PROMO CLICK PREDICTION USING LOGISTIC REGRESSION

MACHINE LEARNING PROJECT
E-COMMERCE CASE STUDY

by :
Muhammad Faqih Abdussalam

PROJECT OVERVIEW

Problem Statement:

- Divisi e-commerce ingin mengetahui apakah user akan mengklik banner promo di halaman website.

Objective:

- Membangun model prediktif berbasis machine learning untuk mengklasifikasi klik iklan berdasarkan perilaku dan data user.

Solution Approach::

- Menggunakan Logistic Regression dan tahapan standard Machine Learning pipeline.

DATASET SUMMARY

Jumlah Data: 1000 records

Jumlah Fitur: 10 fitur (6 numerik, 4 kategorikal)

Target Label: Clicked on Ad (0 = tidak klik, 1 = klik)

Fitur Utama yang Digunakan:

- Daily Time Spent on Site
- Age
- Area Income
- Daily Internet Usage
- Male

Distribusi label:

- Klik: 500 (50%)
- Tidak Klik: 500 (50%)

EXPLORATORY DATA ANALYSIS

Data eksplorasi dengan head()

```
print("\n[1] Data eksplorasi dengan head(), info(), describe(), shape")
print("Lima data teratas:")
print(data.head())
```

```
[1] Data eksplorasi dengan head(), info(), describe(), shape
Lima data teratas:
   Daily Time Spent on Site  Age  Area Income  Daily Internet Usage  \
0                     68.95   35     61833.90                256.09
1                     80.23   31     68441.85                193.77
2                     69.47   26     59785.94                236.50
3                     74.15   29     54806.18                245.89
4                     68.37   35     73889.99                225.58

                          Ad Topic Line           City  Male     Country  \
0     Cloned 5thgeneration orchestration     Wrightburgh     0     Tunisia
1     Monitored national standardization       West Jodi     1       Nauru
2       Organic bottom-line service-desk        Davidton     0  San Marino
3  Triple-buffered reciprocal time-frame  West Terrifurt     1       Italy
4          Robust logistical utilization    South Manuel     0     Iceland

         Timestamp  Clicked on Ad
0   3/27/2016 0:53              0
1    4/4/2016 1:39              0
2  3/13/2016 20:35              0
3   1/10/2016 2:31              0
4    6/3/2016 3:36              0
```

Data eksplorasi dengan info()

```
print("Informasi dataset:")
print(data.info())
```

```
Informasi dataset:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 10 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Daily Time Spent on Site  1000 non-null   float64
 1   Age                       1000 non-null   int64
 2   Area Income               1000 non-null   float64
 3   Daily Internet Usage      1000 non-null   float64
 4   Ad Topic Line             1000 non-null   object
 5   City                      1000 non-null   object
 6   Male                      1000 non-null   int64
 7   Country                   1000 non-null   object
 8   Timestamp                 1000 non-null   object
 9   Clicked on Ad             1000 non-null   int64
dtypes: float64(3), int64(3), object(4)
memory usage: 78.3+ KB
None
```

EXPLORATORY DATA ANALYSIS

Data eksplorasi dengan describe

```
print("Statistik deskriptif dataset:")
print(data.describe)
```

```
Statistik deskriptif dataset:
<bound method NDFrame.describe of      Daily Time Spent on Site  Age  Area Income  Daily Internet Usage  \
0                       68.95   35     61833.90                256.09
1                       80.23   31     68441.85                193.77
2                       69.47   26     59785.94                236.50
3                       74.15   29     54806.18                245.89
4                       68.37   35     73889.99                225.58
..                        ...  ...          ...                   ...
995                     72.97   30     71384.57                208.58
996                     51.30   45     67782.17                134.42
997                     51.63   51     42415.72                120.37
998                     55.55   19     41920.79                187.95
999                     45.01   26     29875.80                178.35

                          Ad Topic Line           City  Male  \
0      Cloned 5thgeneration orchestration     Wrightburgh     0
1       Monitored national standardization       West Jodi     1
2         Organic bottom-line service-desk        Davidton     0
3    Triple-buffered reciprocal time-frame  West Terrifurt     1
4            Robust logistical utilization    South Manuel     0
..                                     ...             ...   ...
995          Fundamental modular algorithm       Duffystad     1
996        Grass-roots cohesive monitoring     New Darlene     1
997          Expanded intangible solution   South Jessica     1
998   Proactive bandwidth-monitored policy     West Steven     0
999        Virtual 5thgeneration emulation     Ronniemouth     0

                    Country        Timestamp  Clicked on Ad
0                   Tunisia   3/27/2016 0:53              0
1                     Nauru    4/4/2016 1:39              0
2                San Marino  3/13/2016 20:35              0
3                     Italy   1/10/2016 2:31              0
4                   Iceland    6/3/2016 3:36              0
..                      ...             ...            ...
995                 Lebanon  2/11/2016 21:49              1
996  Bosnia and Herzegovina   4/22/2016 2:07              1
997                Mongolia   2/1/2016 17:24              1
998                Guatemala  3/24/2016 2:35              0
999                  Brazil   6/3/2016 21:43              1

[1000 rows x 10 columns]>
```

Data eksplorasi dengan shape

```
print("Ukuran dataset:")
print(data.shape)
```

```
Ukuran dataset:
(1000, 10)
```

Cek missing value

```
print("\n[3] Cek missing value")
print(data.isnull().sum().sum())
```

```
[3] Cek missing value
0
```

Data eksplorasi dengan mengecek distribusi label menggunakan fungsi groupby() dan size()

```
print("\n[2] Data eksplorasi dengan mengecek distribusi label menggunakan fungsi groupby() dan size()")
print(data.groupby('Clicked on Ad').size())
```

```
[2] Data eksplorasi dengan mengecek distribusi label menggunakan fungsi groupby() dan size()
Clicked on Ad
0    500
1    500
dtype: int64
```

EXPLORATORY DATA ANALYSIS

Data eksplorasi dengan dengan mengecek korelasi dari setiap feature menggunakan fungsi corr()

```
print("\n[4] Data eksplorasi dengan dengan mengecek korelasi dari setiap feature menggunakan fungsi corr()")
print(data.select_dtypes(include='number').corr())
```

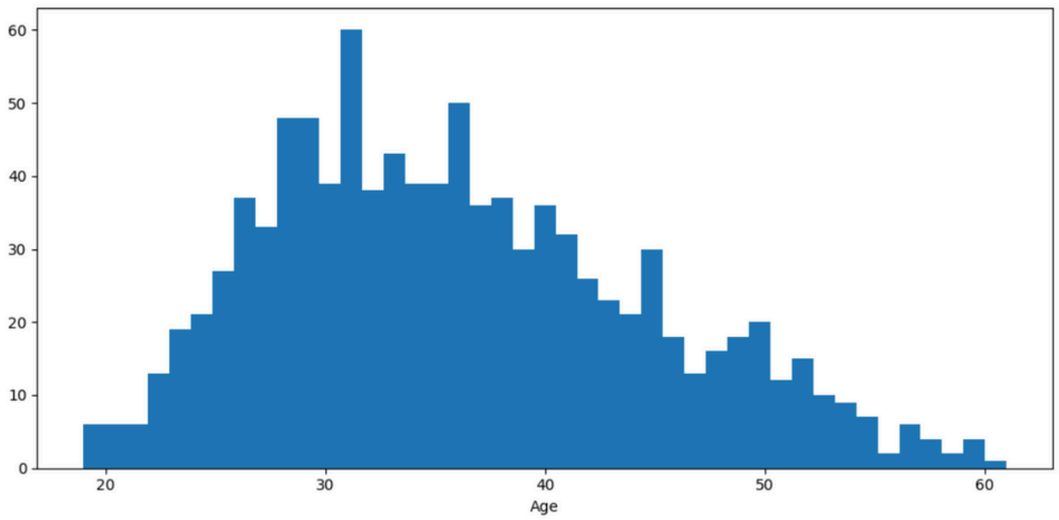
```
[4] Data eksplorasi dengan dengan mengecek korelasi dari setiap feature menggunakan fungsi corr()
                          Daily Time Spent on Site       Age  Area Income  \
Daily Time Spent on Site                  1.000000 -0.331513     0.310954
Age                                      -0.331513  1.000000    -0.182605
Area Income                               0.310954 -0.182605     1.000000
Daily Internet Usage                      0.518658 -0.367209     0.337496
Male                                     -0.018951 -0.021044     0.001322
Clicked on Ad                            -0.748117  0.492531    -0.476255

                          Daily Internet Usage      Male  Clicked on Ad
Daily Time Spent on Site              0.518658 -0.018951      -0.748117
Age                                  -0.367209 -0.021044       0.492531
Area Income                           0.337496  0.001322      -0.476255
Daily Internet Usage                  1.000000  0.028012      -0.786539
Male                                  0.028012  1.000000      -0.038027
Clicked on Ad                        -0.786539 -0.038027       1.000000
```

EXPLORATORY DATA ANALYSIS

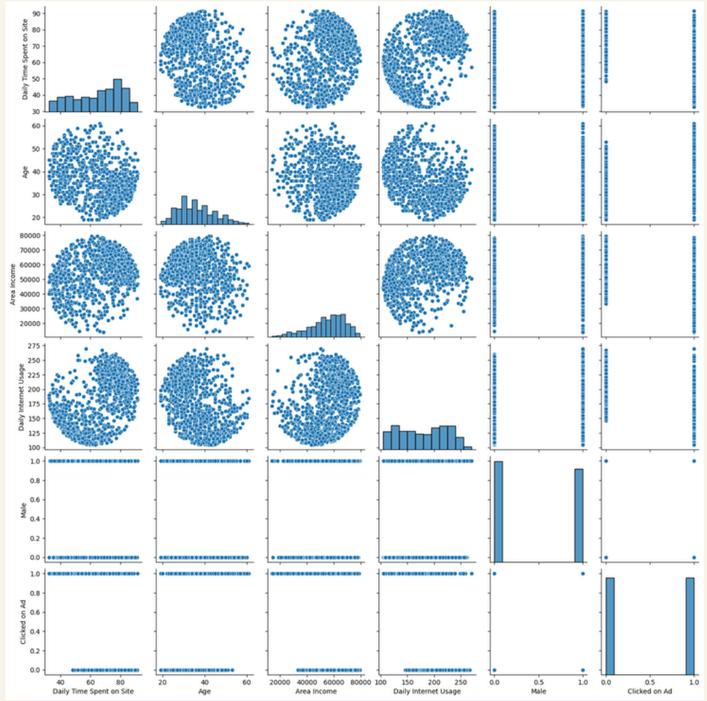
Visualisasi Jumlah user dibagi ke dalam rentang usia (Age) menggunakan histogram (hist()) plot

```
plt.figure(figsize=(10, 5))
plt.hist(data['Age'], bins=data.Age.nunique())
plt.xlabel('Age')
plt.tight_layout()
plt.show()
```



Visualisasi Jumlah user dibagi ke dalam rentang usia (Age) menggunakan histogram (hist()) plot

```
plt.figure()
sns.pairplot(data)
plt.show()
```



DATA PREPARATION

```
print("\n[6] Lakukan pemodelan dengan Logistic Regression, gunakan perbandingan 80:20 untuk training vs testing")
X = data.drop(['Ad Topic Line','City','Country','Timestamp','Clicked on Ad'], axis = 1)
y = data['Clicked on Ad']
```

```
# splitting the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 42)
```

Fitur kategorik (City, Country, dll) **dihapus**

Dataset dipisahkan:

- X: hanya numerik (fitur)
- y: Clicked on Ad (label)

Split data:

- 80% Training
- 20% Testing
- random_state = 42

MODEL BUILDING

```
# Call the classifier
logreg = LogisticRegression()
# Fit the classifier to the training data
logreg = logreg.fit(X_train, y_train)
# Prediksi model
y_pred = logreg.predict(X_test)
```

Model: Logistic Regression

- Dilatih menggunakan X_train & y_train
- Digunakan scikit-learn

EVALUASI MODEL

```
print("Evaluasi Model Performance:")
print("Training Accuracy :", logreg.score(X_train, y_train))
print("Testing Accuracy :", logreg.score(X_test, y_test))
```

```
Evaluasi Model Performance:
Training Accuracy : 0.9675
Testing Accuracy : 0.935
```

```
print("\n[7] Print Confusion matrix dan classification report")

#apply confusion_matrix function to y_test and y_pred
print("Confusion matrix:")
cm = confusion_matrix(y_test, y_pred)
print(cm)

#apply classification_report function to y_test and y_pred
print("Classification report:")
cr = classification_report(y_test, y_pred)
print(cr)
```

```
[7] Print Confusion matrix dan classification report
Confusion matrix:
[[ 84   5]
 [  8 103]]
Classification report:
              precision    recall  f1-score   support

           0       0.91      0.94      0.93        89
           1       0.95      0.93      0.94       111

    accuracy                           0.94       200
   macro avg       0.93      0.94      0.93       200
weighted avg       0.94      0.94      0.94       200
```

CONCLUSIONS

Model sudah sangat baik dalam memprediksi user yang akan mengklik website atau tidak, dapat dilihat dari nilai accuracy = 0.94. Dataset memiliki jumlah label yang seimbang (balance class), sehingga evaluasi performansi dapat menggunakan metrik Accuracy.

THANK YOU