Online Shoppers Purchasing Intention

Victory (Victor) Ma

Goal: Predict whether an online customer will generate revenue based on various features related to their time on the website

Objective: Using results from the model, we can then fine-tune the website to maximize profit for the business

(maximizing shareholder profit is definitely the goal in life)

Opening, Challenge, Action, & Resolution

Opening: We are using the *Online Shoppers Purchasing Intention Dataset* from UC Irvine's Machine Learning Repository, which has 18 variables

Challenge: Can we accurately predict if a customer will purchase something during their time on the website?

Action: We will follow the essential workflow outlined by CSCI 200B in order to create an effective model for the company to use

Resolution: In the end, our tuned **Random Forest Classifier** model had high accuracy, with **90.67%**. Our analysis concluded that the most important factor was **"Page Values"**, which represents the average "value" for a web page that a user visited before completing an e-commerce transaction

But how did we get to that resolution?

Description of the Dataset

This particular dataset from **UC Irvine** gives us 10 numerical variables, 7 categorical variables, and 1 target variable

- Features such as **"Informational", "Informational Duration", "Product Related"** and **"Product Related Duration"** represent the number of different types of pages visited by the visitor in that session and total time spent in each of these page categories

- The value of **"Bounce Rate"** refers to the percentage of customers who enter the site from that page and then leave without triggering any other requests

```
Column data types:
Administrative               int64
Administrative_Duration    float64
Informational                int64
Informational_Duration     float64
ProductRelated               int64
ProductRelated_Duration    float64
BounceRates                float64
ExitRates                  float64
PageValues                 float64
SpecialDay                 float64
Month                       object
OperatingSystems             int64
Browser                      int64
Region                       int64
TrafficType                  int64
VisitorType                 object
Weekend                       bool
Revenue                       bool
dtype: object
```

- The value of **"Exit Rate"** is calculated as for all pageviews to the page, the percentage that were the last in the session

- The target variable is named **"Revenue"**, indicating whether or not a purchase was made during the session

```
Revenue
False    84.525547
True     15.474453
Name: proportion, dtype: float64
```

```
EDA: Statistical measures:
       Administrative  Administrative_Duration  ...        Region   TrafficType
count    12330.000000             12330.000000  ...  12330.000000  12330.000000
mean         2.315166                80.818611  ...      3.147364      4.069586
std          3.321784               176.779107  ...      2.401591      4.025169
min          0.000000                 0.000000  ...      1.000000      1.000000
25%          0.000000                 0.000000  ...      1.000000      2.000000
50%          1.000000                 7.500000  ...      3.000000      2.000000
75%          4.000000                93.256250  ...      4.000000      4.000000
max         27.000000              3398.750000  ...      9.000000     20.000000
```

```
Summary statistics for categorical features:
        Month        VisitorType Weekend Revenue
count   12330              12330   12330   12330
unique     10                  3       2       2
top       May  Returning_Visitor   False   False
freq     3364              10551    9462   10422
```

Exploring The Data

Step 1: Check Data

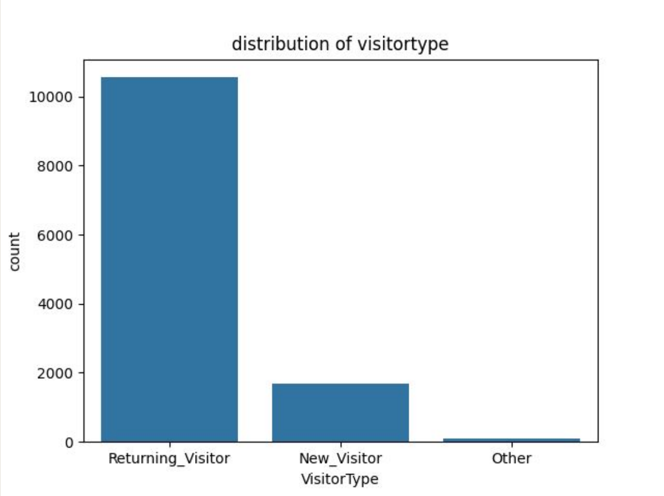
- Reviewed variable types and checked for missing values
- Noted that “Revenue” is a binary class, 1 indicating a purchase and 0 indicating no purchase
- Initial analysis showcased that…

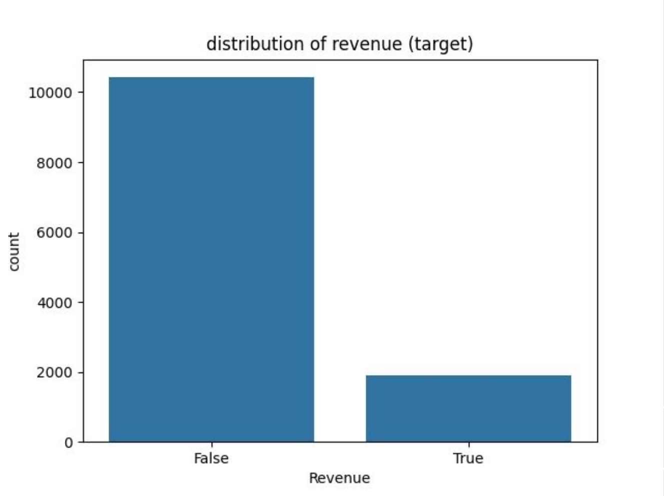
Step 2: Visualizations

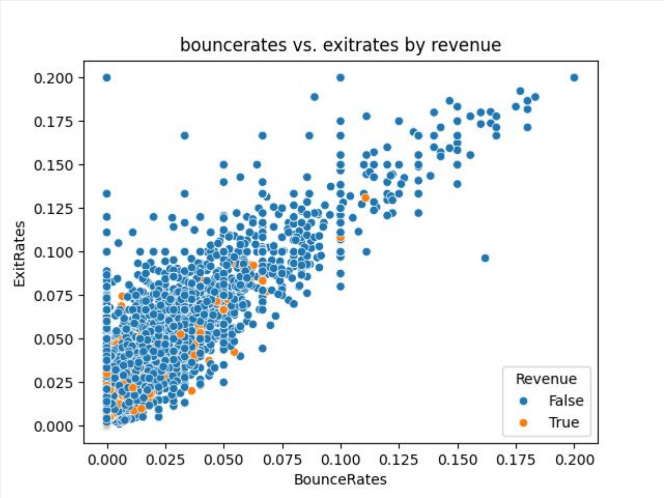
- Generated basic bar plots, scatterplots, histograms, etc. to understand the features and relationships much better

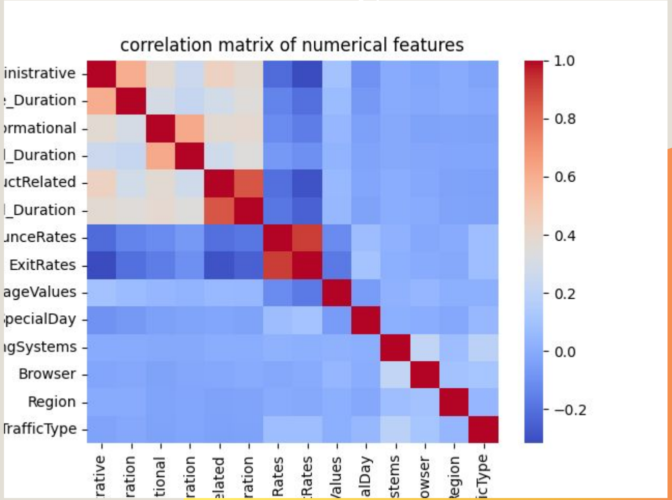
Step 3: Statistical Tests/Measures

- Conducted statistical tests (ANOVA, T-Tests, etc.) to identify key relationships between features and the target


distribution of visitortype
count
0
2000
4000
6000
8000
10000
Returning_Visitor
New_Visitor
Other
VisitorType


distribution of revenue (target)
count
0
2000
4000
6000
8000
10000
False
True
Revenue


bouncerates vs. exitrates by revenue
ExitRates
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
0.200
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
0.200
BounceRates
Revenue
False
True


correlation matrix of numerical features
inistrative
_Duration
ormational
_Duration
uctRelated
_Duration
unceRates
ExitRates
ageValues
pecialDay
ngSystems
Browser
Region
TrafficType
1.0
0.8
0.6
0.4
0.2
0.0
−0.2

Data Cleaning & Wrangling

- Confirmed that there are no missing values in the columns
- Checked for duplicates/strange values (i.e. NA, None, etc.)
- Checked for outliers or problematic ranges of values
- OHE (One-Hot Encoded) categorical features such as "Weekend" and "Revenue"
- No other features need transformation, which is great
- Based on descriptions and context, we will select 8-10 main features to include

Selecting Our Models

Our learning approach is Supervised Learning for Classification, because we have a defined target variable—Revenue—indicating a class (purchase or no purchase)

Models That Were Chosen:

Null Model: Assigning all observations to the majority class (in this case, no revenue). This will be our baseline - **Accuracy: 84.5%**

Logistic Regression: Utilized for its simplicity and interpretability our problem, because its binary classification - **Accuracy: 88.37%**

Random Forest: Utilized for its potential to capture more complex relationships in the data for classification tasks - **Accuracy: 90.67%**

Training Our Models

1

Data Split

We split our data into training (70%) and testing (30%), stratified by ‘Revenue’

2

Tuning Hyperparameters

We used GridSearchCV with 5-fold stratified cross-validation to find optimal parameters based on ROC-AUC

3

Fitting The Model

We trained the Null model, best Logistic Regression model, and best Random Forest model on the full training data

4

Measuring Performance

We finally evaluated the final models on the unseen data (hold-out set)

Evaluating Our Models

1

Calculate Performance Metrics

For each model, we calculated **Accuracy, Precision, Recall, F1,** and **ROC-AUC**

2

Compare Models

Our **Random Forest Classifier** model outperformed the other two models we chose by a fair bit

3

Analyze Confusion Matrices

Our best model's confusion matrix:
[3035 92]
[253 319] ... We correctly identified revenue 319 times, but missed out on 253 potential customers

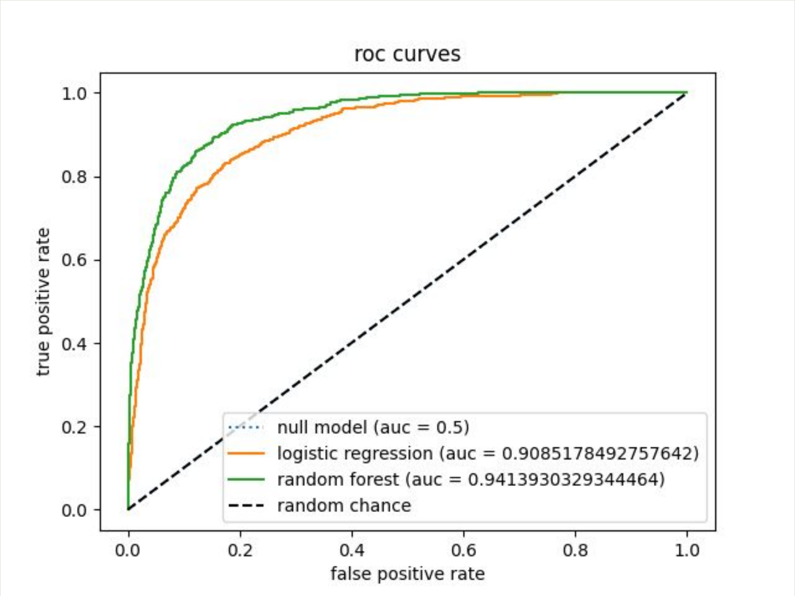
4

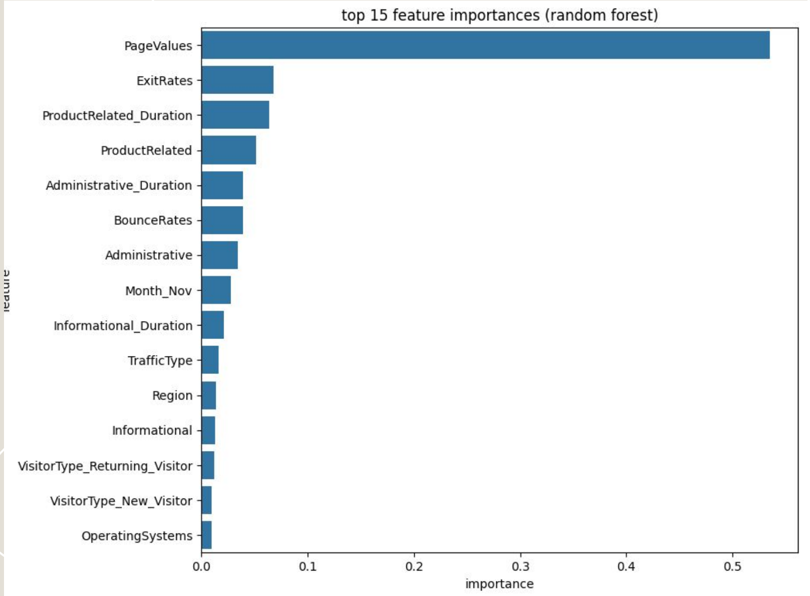
Generate Model Insight

Our **"Page Values"** feature was most important when predicting revenue. **"Exit Rates", "Bounce Rates"**, and **"Product Related"** were also important!

Overall Insights & Concluding Ideas!

Null Model	0.8454	0.0000	0.0000	0.0000	0.5000
Logistic Regression	0.8838	0.7518	0.3706	0.4965	0.9085
Random Forest	0.9067	0.7762	0.5577	0.6490	0.9414





Any questions?

Thank you for helping me maximize shareholder profit. I am sure that they will be happy.