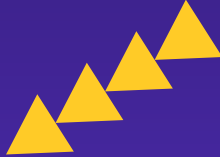




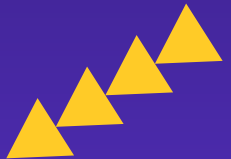
Online Shoppers Purchasing Intention



Background Story



As a Data Scientist team at e-commerce company at PT Anaconda, responsible for analyze data related to current visitor behavior and potential improvements purchase amount



Meet with **our team**



Muhammad **Irfan**
Fadlurahman



Ramado Dipradelana



Bima Purnomo
Sandi



Deni Indra Permana



TABLE OF CONTENTS

STEP
01

PROBLEM STATEMENT

Bima Purnomo Sandi



STEP
02

EDA DAN INSIGHT

Ramado Dipradelana



STEP
03

DATA PRE-PROCESSING

M. Irfan Fadhlurrahman



STEP
04

MODELING

M. Irfan Fadhlurrahman



STEP
05

RECOMMENDATION

Deni Indra Permana





01

Problem Statement

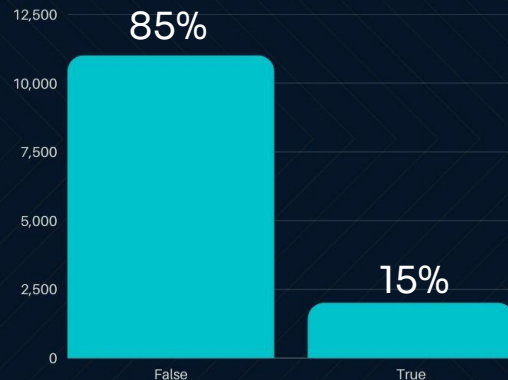


Problem Statement

Out of all visitors only 15% converted and make purchase. Based on the most updated [reference](#), average ecommerce conversion/purchase rate is between 2-5%

Then why
this is a
problem?

TRANSACTION WITH REVENUE



Based on Data...



On 2021, consumers who shop online increase up to 88%*. **15% of Purchase Rate** is too small.

*CNN Indonesia, Konsumen Belanja Online RI Melonjak 88 Persen pada 2021

CVR*



Actual :15%
Reference : 2-5%

*CVR(Conversion Rate)
or Purchase Rate: the
percentage of website
visitors who buy
something on the site.

BOUNCE RATE*



Actual :Up to 20%
Reference : 26%-70%

*Bounce Rate: the rate at
which new visitors visit a
site and immediately
click away without doing
anything

EXIT RATE*



Actual :Up to 20%
Reference : 26%-70%

*Exit Rate: The
percentage of pageviews
on the website that end
at that specific page

Compared to the low bounce rate and exit rate, our
conversion rate 15% would be relatively low. With such a low
bounce rate & exit rate, we should have achieved more
conversion.



Our Objective

- Giving insights and action recommendation to increase conversion rate
- Build a machine learning model to automatically predict which visitor will purchase
- Analyze Visitors behavior for better understanding

Now
15%
Conversion Rate

➡➡

Goal!
20%
Conversion Rate

Metric

$$\text{Conversion Rate} = \frac{\text{Number of Transaction}}{\text{Number of Web Visit}} \times 100\%$$



EDA & Insight



EDA & Insight

There are 12.330 rows and 18 column of online shopper customers sessions data in a one-year period.

- Administrative
- Administrative Duration
- Informational
- Informational Duration
- Product Related
- Product Related Duration
- Bounce Rate
- Exit Rate
- Page Values
- Special Day
- Month
- Operating Systems
- Browser
- Region
- Traffic Type
- Visitor Type
- Weekend
- **Revenue**

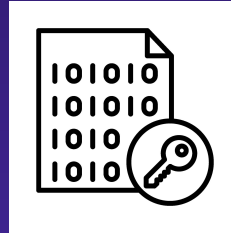


EDA & Insight



Low-Variance numeric feature:

- Visitor Type
- Special Day



No actual categories :

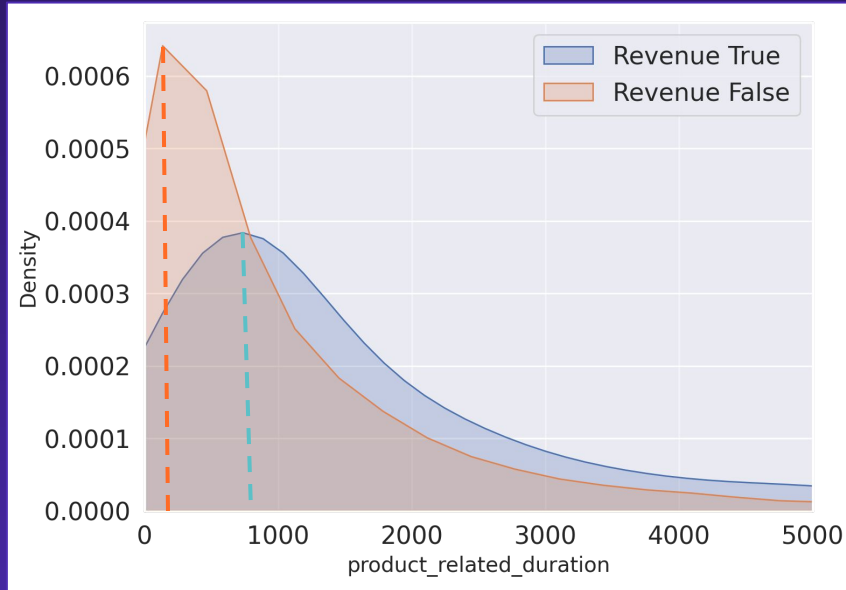
- Operating System
- Region
- Browser
- Traffic Type



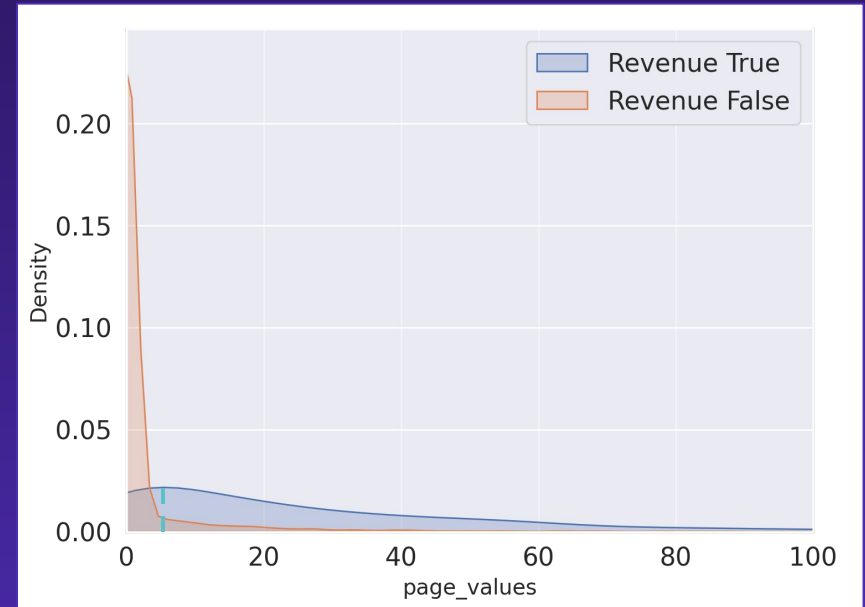
**10 numerical features are
positive Skew**

EDA & Insight

Product related duration



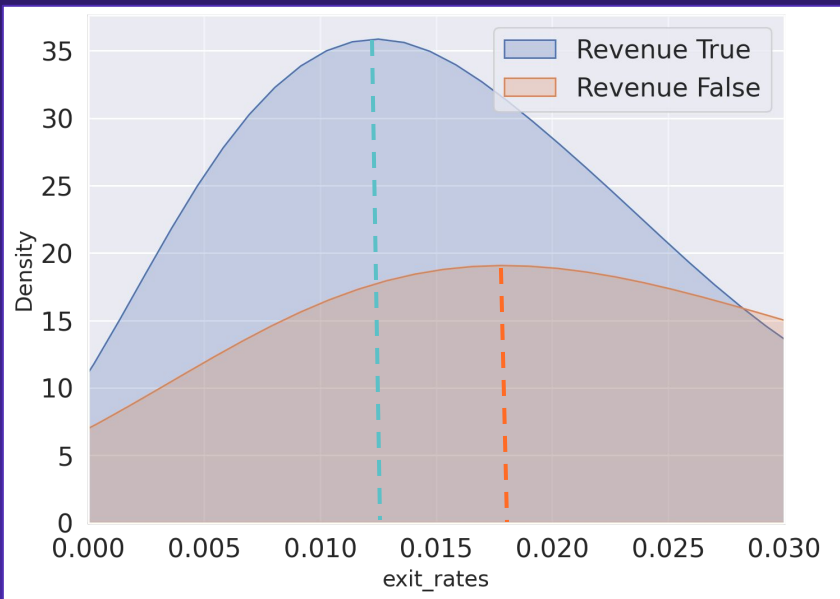
Page values



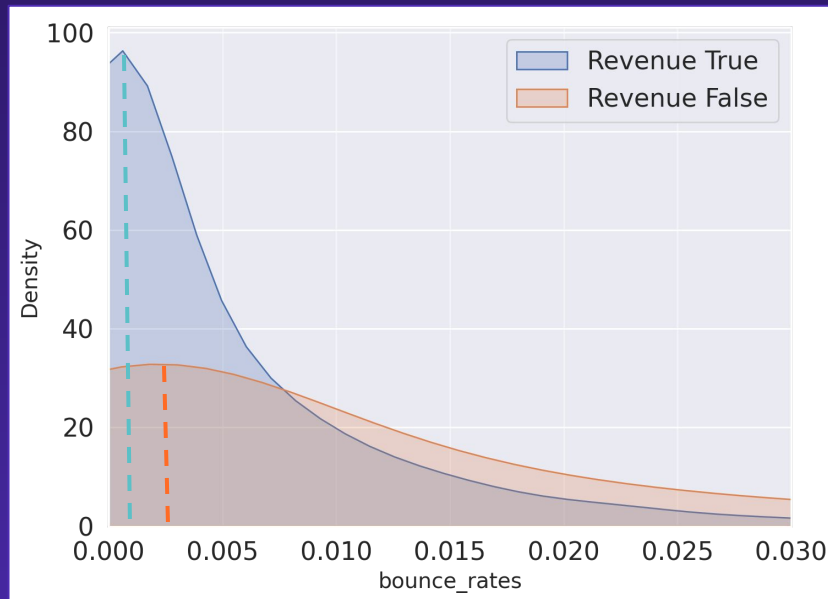
Visitor with higher page values and higher product related duration tend to make a purchase.

EDA & Insight

Exit rates

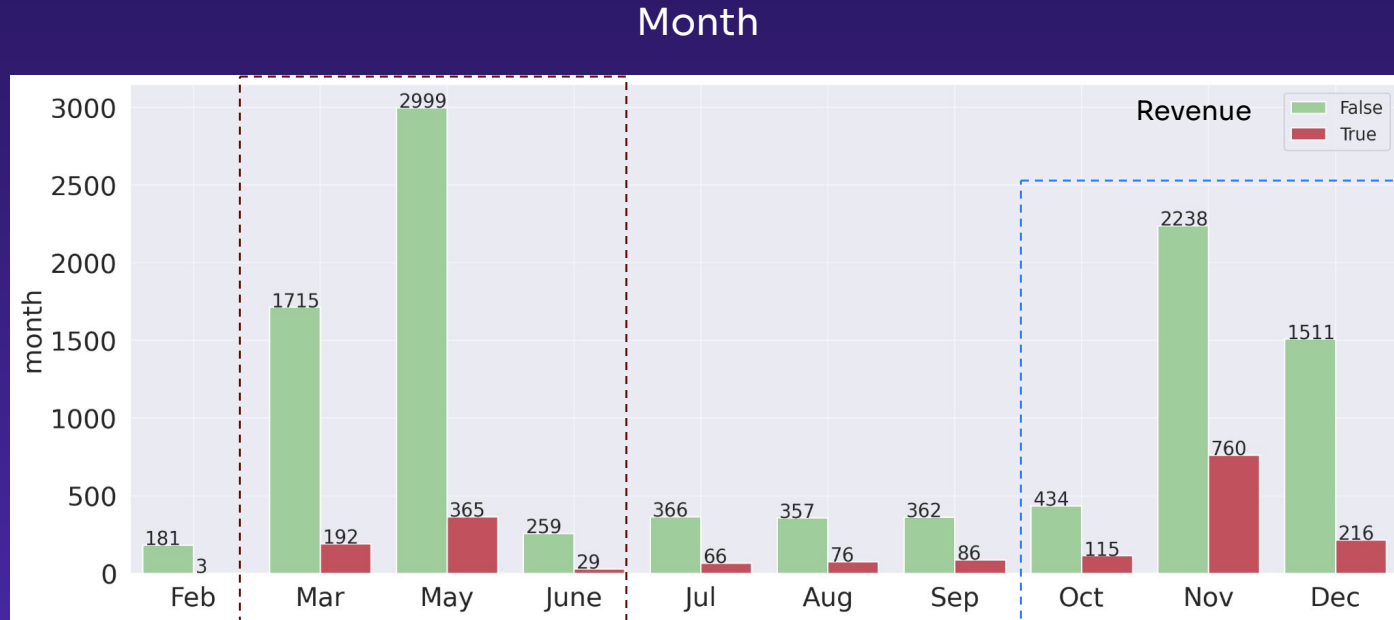


Bounce rates



Visitor with lower exit rates and lower bounce rates tend to make a purchase.

EDA & Insight



The highest amount of web traffic is at March - June but the conversion rate is lower compared to October - December. There is a potential to increase the conversion rate at March - June.



03

Data Pre-Processing



Data Preprocessing

Missing Value

There is no missing value in the dataset so nothing to handle

Handling Outliers

There are numerical features that indicates outliers, handled using Log Transformer

Categorical Feature Encoding

Month: Quarter Binning*
Weekend & Visitor type::One Hot

Duplicated Data

There are 125 rows of duplicated data, it must be dropped

Numeric Feature Scaling

RobustScaler

Handling Class Imbalance

SMOTE

*quarter binning encoded in order to represents whole months

Feature Engineering

Feature Selection

Drop Features* :

- Operating System
- Browser
- Region
- Traffic Type

Feature Extraction

There are 3 additional features

- Average administrative duration per page
- Average informational duration per page
- Average product related duration per page

Split Dataset

80% Data Train, 20% Data Test



*Already encoded and no additional informations that represents actual categories



04

Modelling & Evaluation



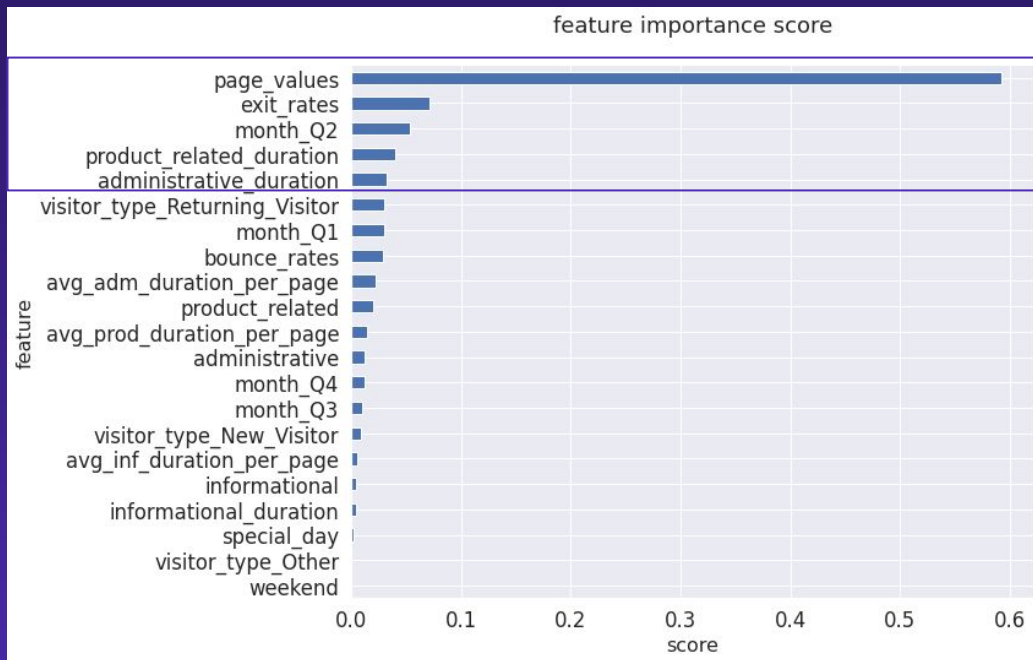
Modelling & Evaluation

Model	ROC-AUC		Precision		Recall	
	Train	Test	Train	Test	Train	Test
Random Forest	0.94	0.92	0.63	0.62	0.80	0.78
Extra Trees	0.82	0.81	0.35	0.35	0.57	0.54
XGBoost	0.98	0.92	0.75	0.60	0.90	0.73

The primary metric is **Precision**. We try to minimize the false positive, which is predicted make a purchase but actually not purchase, is costly to our revenue. We do not want the visitor who actually does not make a purchase, not detected by our model.

The best model we choose is **Random Forest** because the difference between train and test metrics are lowest compared to other model. After experimenting with hyperparameter tuning and cross validation, the model performance did not improve and they overfitted so we use the model before hyperparameter tuning with parameter **max_depth = 7** and **class_weight = balanced_subsample**.

Feature Importance



Visitors with high number of visited pages are likely to make a purchase.

Top Five Feature Importance:

- Page Value
- Exit Rates
- Month: May and June
- Product Related Duration
- Administrative Duration

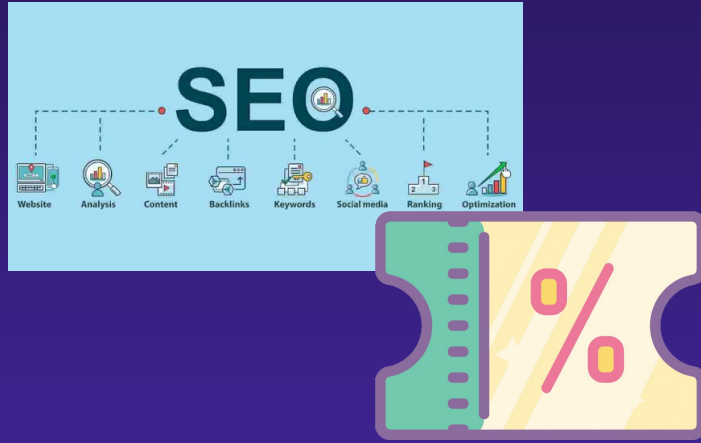




Business Recommendation



Page Value



Page value can be increased by increasing traffic count quality. Page value can be optimized with Search Engine Optimization (SEO) or giving voucher per category product.

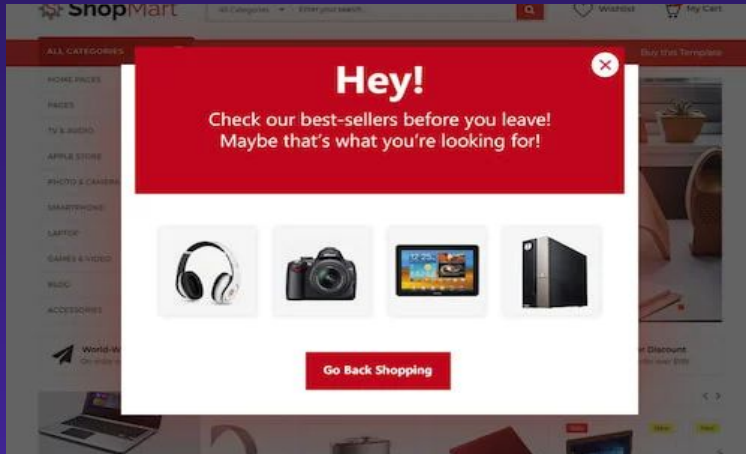
Month



Give some vouchers and holding an event in May.

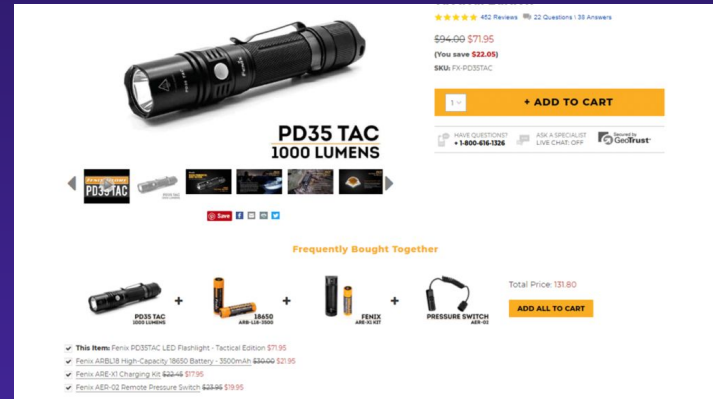
Reference :
<https://www.wordtracker.com/academy/seo/page-optimization/how-to-optimize-web-page>

Exit Rate



Optimize pages with a high exit rate. Add Exit Intent Popups, CTA in the right place, and provide a live chat feature.

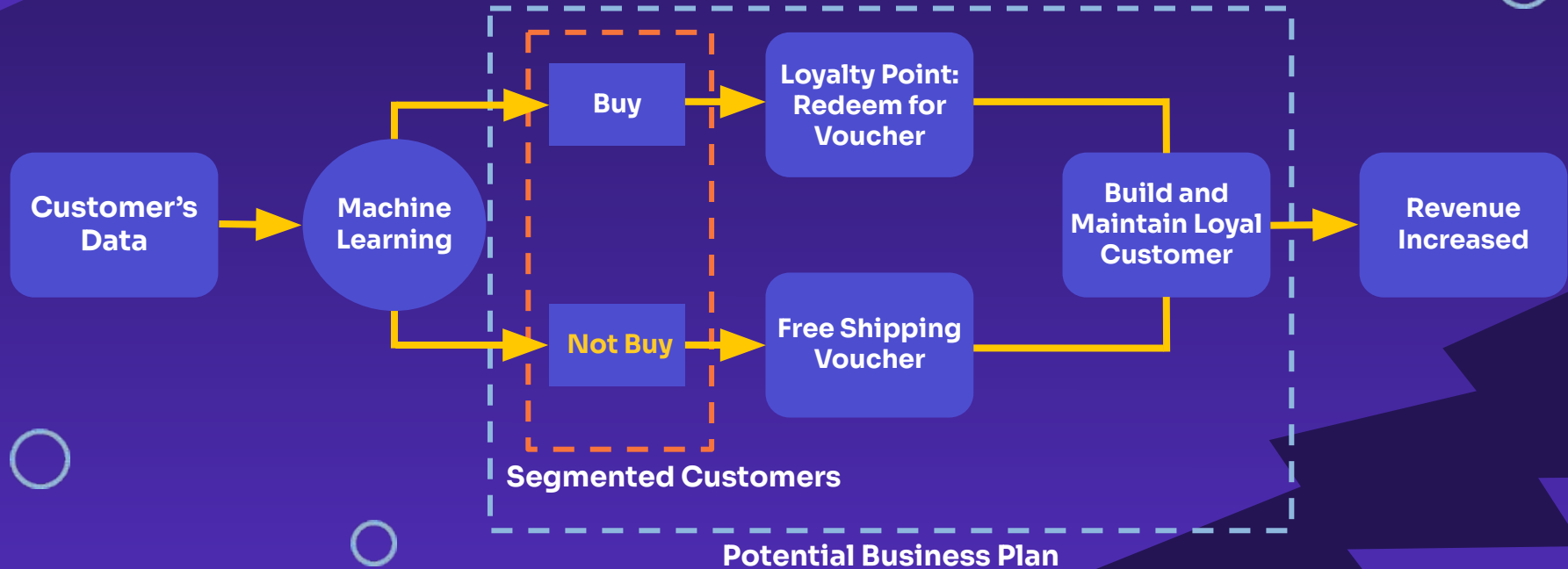
Product Related Duration



Improvements in terms of interface, ease of access, clarity of information, and product demonstration on Related Products.

Reference : <https://databox.com/lower-exit-ratee>

Business Recommendation Flow



Our Strategies

Our Focused Program

Voucher: Free Shipping



Mechanism:

- Customer get free shipping voucher.

9 out of 10 consumers say free shipping No. 1 incentive to shop online more



9 out of 10

61% of consumers are at "somewhat likely" to cancel their purchase if free shipping is not offered



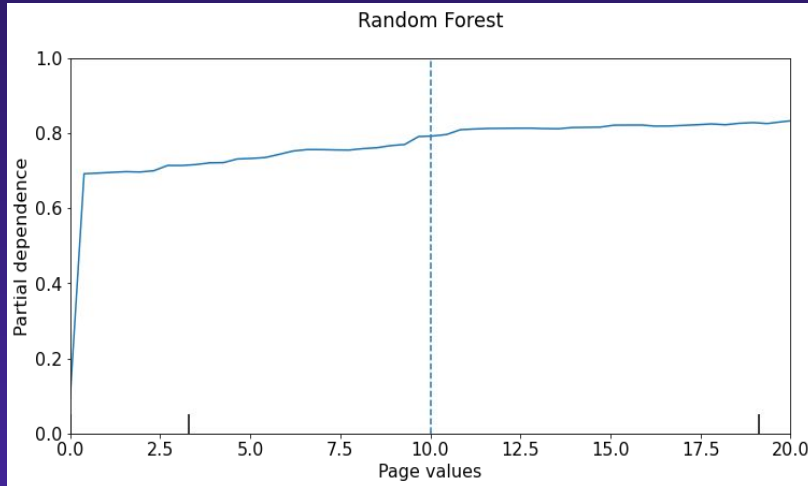
61%

93% of online buyers are encouraged to buy more products if free shipping options are available



93%

Business Simulation



Visitors who are predicted not to make a transaction will get a voucher. When these visitors get a voucher, they tend to explore more in our e-commerce so that the page values will likely to increase. As per the left plot, we can assume the average page values of these visitors increase to 10 pages. From this increase in page values, the conversion rate increases 75%.



Potential Revenue



15% Conversion
Rate



75% Conversion
Rate

Rp.30.000*

382

x

Rp.11.460.000



Rp.30.000*

1849

x

Rp.55.470.000

Potential
Revenue

Increased
4.8X

*Assumption: Average revenue per online shopper per transaction in Indonesia

xx xx xx xx
xx xx xx xx
xx xx xx xx
xx xx xx xx
xx xx xx xx

○

Danke!

Do you have any questions?

○



Appendix

Feature Selection

Evaluation Metric	Before Selection (%)	After Selection (%)
ROC-AUC	92.2	91.1
Precision	61.8	60.2
Recall	78.0	77.2

The model with **5 most important features** have **slightly lower** precision and ROC-AUC score on test set then the model with all features.



Notebook for simulation: [link](#)

```
# select the threshold based on PDP
page_values_threshold = 10

# change the page_values data for page_values <= 5
filter_for_revenue_false = X_test_new['revenue'] == 0
func = lambda x: page_values_threshold if x <= page_values_threshold else x
X_test_new.loc[filter_for_revenue_false, 'page_values'] = X_test_new.loc[filter_for_revenue_false, 'page_values'].apply(func)

# prediction on test data
y_proba = model.predict_proba(X_test_new.drop('revenue', axis=1))[:, 1]
y_pred = (y_proba >= 0.5).astype(int)

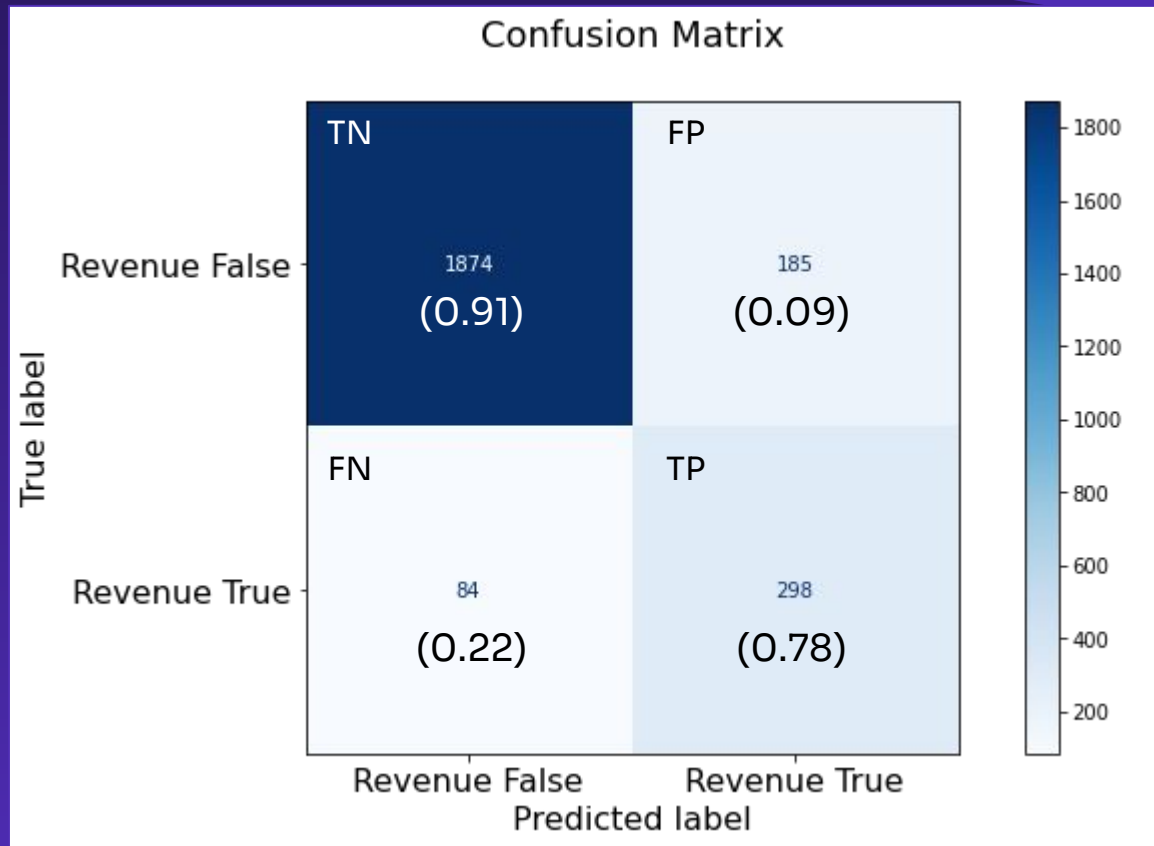
# calculate convert visitors
total_treated_visitors = X_test[filter_for_revenue_false].shape[0]
potential_convert_visitors = ((X_test_new['revenue'] == 0) & (y_pred == 1)).sum()
purchase_rate_after_treatment = potential_convert_visitors * 100 / X_test_new.shape[0]

# display the results
print(f"Page values threshold: {page_values_threshold}")
print(f"Total visitors who get a treatment: {total_treated_visitors} visitors")
print(f"Convert-to-purchase visitors after get a treatment: {potential_convert_visitors} visitors")
print(f"Purchase Rate After Treatment: {purchase_rate_after_treatment:.2f}%")
```

```
Page values threshold: 10
Total visitors who get a treatment: 2059 visitors
Convert-to-purchase visitors after get a treatment: 1849 visitors
Purchase Rate After Treatment: 75.75%
```

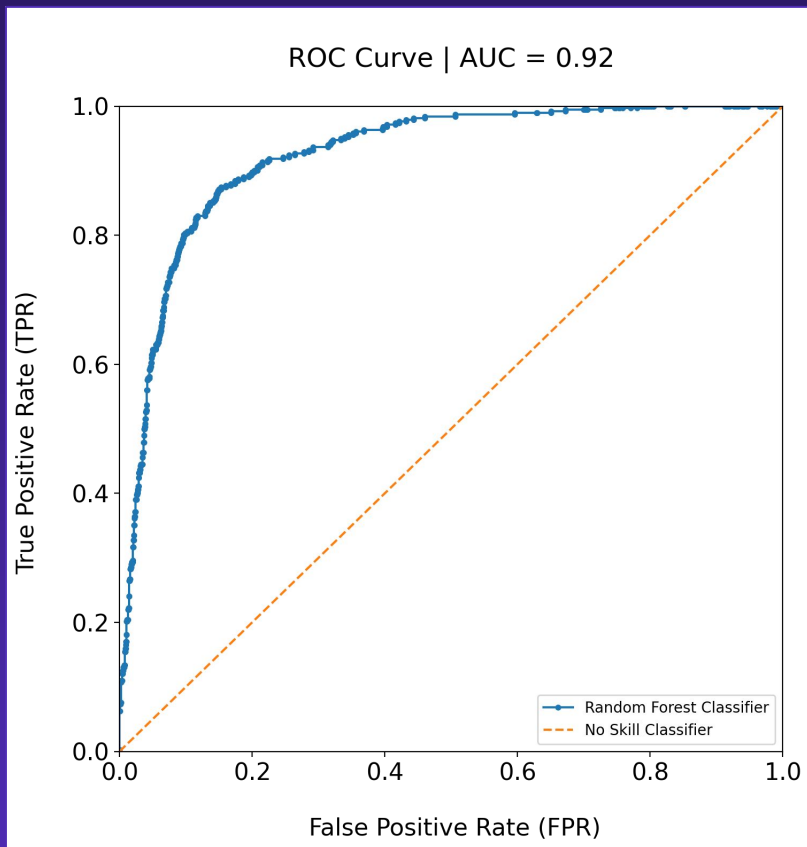

Confusion Matrix (Random Forest)

For Test Data



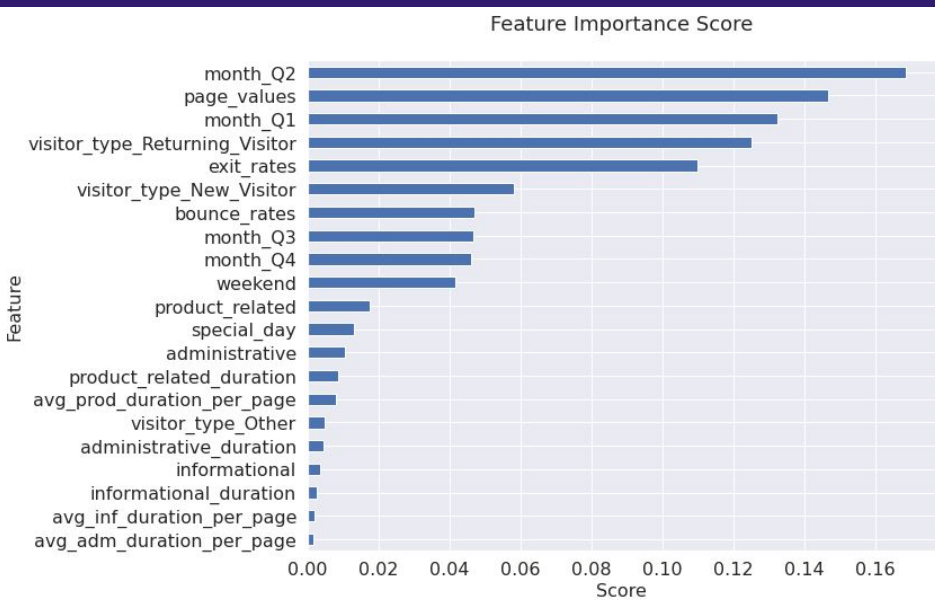
ROC (Random Forest)

For Test Data



	threshold	tpr	fpr
0	0.146	0.110	0.002
1	0.158	0.154	0.009
2	0.329	0.649	0.061
3	0.377	0.702	0.068
4	0.611	0.819	0.114
5	0.704	0.859	0.146
6	0.826	0.963	0.369
7	0.839	0.971	0.403
8	0.958	1.000	0.801
9	0.990	1.000	0.938

Extra Trees



XGBoost

