We Care About
Your Future

Prediksi Churn Customer E-Commerce

Final Project

Oleh : Yayang Nurul Aulia, Vety Bhakti Lestari, Daud Fernando, Aufa Azhari Hafidh

Business Understanding

Perusahaan XYZ tengah melakukan analisis terkait perkembangan pelanggan yang ada di salah satu saluran perdagangan elektroniknya. Terdapat sebuah kondisi fluktuatif terhadap alur keuangan yang ada dari para pelanggannya. Pimpinan menginginkan adanya kejelasan terhadap tren yang terjadi terhadap kondisi tersebut berupa mesin prediksi kriteria pelanggan yang ada.

Business Question

1. Variabel apa yang menentukan pelanggan tidak melakukan pembelian repetitif di perusahaan XYZ?
2. Model algoritma apa yang terbaik dalam mengkategorisasi pelanggan milik perusahaan XYZ?

Objective

1. Melakukan eksplorasi data untuk menentukan keterhubungan antar variabel apakah *churn* atau tidak.
2. Membuat perbandingan beberapa algoritma untuk membuat sebuah model terbaik dalam mengklasifikasi pelanggan di perusahaan XYZ.

We Care About
Your Future

Data Dictionary

Data	Variable	Discerption
E Comm	CustomerID	Unique customer ID
E Comm	Churn	Churn Flag
E Comm	Tenure	Tenure of customer in organization
E Comm	PreferredLoginDevice	Preferred login device of customer
E Comm	CityTier	City tier
E Comm	WarehouseToHome	Distance in between warehouse to home of customer
E Comm	PreferredPaymentMode	Preferred payment method of customer
E Comm	Gender	Gender of customer
E Comm	HourSpendOnApp	Number of hours spend on mobile application or website
E Comm	NumberOfDeviceRegistered	Total number of deceives is registered on particular customer
E Comm	PreferedOrderCat	Preferred order category of customer in last month
E Comm	SatisfactionScore	Satisfactory score of customer on service
E Comm	MaritalStatus	Marital status of customer
E Comm	NumberOfAddress	Total number of added added on particular customer
E Comm	Complain	Any complaint has been raised in last month
E Comm	OrderAmountHikeFromlastYear	Percentage increases in order from last year
E Comm	CouponUsed	Total number of coupon has been used in last month
E Comm	OrderCount	Total number of orders has been places in last month
E Comm	DaySinceLastOrder	Day Since last order by customer
E Comm	CashbackAmount	Average cashback in last month

We Care About
Your Future

DATA PREPROCESSING

1. Check and Handling Missing Values

```
# cek
df_churn.isna().sum()
```

```
CustomerID                     0
Churn                          0
Tenure                       264
PreferredLoginDevice           0
CityTier                       0
WarehouseToHome              251
PreferredPaymentMode           0
Gender                         0
HourSpendOnApp               255
NumberOfDeviceRegistered       0
PreferedOrderCat               0
SatisfactionScore              0
MaritalStatus                  0
NumberOfAddress                0
Complain                       0
OrderAmountHikeFromlastYear  265
CouponUsed                   256
OrderCount                   258
DaySinceLastOrder            307
CashbackAmount                 0
dtype: int64
```

```
# handling with median
df_churn['Tenure'].fillna(df_churn['Tenure'].median(), inplace=True)
df_churn['WarehouseToHome'].fillna(df_churn['WarehouseToHome'].median(), inplace=True)
df_churn['HourSpendOnApp'].fillna(df_churn['HourSpendOnApp'].median(), inplace=True)
df_churn['OrderAmountHikeFromlastYear'].fillna(df_churn['OrderAmountHikeFromlastYear'].median(), inplace=True)
df_churn['CouponUsed'].fillna(df_churn['CouponUsed'].median(), inplace=True)
df_churn['OrderCount'].fillna(df_churn['OrderCount'].median(), inplace=True)
df_churn['DaySinceLastOrder'].fillna(df_churn['DaySinceLastOrder'].median(), inplace=True)
df_churn.isna().sum()
```

fill in missing values with median

```
CustomerID                     0
Churn                          0
Tenure                         0
PreferredLoginDevice           0
CityTier                       0
WarehouseToHome                0
PreferredPaymentMode           0
Gender                         0
HourSpendOnApp                 0
NumberOfDeviceRegistered       0
PreferedOrderCat               0
SatisfactionScore              0
MaritalStatus                  0
NumberOfAddress                0
Complain                       0
OrderAmountHikeFromlastYear    0
CouponUsed                     0
OrderCount                     0
DaySinceLastOrder              0
CashbackAmount                 0
dtype: int64
```

DATA PREPROCESSING

2. Check and Handling Inconsistent Data

```
cat = ['PreferredLoginDevice','PreferredPaymentMode','Gender','PreferedOrderCat','MaritalStatus','Complain']

for ftr in cat:
    print(df_churn[ftr].value_counts(),'\n')
```

```
Mobile Phone    2765
Computer        1634
Phone           1231
Name: PreferredLoginDevice, dtype: int64
```

```
Debit Card          2314
Credit Card         1501
E wallet             614
UPI                  414
COD                  365
CC                   273
Cash on Delivery     149
Name: PreferredPaymentMode, dtype: int64
```

```
Laptop & Accessory    2050
Mobile Phone          1271
Fashion                826
Mobile                 809
Grocery                410
Others                 264
Name: PreferedOrderCat, dtype: int64
```

↓

```
Mobile Phone    3996
Computer        1634
Name: PreferredLoginDevice, dtype: int64
```

```
Debit Card    2314
CC            1774
E wallet       614
COD            514
UPI            414
Name: PreferredPaymentMode, dtype: int64
```

```
Mobile Phone          2080
Laptop & Accessory    2050
Fashion                826
Grocery                410
Others                 264
Name: PreferedOrderCat, dtype: int64
```

DATA PREPROCESSING

3. Check and Handling Duplicated Values

```
df_churn.duplicated().sum()
```

```
0
```

Tidak ditemukan nilai yang duplikat.

Diketahui bahwa jumlah baris sebelum memfilter outlier sebanyak 5630. Kemudian setelah dihandling menggunakan Z-Score, jumlah barisnya menjadi 5350 baris.

4. Check and Handling Outlier

```
# using zscore
from scipy import stats

print(f'Jumlah baris sebelum memfilter outlier: {len(df_churn)}')

filtered_entries = np.array([True] * len(df_churn))
for col in num:
    zscore = abs(stats.zscore(df_churn[col]))
    filtered_entries = (zscore < 3) & filtered_entries

churn_filtered = df_churn[filtered_entries]

print(f'Jumlah baris setelah memfilter outlier: {len(churn_filtered)}')
```

```
Jumlah baris sebelum memfilter outlier: 5630
Jumlah baris setelah memfilter outlier: 5350
```

DATA PREPROCESSING

5. Feature Engineering

```
# feature selection
churn_filtered.drop(columns = ['CustomerID','NumberOfAddress'], inplace = True)
churn_filtered
```

Menghapus kolom CustomerID dan NumberOfAddress karena tidak diperlukan dalam analisis.

Memberi code pada data yang berlabel kategori seperti tertera pada syntax di samping.

```
# feature encode
for cat in [['PreferredLoginDevice','PreferredPaymentMode','Gender','PreferedOrderCat','MaritalStatus']]:
    onehots = pd.get_dummies(churn_filtered[cat], prefix=cat)
    churn_filtered2 = churn_filtered.join(onehots)
churn_filtered2.head()
```

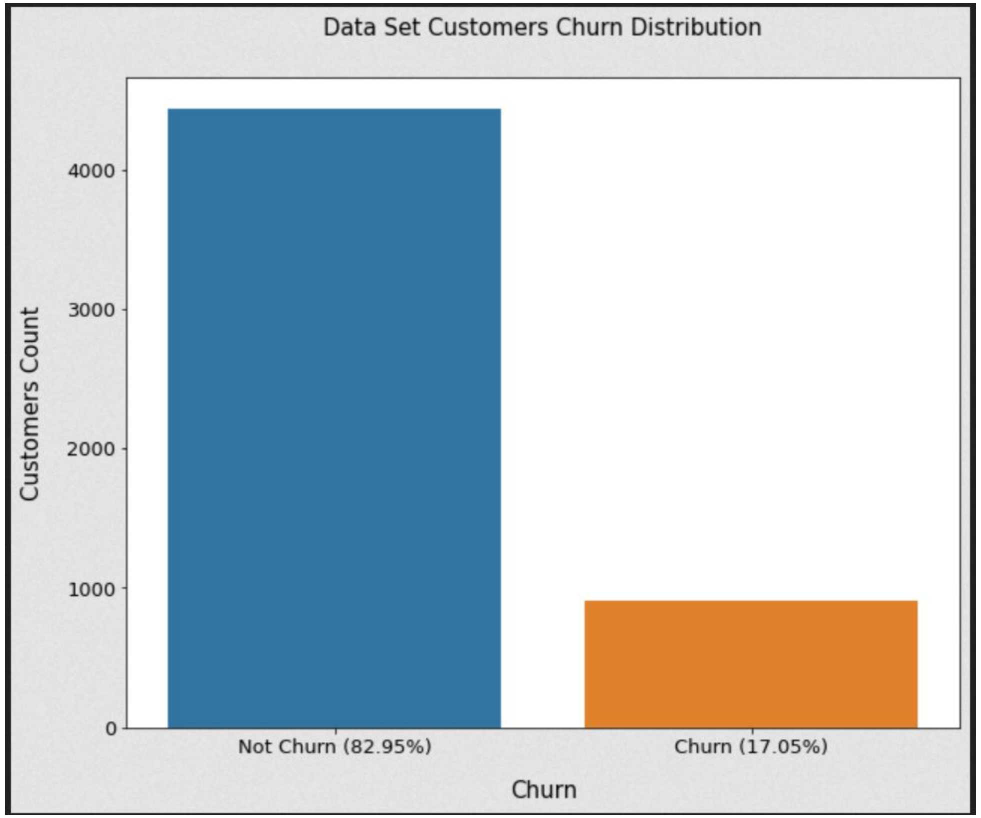
```
# feature scaling
from sklearn.preprocessing import StandardScaler
x_churn = churn_filtered2.drop(columns = ['Churn','PreferredLoginDevice','PreferredPaymentMode',
                                          'Gender','PreferedOrderCat','MaritalStatus'], axis=1)

y_churn = churn_filtered2['Churn']

sc = StandardScaler()
x_churn2 = sc.fit_transform(x_churn)
```

Menstandarisasi data yang sebelumnya di-*encode*.

We Care About
Your Future

DATA PREPROCESSING

6. Check and Handling Imbalanced Class

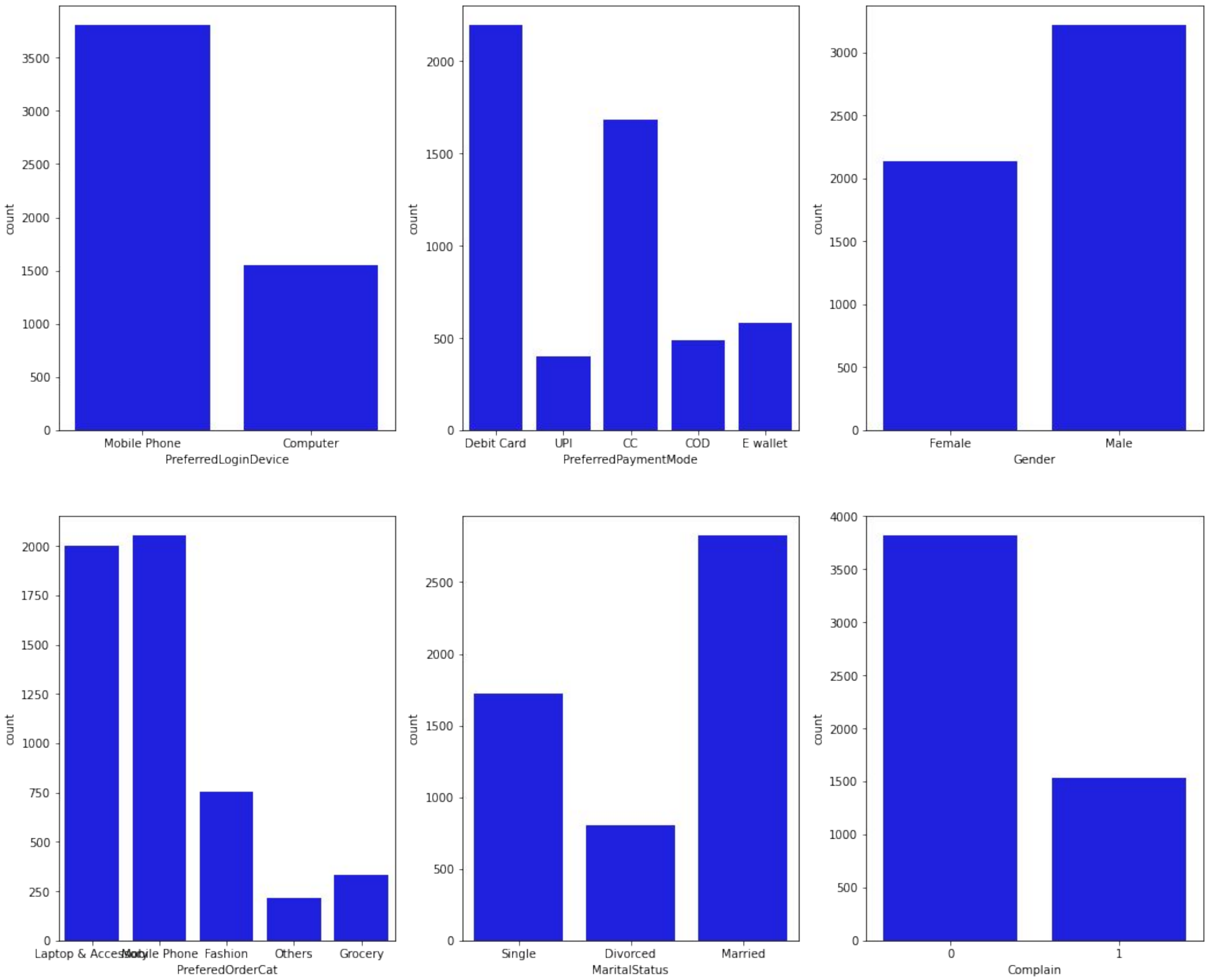


- Ada 912 dari 5350 customer yang memilih untuk churn
- Ada 4438 dari 5350 customer yang tidak churn

```
# Handling with SMOTE
from imblearn.over_sampling import SMOTE
oversample = SMOTE()
x_smote, y_smote = oversample.fit_resample(x_churn3, y_churn)
```

We Care About
Your Future

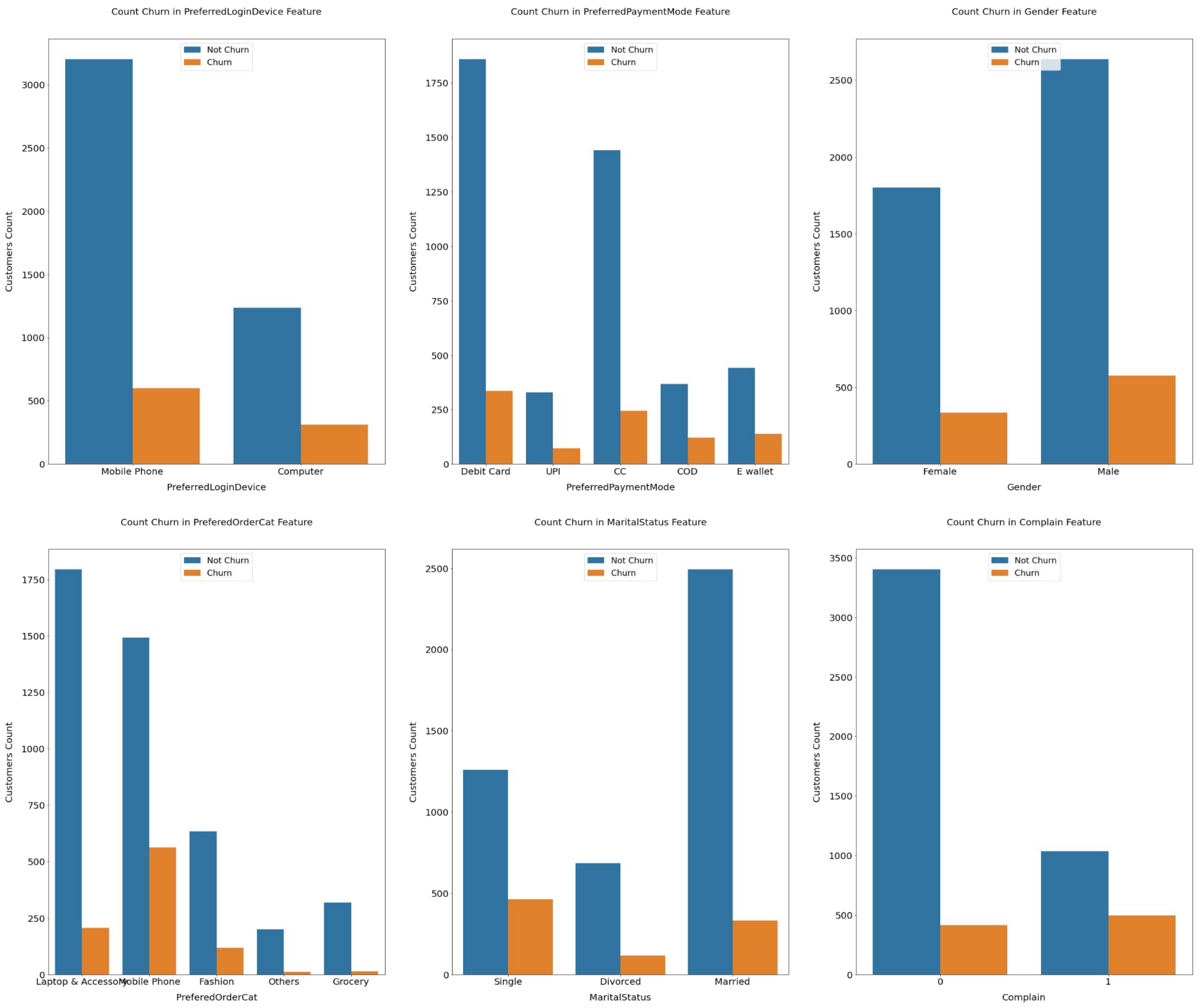
EXPLORATORY DATA ANALYSIS



- Sebagian besar customer login aplikasi via mobile phone (71%).
- Sebagian besar customer melakukan pembayaran menggunakan Debit Card (41%) dan CC (32%).
- Sebagian besar customer berjenis kelamin pria (60%).
- Sebagian besar customer membeli produk kategori Laptop & Electronic (37%) dan Mobile Phone (38%).
- Sebagian besar customer sudah menikah (52%).
- Sebagian besar customer tidak mengajukan complain (71%).

We Care About
Your Future

EXPLORATORY DATA ANALYSIS



Sebagian besar customer yang memilih untuk churn memiliki karakteristik :

- Login melalui mobile phone.
- Melakukan pembayaran menggunakan Debit Card.
- Berjenis kelamin pria.
- Membeli kategori produk Mobile Phone.
- Berstatus belum menikah atau single.
- Pernah mengajukan complain.

MODELLING AND EVALUATION

1. Logistic Regression

```
from sklearn.linear_model import LogisticRegression

model_lr = LogisticRegression(random_state=42)
model_lr.fit(x_train, y_train)

lr_pred = model_lr.predict(x_test)

eval_classification(model_lr,lr_pred, x_train, y_train, x_test, y_test)
```

```
Accuracy (Test Set): 0.8164
Precision (Test Set): 0.8202
Recall (Test Set): 0.8184
F1-Score (Test Set): 0.8193
AUC: 0.82
```

```
# Cek overfitting
# print the scores on training and test set

print('Training set score: {:.4f}'.format(model_lr.score(x_train, y_train)))
print('Test set score: {:.4f}'.format(model_lr.score(x_test, y_test)))
```

```
Training set score: 0.8165
Test set score: 0.8164
```

MODELLING AND EVALUATION

2. Random Forest

```
# Bagging (random forest)

from sklearn.ensemble import RandomForestClassifier
rf = RandomForestClassifier(random_state=42)
rf.fit(x_train, y_train)

y_pred = rf.predict(x_test)
eval_classification(rf, y_pred, x_train, y_train, x_test, y_test)
```

```
Accuracy (Test Set): 0.9865
Precision (Test Set): 0.9878
Recall (Test Set): 0.9856
F1-Score (Test Set): 0.9867
AUC: 0.99
```

```
# Cek overfitting
# print the scores on training and test set

print('Training set score: {:.4f}'.format(rf.score(x_train, y_train)))
print('Test set score: {:.4f}'.format(rf.score(x_test, y_test)))
```

```
Training set score: 1.0000
Test set score: 0.9865
```

MODELLING AND EVALUATION

3. XGBoost

```
# Boosting (XGBoost)

from xgboost import XGBClassifier
xg = XGBClassifier(random_state=50)
xg.fit(x_train, y_train)

y_pred = xg.predict(x_test)
eval_classification(xg, y_pred, x_train, y_train, x_test, y_test)
```

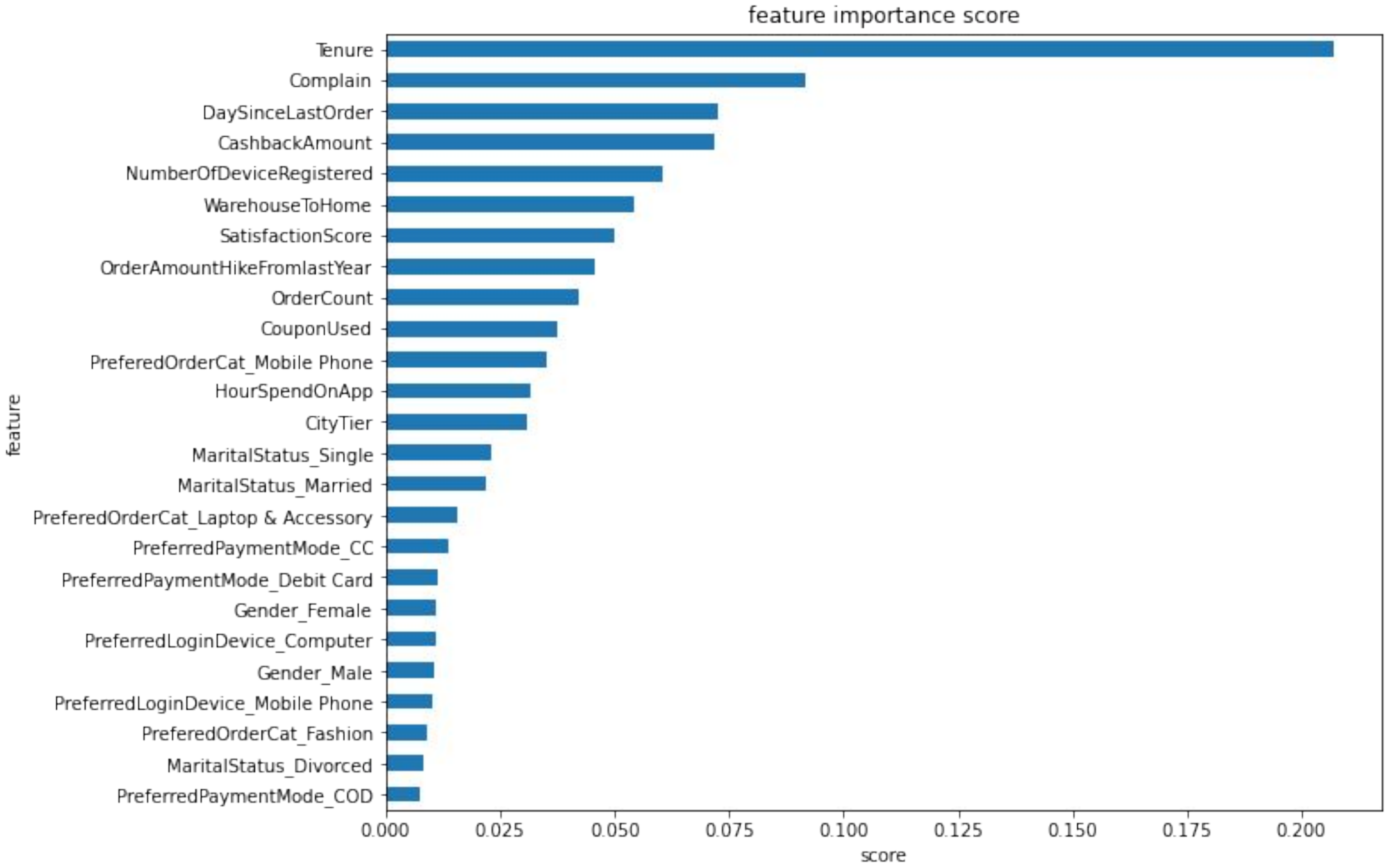
```
Accuracy (Test Set): 0.9161
Precision (Test Set): 0.9363
Recall (Test Set): 0.8959
F1-Score (Test Set): 0.9157
AUC: 0.92
```

```
# Cek overfitting
# print the scores on training and test set

print('Training set score: {:.4f}'.format(xg.score(x_train, y_train)))
print('Test set score: {:.4f}'.format(xg.score(x_test, y_test)))
```

```
Training set score: 0.9275
Test set score: 0.9161
```

MODELLING AND EVALUATION

3. XGBoost



We Care About
Your Future

HYPOTHESIS TESTING

Statement :
Apakah customer yang tidak churn cenderung memiliki rata-rata tenure yang lebih lama daripada customer yang churn?

Hipotesis :
H0 : mean tenure customer not churn <= mean tenure customer churn
H1 : mean tenure customer not churn > mean tenure customer churn

```
import scipy.stats as st
```

```
uji_t = st.ttest_ind(a=dfchurn['Tenure'], b=dfnotchurn['Tenure'])
p_value = uji_t.pvalue
print("p_value:", p_value)
```

```
p_value: 7.953120970974895e-144
```

Kesimpulan :
Karena pvalue < alpha 0.05, maka sudah cukup bukti untuk menolak H0. Jadi dapat disimpulkan bahwa memang benar customer yang tidak churn cenderung memiliki rata-rata tenure yang lebih lama dibandingkan dengan customer yang churn.

We Care About
Your Future

SUMMARY AND RECOMMENDATION

Summary

1. Sebanyak 912 dari 5350 (17.05%) customer memilih untuk churn.
2. Marial Status, Complain dan Mobile Phone memiliki korelasi positif terhadap penambahan Churn, sedangkan Tenure dan Day Last Order memiliki korelasi negatif terhadap penambahan Churn.
3. Hasil perbandingan model prediksi yang digunakan menunjukkan Random Forest memiliki nilai AUC tertinggi yaitu sebesar 0.99, namun jika dilihat dari akurasi data testing maupun training, model XGBoost lebih baik karena selisih perbedaannya lebih kecil (1,14%) dibandingkan dengan Random Forest (1.35%)
4. Pembayaran COD merupakan penyebab churn rate tinggi berdasarkan jenis pembayaran, setelah ditelusuri ternyata pembeli merasa kecewa karena ketidaksesuaian produk dengan deskripsi.

Recommendation

1. Mengevaluasi ulasan dan complain customer terhadap produk.
2. Intensitas penawaran voucher gratis ongkir untuk customer yang akan churn ditingkatkan.
3. Memantau perilaku customer yang akan churn dan memberikan penawaran khusus berupa membership.

We Care About
Your Future

Thanks For Your Attention.

Follow our social media on :

@data_bangalore Data Bangalore Data Bangalore Id