Predict Customer Personality to boost marketing campaign by using Machine Learning

Created by:
Haolia
haoliaaul@gmail.com
https://www.linkedin.com/in/haolia/

"Geophysical engineering graduates who diligently want to fulfil a role where intellectual, integrity, and curiosity are highly valued. Motivated, able to research, design, implement new features and learn various software. Skill handling problems with unique ways to develop innovative solutions. Proficient using Python, SQL, Tableau and other statistical tools for data multi purposes. Looking for opportunities in data analyst, data science, data engineer and Business Intelligence. "

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

Data Cleaning & Preprocessing

- Cek missing Value : Setelah dilkukan pengecekan pada missing value. Feature `*income*` memiliki 24 baris data yang *null* dengan percentase 0.010714% dari jumlah row data pada pada feature ini. Sehingga dapat dikatakan data null ini sangat sedikit sehingga akan drop saja.

```
# Percentage of null values
df_row.isna().sum()/len(df_row)
```

[6] ✓ 0.1s

```
Output exceeds the size limit. Open the full output data in a text editor
Unnamed: 0            0.000000
ID                    0.000000
Year_Birth            0.000000
Education             0.000000
Marital_Status        0.000000
Income                0.010714
Kidhome               0.000000
Teenhome              0.000000
Dt_Customer           0.000000
Recency               0.000000
MntCoke               0.000000
MntFruits             0.000000
MntMeatProducts       0.000000
MntFishProducts       0.000000
MntSweetProducts      0.000000
MntGoldProds          0.000000
NumDealsPurchases     0.000000
NumWebPurchases       0.000000
NumCatalogPurchases   0.000000
```

- Tidak terdapat data duplikat

```
df.duplicated().sum()
```

```
0
```

```
df.shape
```

```
(2216, 39)
```

Untuk selengkapnya, dapat melihat jupyter notebook disini

Data Cleaning & Preprocessing

Proses ***feature encoding*** dan ***feature standardisation***.

1. Label Encoder dilakukan pada data categorical dengan memberikan label baru, dalam hal ini feature Pendidikan, dilabeli angka 0 -1 agar dapat diolah.
2. One Hot Encoder untuk membuat kolom baru dari variable categorical dari label yang ada sebagai vector biner yang bernilai inteher, 0 dan 1.

```
# label encoder
mapping_education = {
                    'SMA' : 0,
                    'D3' : 1,
                    'S1' : 2,
                    'S2' : 3,
                    'S3' : 4
                    }

df['Education'] = df['Education'].map(mapping_education)
```

[29] ✓ 0.1s

```
df['Education'].sample(5)
```

[35] ✓ 0.9s

```
325     2
1386    4
597     3
2117    4
97      2
Name: Education, dtype: int64
```

```
df_cats = df[['Education']].copy()
# One hot encoder
for cat in ['Marital_Status', 'age_range', 'is_parents']:
    onehots = pd.get_dummies(df[cat], prefix=cat)
    df_cats = df_cats.join(onehots)
```

✓ 0.5s

```
df_cats.sample(5)
```

✓ 0.8s

	Education	Marital_Status_Bertunangan	Marital_Status_Cerai	Marital_Status_Duda	Marital_Status_Janda	Marital_Status_Lajang	Marital_Status_Menikah	age_range_midd
673	2	0	0	0	0	0	1	
103	4	0	0	0	0	0	1	
2152	2	0	0	0	0	0	1	
563	1	0	0	0	0	0	1	
1069	2	0	0	0	0	0	1	

Untuk selengkapnya, dapat melihat jupyter notebook disini

Standardization

3. Standardization, dilakukan untuk merubah sebaran data agar mendekati distribusi normal

```
# Standardization
from sklearn.preprocessing import StandardScaler
df_scaled = df.copy()
ss = StandardScaler()

for col in numerical_features:
    df_scaled[col] = ss.fit_transform(df_scaled[[col]])

display(df_scaled.shape, df_scaled.head(3))
```

[36] ✓ 0.7s

```
(2240, 40)
```

	Unnamed: 0	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Recency	...	age	join_at_age	total_kids	is_parents
0	0	5524	1957	2	Lajang	0.234063	-0.825218	-0.929894	2012-04-09	0.307039	...	0.985345	0.896633	-1.264505	0
1	1	2174	1954	2	Lajang	-0.234559	1.032559	0.906934	2014-08-03	-0.383664	...	1.235733	1.312600	1.396361	1
2	2	4141	1965	2	Bertunangan	0.769478	-0.825218	-0.929894	2013-08-21	-0.798086	...	0.317643	0.314278	-1.264505	0

3 rows × 40 columns