Rakamin
Academy

Predict Clicked Ads Customer Classification by using Machine Learning

Created by:
Destya Febiolla

destya.febiolla@gmail.com
https://www.linkedin.com/in/destya-febiolla
https://github.com/Destyf

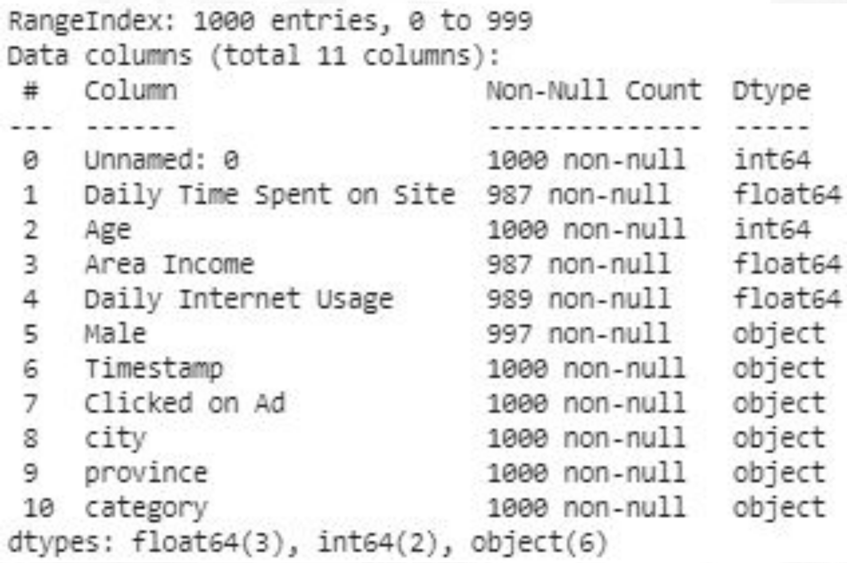
Hello! I'm Destya, a career shifting enthusiast with a strong interest in Data Science. Equipped with a solid foundation from a Data Science Bootcamp, I specialize in data preprocessing and machine learning. With it I am determined to make a meaningful impact as a Data Scientist, and eager to collaborate with fellow data enthusiasts and organizations seeking to harness the power of data for informed decision-making.

Supported by:
Rakamin Academy
Career Acceleration School
www.rakamin.com

Overview

“A company in Indonesia wants to assess the effectiveness of an advertisement they have aired. This is crucial for the company to understand the extent of the advertisement's reach, thereby attracting customers to view it. By analyzing historical advertisement data and uncovering insights and patterns, this can assist the company in determining their marketing targets. The focus of this case is to create a machine learning classification model that can accurately identify the right target customers. ”

Explanatory Data Analysis



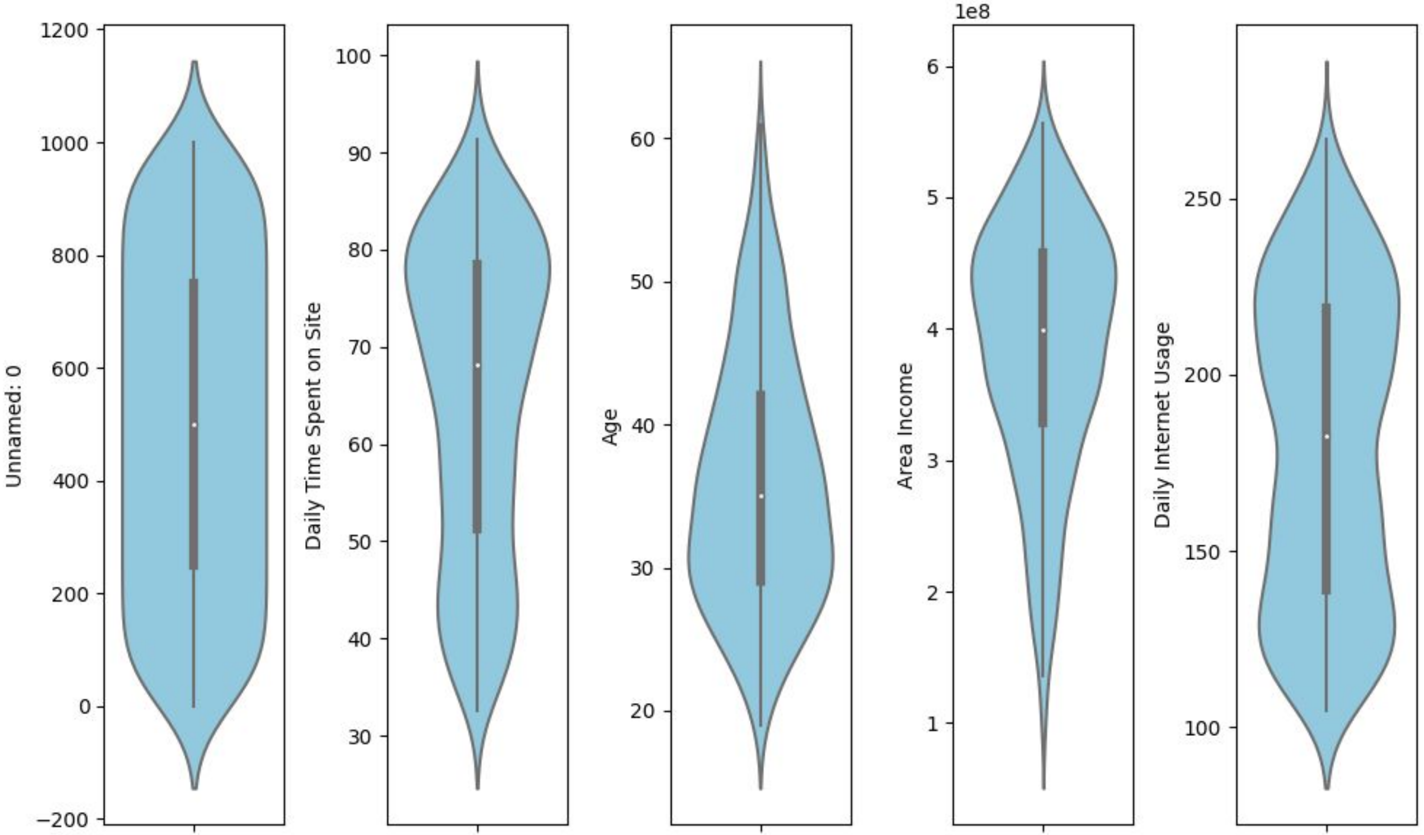
```
RangeIndex: 1000 entries, 0 to 999
Data columns (total 11 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Unnamed: 0                1000 non-null   int64  
 1   Daily Time Spent on Site  987 non-null    float64
 2   Age                       1000 non-null   int64  
 3   Area Income               987 non-null    float64
 4   Daily Internet Usage      989 non-null    float64
 5   Male                      997 non-null    object 
 6   Timestamp                 1000 non-null   object 
 7   Clicked on Ad             1000 non-null   object 
 8   city                      1000 non-null   object 
 9   province                  1000 non-null   object 
 10  category                  1000 non-null   object 
dtypes: float64(3), int64(2), object(6)
```

- Rows : 1.000
- Features : 11
- Duplicated :0
- Missing Value :

Daily Time Spent on Site	13
Area Income	13
Daily Internet Usage	11
Male	3

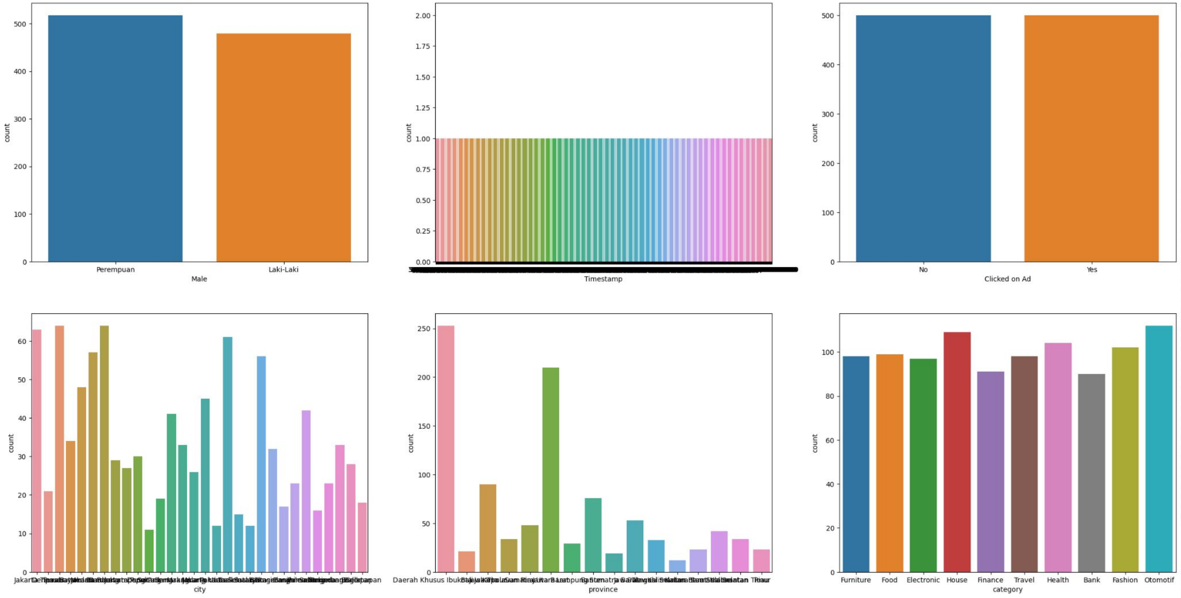
For the detail of my codes, you can see here here

Univariate Analysis



For the detail of my codes, you can see here here

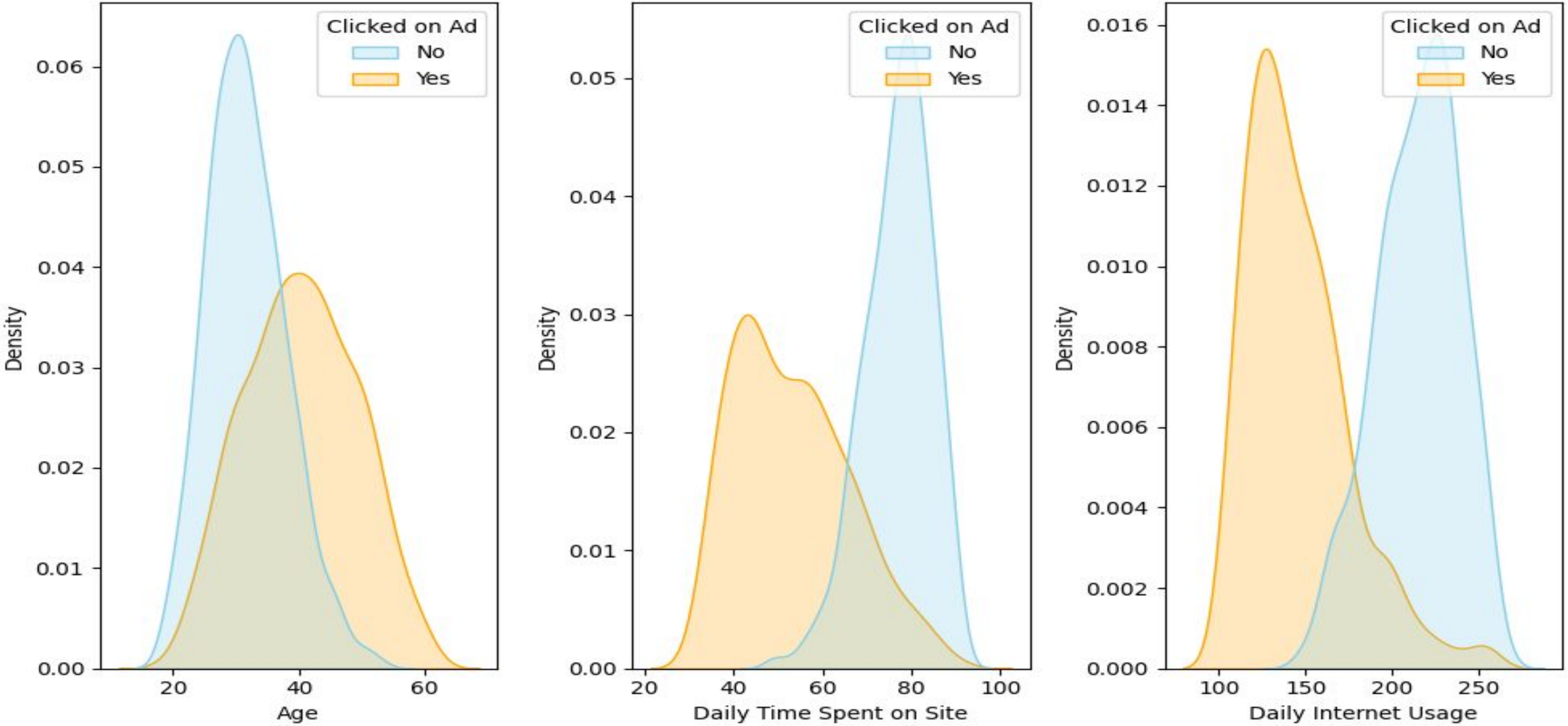
Univariate Analysis



- There are outliers in the feature Area Income.
- The data distribution of the features Daily Internet Usage and Daily Time Spent on Site is bimodal, whereas Age has a positively skewed distribution, and Area Income has a negatively skewed distribution.
- The categorical feature that will be used as the target for this project is Clicked on Ad, where this feature has values Yes & No with equal frequencies of 500.

For the detail of my codes, you can see here here

Bivariate Analysis



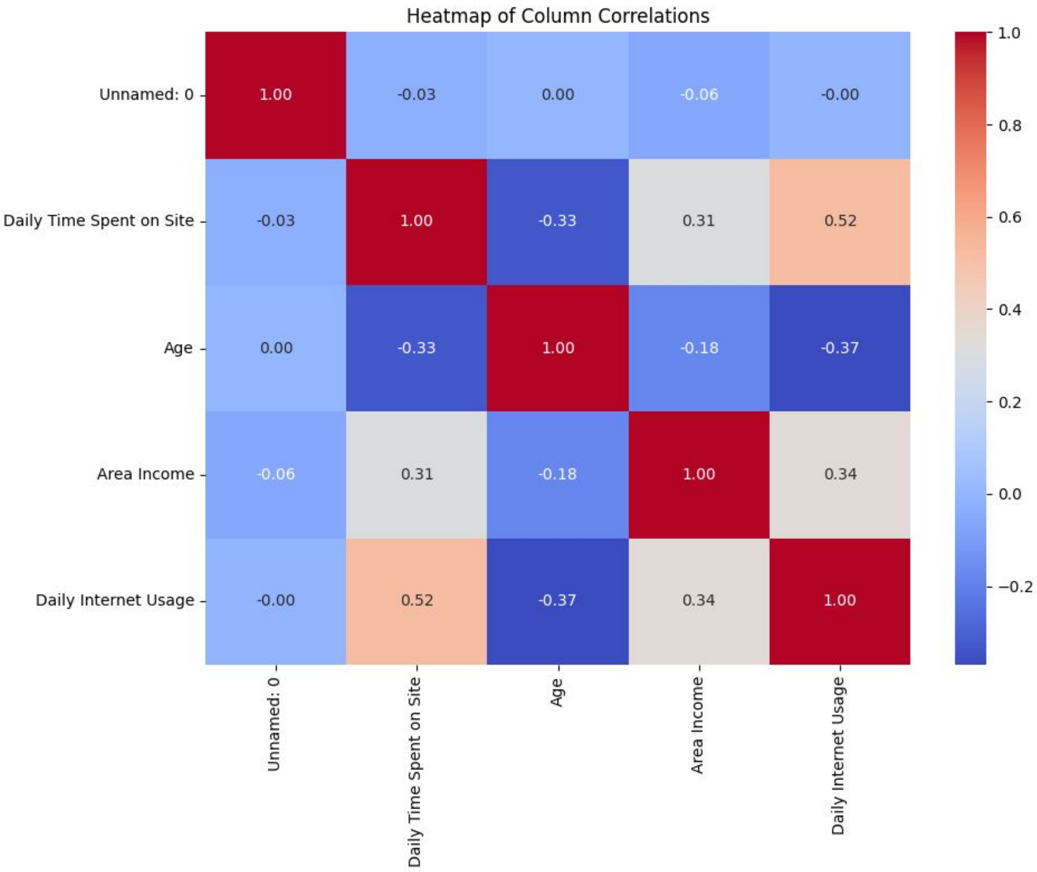
For the detail of my codes, you can see here here

Bivariate Analysis

- **Feature Age vs Clicked on Ad**: The number of customers who did not click on ads is highest at the age of 30, higher than customers who clicked on ads at the age of 40.
- **Feature Daily Time Spent on Site vs Clicked on Ad**: The longer time a customer spends on the site, the higher the likelihood that the customer did not click on ads.
- **Feature Daily Internet Usage vs Clicked on Ad**: The lower the internet usage by the customer, the higher the likelihood that the customer clicked on ads; conversely, the higher the internet usage, the higher the likelihood that the customer did not click on ads. Therefore, the number of 'Clicked Ads' (both Yes & No) in Daily Internet Usage has equal frequency.

For the detail of my codes, you can see here here

Correlation Heatmap



- The features with high correlation are Daily Time Spent on Site, Daily Internet Usage, and Area Income.
- The feature with low correlation is Age.

For the detail of my codes, you can see here here

Data Cleaning & Preprocessing

Rakamin Academy

01 **Handling Missing Value**

- Daily Time Spent on Site (13): Imputed with the median value.
- Area Income (13): Imputed with the mean value.
- Daily Internet Usage (11): Imputed with the median value.
- Male (3): Imputed with the mode value.

02 **Correcting Dtypes**

Change Data Type of Feature Timestamp to Datetime

03 **Feature Extraction**

Extract data from the 'Timestamp' column into year, month, week, and day

04 **Remove /Drop Column**

Drop columns 'Unnamed: 0', 'Area Income', 'Male', 'city', 'province', 'category', 'Timestamp'.

For the detail of my codes, you can see here here

Data Cleaning & Preprocessing

Rakamin Academy

05 **Feature Encoding**

Encode column ‘Clicked on Ad’

06 **Normalization Data**

Normalization data using MinMaxScaler

07 **Split Data**

- X (Feature): Daily Time Spent on Site, Age, Daily Internet Usage, Year, Month, Week, Day
- Y (Target) : Clicked on Ad:

For the detail of my codes, you can see here here

Data Modeling

Data Train : Data Test = 80 : 20

LogisticRegression

- Accuracy : 0.97
- Precision : 0.99
- Recall : 0.95
- F1-Score : 0.97
- AUC : 0.99

DecisionTree Classifier

- Accuracy : 0.94
- Precision : 0.97
- Recall : 0.91
- F1-Score : 0.94
- AUC : 0.94

RandomForest Classifier

- Accuracy : 0.96
- Precision : 0.98
- Recall : 0.95
- F1-Score : 0.96
- AUC : 0.99

KNeighbors Classifier

- Accuracy : 0.97
- Precision : 0.99
- Recall : 0.95
- F1-Score : 0.97
- AUC : 0.98

GradientBoosting Classifier

- Accuracy : 0.96
- Precision : 0.97
- Recall : 0.96
- F1-Score : 0.96
- AUC : 0.98

For the detail of my codes, you can see here here

Hyperparameter Tuning

LogisticRegression

- Accuracy	: 0.96
- Precision	: 1.00
- Recall	: 0.93
- F1-Score	: 0.96
- AUC	: 0.99

Parameters:

- penalty = ['l1', 'l2']
- C = [float(x) for x in np.linspace (0.001, 0.1, 1, 100)]

KNeighbors Classifier

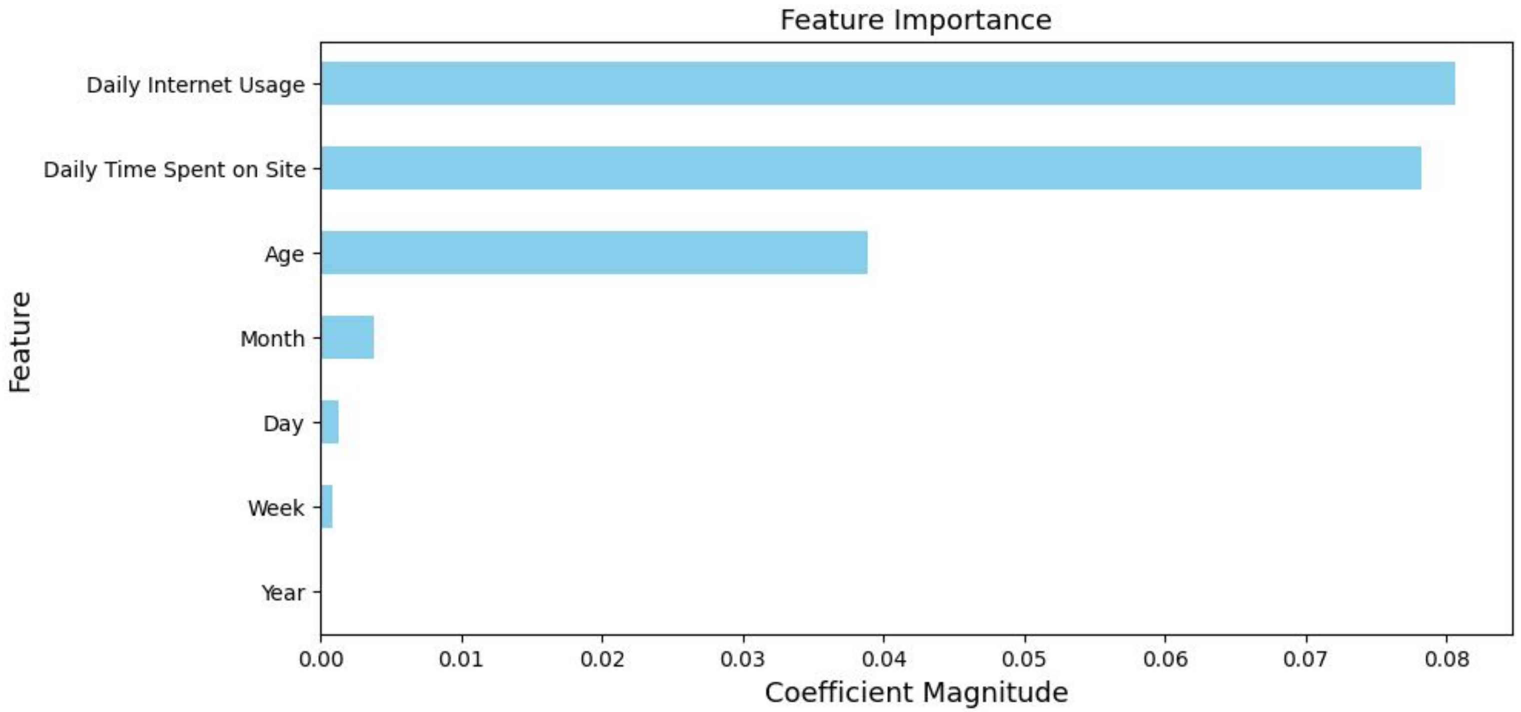
- Accuracy	: 0.95
- Precision	: 1.00
- Recall	: 0.90
- F1-Score	: 0.95
- AUC	: 0.98

Parameters:

- n_neighbors = list(range(1,30))
- p=[1,2]
- algorithm = ['auto', 'ball_tree', 'kd_tree', 'brute']

For the detail of my codes, you can see here here

Features Importances

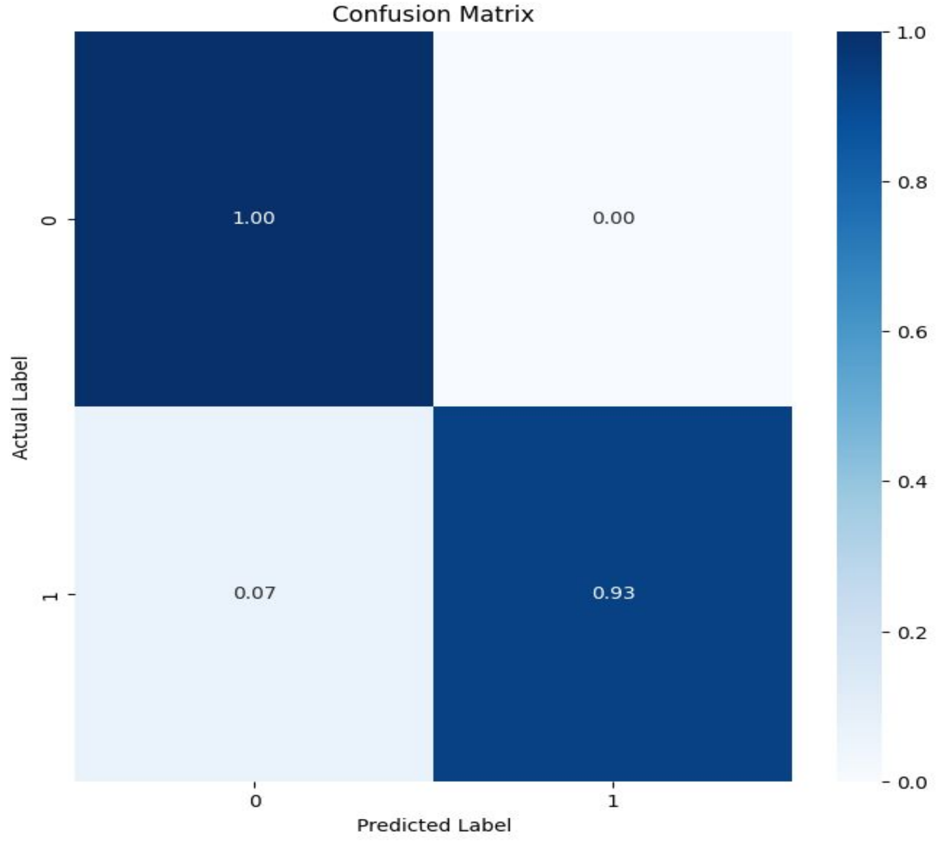


Coefficients for top features:

- Daily Internet Usage 0.080685
- Daily Time Spent on Site -0.078246

For the detail of my codes, you can see here here

Confusion Metrics



For the detail of my codes, you can see here here

Business Recommendation

From the results of the Logistic Regression model, the two most important features are 'Daily Internet Usage' and 'Daily Time Spent on Site'. The business recommendations are:

- **Optimize Website Engagement**: Encourage customers to spend more time on website by enhancing user experience, providing valuable content, and ensuring easy navigation.
- **Target Low Internet Usage Segments**: Identify customer segments with lower internet usage, as they are more likely to click on ads. Consider tailoring specific ad campaigns for these segments.
- **Segmented Ad Strategies**: Since Daily Internet Usage and Daily Time Spent on Site have nearly equal influence, consider segmenting ad strategies based on these factors. For customers with high internet usage and low time spent on the site, focus on attention-grabbing ads. For customers with low internet usage and high time spent on the site, invest in engaging content and interactive ad formats.

For the detail of my codes, you can see here here

Business Simulation

Without Machine Learning Model

Assuming:

Num of Users Viewing the Ad : 1000

Marketing Cost : 500.000

ConversionRate (Clicked Ad & Buy Product) : 2%

Price Per Product : 70.000

Clicked Ad : 50% (Based on Dataset)

Simulations:

Total Revenue = Num of Users × Conversion Rate × Price Per Product x Clicked Ad

= 1000 × 2% x 70.000 x 50% = 700.000

Profit = Total Revenue − Marketing Cost = 700.000 − 500.000 = **Rp200.000**

For the detail of my codes, you can see here here

Business Simulation

With Machine Learning Model

Assuming:

Num of Users Viewing the Ad : 1000

Marketing Cost : 500.000

ConversionRate (Clicked Ad & Buy Product) : 2%

Price Per Product : 70.000

Clicked Ad : 50% (Based on Dataset)

Accuracy 97%

Simulations:

Total Revenue = Num of Users × Conversion Rate × Price Per Product x 96%

= 1000 × 2% x 70.000 x 96% = 1.358.000

Profit = Total Revenue − Marketing Cost = 1.358.000 − 500.000 = **Rp858.000**

Profit Increased Up to

429%

For the detail of my codes, you can see here here