required (No feature scaling made)
  - Outliers have minimal impact
  - Good model performance





Limitations:

- Dataset limited by 1 year only
- April data is absent
- Result is not that much different after optimising by *Randomized Search CV*, potentially hit the limit with this model
- Explore other algorithm such as `LGBMClassifier` & `BaggingClassifier`

Recommendation

What's next?

Prospecting New Customers

1. Expand **customer base**
2. Increasing **traffic and sales volume**
3. Provide **more comprehensive data samples** for model accurate prediction

Find new quality customers from traffic channel (e.g. Type 4,8 & 20) which has proven higher conversion rate

Optimize any paid or marketing campaign by the identified target segments, highly converting OS & region users.

Promotion strategy during peak season

THANK YOU

STEPS

1. FAMILIARIZE VARIABLES
2. UNDERSTAND DATASET & DATA CLEANING
3. EDA ANALYSIS & RECOMMENDATION (maybe some clustering as well)
4. ENCODING & SAMPLING
5. FEATURE SCALING
6. MODEL BUILDING & HYPERTUNING

GOAL

1

Understand **visitor characteristics** & corresponding **web behavior** to provide recommendations through EDA

2

Build **purchase prediction model** for purchase intention forecast

RECOMMENDATION 1

Prospecting new customers

- To sustain the business and strengthen customer base, finding new customers is essential at current stage
- It helps to increase web traffic & sales volume
- The company can further source new quality users from Traffic Type 4, 8 & 20 as indicated below that conversion rate is higher.

TrafficType	Revenue 0	1	conversion rate
1	30.0	7.0	18.918919
2	698.0	238.0	25.427350
3	115.0	25.0	17.857143
4	66.0	33.0	33.333333
5	110.0	39.0	26.174497
6	31.0	11.0	26.190476
7	4.0	1.0	20.000000
8	125.0	50.0	28.571429
9	9.0	NaN	NaN
10	17.0	5.0	22.727273
11	45.0	6.0	11.764706
13	6.0	NaN	NaN
14	1.0	NaN	NaN
15	2.0	NaN	NaN
16	1.0	NaN	NaN
18	1.0	NaN	NaN
19	1.0	NaN	NaN
20	10.0	7.0	41.176471

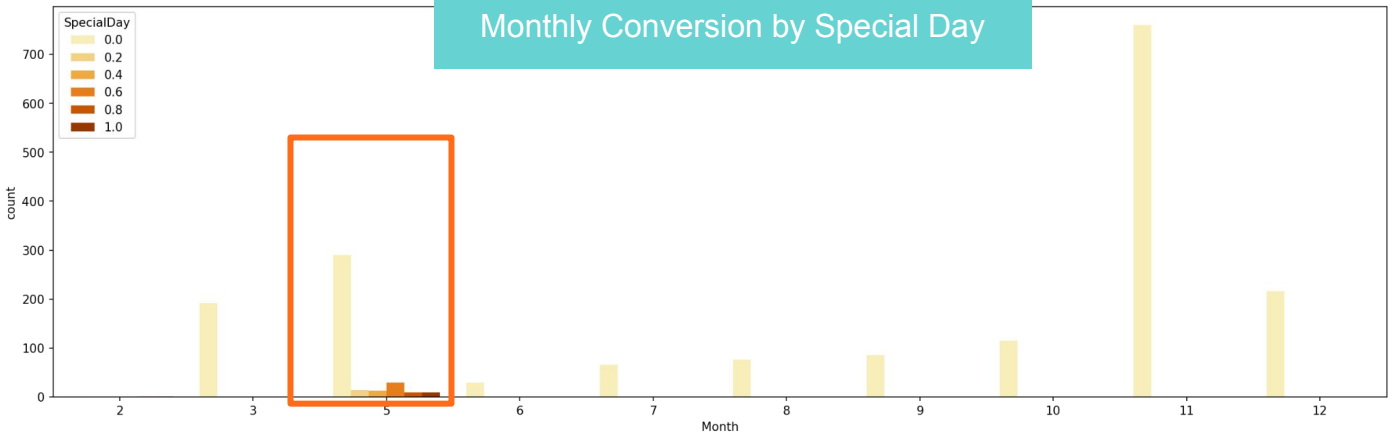
Traffic source of new visitors

Monthly Sale Performance

Monthly Conversion (Purchase & Non-Purchase)

Avg.Monthly Conversion:

Monthly Conversion by Special Day



>96% Conversion come from non-Special Day

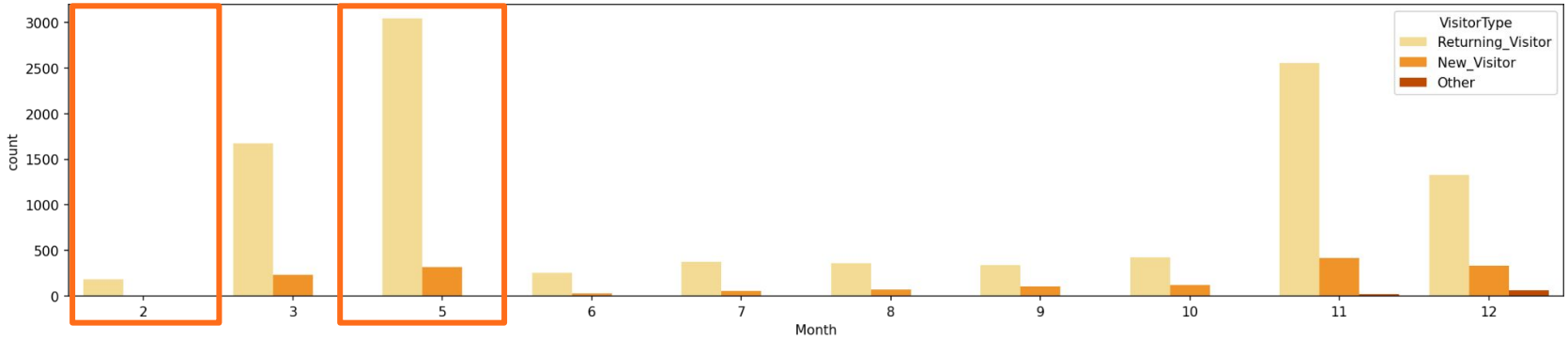
*Data of April is missing in the raw data

"Special Day" : indicates the closeness of the site visiting time to a specific special day (e.g. Mother's Day, Valentine's Day) in which the sessions are more likely to be finalized with transaction. Determined by considering the dynamics of e-commerce(e.g.the duration between the order date and delivery date)

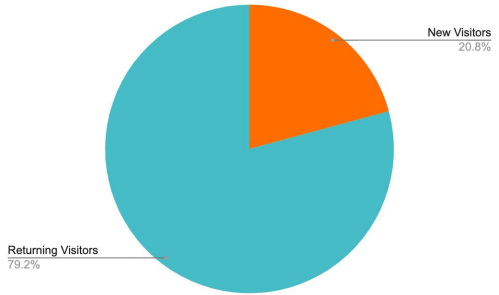
INSIGHT 2 – Seasonality

How seasonal/festive factors influence **leads generation of new customers**

New VS Returning Visitor



Special Day Conversion by Customers



20% of conversion (16) by Special Day come from new visitors.

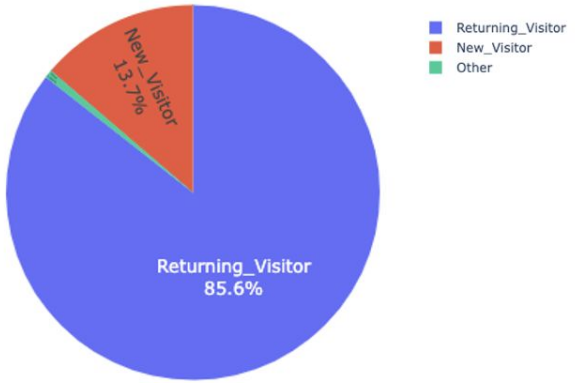
"Special Day" : indicates the closeness of the site visiting time to a specific special day (e.g. Mother's Day, Valentine's Day) in which the sessions are more likely to be finalized with transaction. Determined by considering the dynamics of e-commerce(e.g.the duration between the order date and delivery date)

RECOMMENDATION 2

- Budget re-allocate on highly converted channels
- Optimize unconverted traffic channels conversion
- Explore channels for driving new customers
- Optimization of technical development on Browser 2 which over 60% of traffic is browsed by.

Visitors Type

All Sessions



Purchased Sessions

1 out of 3

New Visitors have purchased

1 out of 6

Returning Visitors have purchased

Key Takeaways

1. Returning visitors accounts for most of our traffic and transaction source.
2. It indicates that the business has high customer loyalty & retention.
3. New visitors are likely to purchase or of high quality.

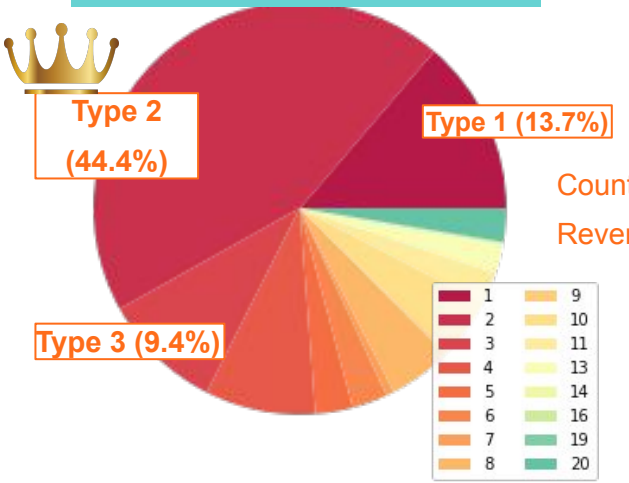
Potential Business Concerns

1. Revenue source relies on same group of customer, whereas new visitors are minority.
2. It might be hard for business to scale up sales volume given loyal customer may have limited buying power.
3. Buying potential can be observed from new visitors

INSIGHT 4 - Traffic Acquisition

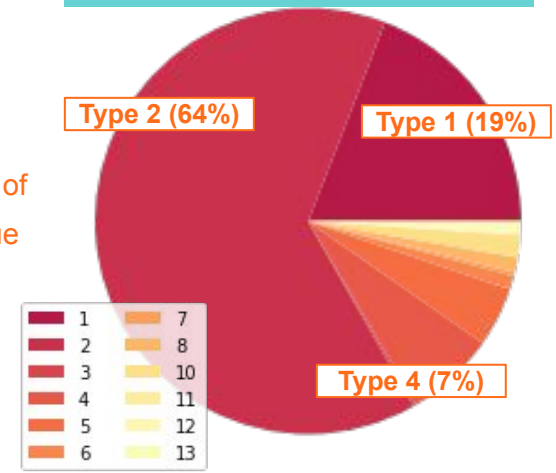
By Traffic Type & Browsers Type

Most Converted Traffic Types



Counts of Revenue

Most Converted Browsers



Traffic mainly drives from 3 major traffic types (over 60%) for new and returning visitors

Browser Type 2 brings highest conversions among different traffic types.

No significant difference between traffic types on driving certain types of visitors.

Recommendation

Prospecting New Customers

1. **Expand new customer base to sustain business growth by increasing web traffic and sales volume**

2. **Provide more data samples for modelling prediction**

Explore strategy on traffic channel with high CVR but low traffic (e.g. Type 4,8,20)

Strategy on audience segmentation and targeting (e.g location, device) in paid marketing activities

New customer promotion on peak seasons