Machine Learning Assignment

21MBMB40

A V Vishnu Vardhan Reddy

MBA Business Analytics

Importing Libraries

In [15]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.preprocessing import OneHotEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import LabelEncoder
from sklearn.compose import ColumnTransformer
```

Importing Dataset

In [2]:
```
dataset = pd.read_csv('Churn_Modelling.csv')
dataset
```

Out[2]:

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure
0	1	15634602	Hargrave	619	France	Female	42	2
1	2	15647311	Hill	608	Spain	Female	41	1
2	3	15619304	Onio	502	France	Female	42	8
3	4	15701354	Boni	699	France	Female	39	1
4	5	15737888	Mitchell	850	Spain	Female	43	2
...	...	...	...	...	...	...	...	...
9995	9996	15606229	Obijiaku	771	France	Male	39	5
9996	9997	15569892	Johnstone	516	France	Male	35	10
9997	9998	15584532	Liu	709	France	Female	36	7
9998	9999	15682355	Sabbatini	772	Germany	Male	42	3
9999	10000	15628319	Walker	792	France	Female	28	4

10000 rows × 14 columns

In [3]:
```
dataset.isnull().sum()
```

Out[3]:
```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          0
Gender             0
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

In [6]:
```
dataset = dataset.drop(['RowNumber', 'CustomerId', 'Surname'], axis
dataset
```

Out[6]:

	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCar
0	619	France	Female	42	2	0.00	1	
1	608	Spain	Female	41	1	83807.86	1	
2	502	France	Female	42	8	159660.80	3	
3	699	France	Female	39	1	0.00	2	
4	850	Spain	Female	43	2	125510.82	1	
...	...	...	...	...	...	...	...	
9995	771	France	Male	39	5	0.00	2	
9996	516	France	Male	35	10	57369.61	1	
9997	709	France	Female	36	7	0.00	1	
9998	772	Germany	Male	42	3	75075.31	2	
9999	792	France	Female	28	4	130142.79	1	

10000 rows × 11 columns

Encoding categorical data

In [16]:
```
le=LabelEncoder()
```

In [19]:
```
dataset.Gender=le.fit_transform(dataset.Gender)
dataset.Geography=le.fit_transform(dataset.Geography)
```

In [10]:
```
dataset = pd.concat([dataset,Geography,Gender], axis=1)
```

In [20]:
```
dataset.Gender
```

Out[20]:
```
0       0
1       2
2       0
3       0
4       2
       ..
9995    0
9996    0
9997    0
9998    1
9999    0
Name: Geography, Length: 10000, dtype: int64
```

In [21]:
```
dataset.Geography
```

Out[21]:
```
0       0
1       2
2       0
3       0
4       2
       ..
9995    0
9996    0
9997    0
9998    1
9999    0
Name: Geography, Length: 10000, dtype: int64
```

Data Preprocessing

In [22]:
```
X =  dataset.drop(['Exited'], axis=1)
y = dataset['Exited']
```

Machine Learning Algorithm Training

In [23]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size
```

In [24]:
```
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators=200, random_state=
classifier.fit(X_train, y_train)
predictions = classifier.predict(X_test)
```

Machine Learning Algorithm Evaluation

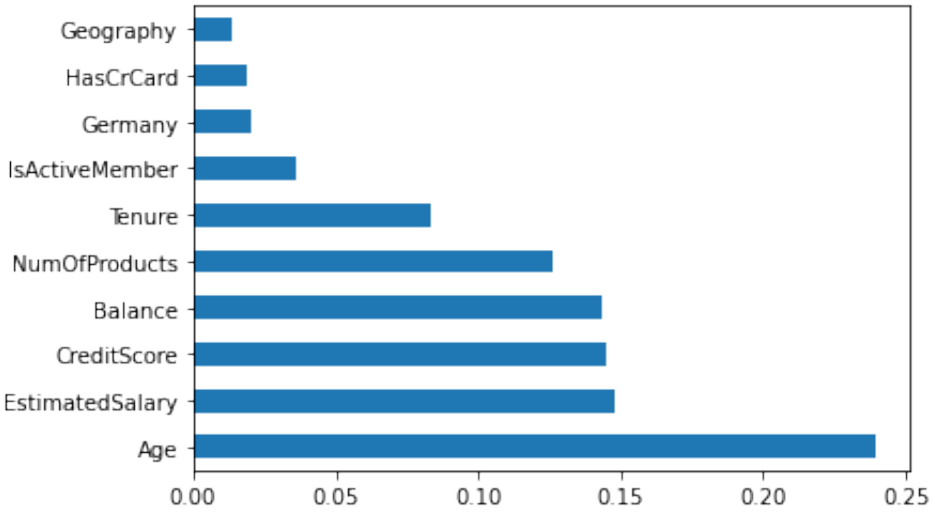
In [25]:
```
from sklearn.metrics import classification_report, accuracy_score
print(classification_report(y_test,predictions ))
print(accuracy_score(y_test, predictions ))
```

```
              precision    recall  f1-score   support

           0       0.89      0.96      0.92      1595
           1       0.77      0.52      0.62       405

    accuracy                           0.87      2000
   macro avg       0.83      0.74      0.77      2000
weighted avg       0.86      0.87      0.86      2000

0.871
```

Feature Evaluation

In [26]:
```
feat_importances = pd.Series(classifier.feature_importances_, index
feat_importances.nlargest(10).plot(kind='barh')
```

Out[26]: <AxesSubplot:>



Hence, the Model has an F1 Score of .92 and accuracy of 87% as seen above.

In [ ]: