CUSTOMER CHURN PREDICTION

File Edit View Insert Runtime Tools Help All changes saved

Share

+ Code + Text

Connect Gemini

```
import warnings
warnings.filterwarnings("ignore")
```

```
data = pd.read_csv("Churn_Modelling.csv")
data
```

| | RowNumber | CustomerId | Surname | CreditScore | Geography | Gender | Age | Tenure | Balance | NumOfProducts | HasCrCard | IsActiveMember | EstimatedSalar |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 15634602 | Hargrave | 619 | France | Female | 42 | 2 | 0.00 | 1 | 1 | 1 | 101348.8 |
| 1 | 2 | 15647311 | Hill | 608 | Spain | Female | 41 | 1 | 83807.86 | 1 | 0 | 1 | 112542.5 |
| 2 | 3 | 15619304 | Onio | 502 | France | Female | 42 | 8 | 159660.80 | 3 | 1 | 0 | 113931.5 |
| 3 | 4 | 15701354 | Boni | 699 | France | Female | 39 | 1 | 0.00 | 2 | 0 | 0 | 93826.6 |
| 4 | 5 | 15737888 | Mitchell | 850 | Spain | Female | 43 | 2 | 125510.82 | 1 | 1 | 1 | 79084.1 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 9995 | 9996 | 15606229 | Obijiaku | 771 | France | Male | 39 | 5 | 0.00 | 2 | 1 | 0 | 96270.6 |
| 9996 | 9997 | 15569892 | Johnstone | 516 | France | Male | 35 | 10 | 57369.61 | 1 | 1 | 1 | 101699.7 |
| 9997 | 9998 | 15584532 | Liu | 709 | France | Female | 36 | 7 | 0.00 | 1 | 0 | 1 | 42085.5 |
| 9998 | 9999 | 15682355 | Sabbatini | 772 | Germany | Male | 42 | 3 | 75075.31 | 2 | 1 | 0 | 92888.5 |
| 9999 | 10000 | 15628319 | Walker | 792 | France | Female | 28 | 4 | 130142.79 | 1 | 1 | 0 | 38190.7 |

10000 rows × 14 columns

CUSTOMER CHURN PREDICTION

File Edit View Insert Runtime Tools Help All changes saved

Share

+ Code + Text Connect Gemini

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 9998 | 9999 | 15682355 | Sabbatini | 772 | Germany | Male | 42 | 3 | 75075.31 | 2 | 1 | 0 | 92888.5 |
| 9999 | 10000 | 15628319 | Walker | 792 | France | Female | 28 | 4 | 130142.79 | 1 | 1 | 0 | 38190.7 |

10000 rows × 14 columns

```
[ ] data.head()
```

| | RowNumber | CustomerId | Surname | CreditScore | Geography | Gender | Age | Tenure | Balance | NumOfProducts | HasCrCard | IsActiveMember | EstimatedSalary | Ex |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 15634602 | Hargrave | 619 | France | Female | 42 | 2 | 0.00 | 1 | 1 | 1 | 101348.88 | |
| 1 | 2 | 15647311 | Hill | 608 | Spain | Female | 41 | 1 | 83807.86 | 1 | 0 | 1 | 112542.58 | |
| 2 | 3 | 15619304 | Onio | 502 | France | Female | 42 | 8 | 159660.80 | 3 | 1 | 0 | 113931.57 | |
| 3 | 4 | 15701354 | Boni | 699 | France | Female | 39 | 1 | 0.00 | 2 | 0 | 0 | 93826.63 | |
| 4 | 5 | 15737888 | Mitchell | 850 | Spain | Female | 43 | 2 | 125510.82 | 1 | 1 | 1 | 79084.10 | |

```
[ ] data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   RowNumber        10000 non-null  int64
 1   CustomerId       10000 non-null  int64
```

```
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   RowNumber        10000 non-null  int64
 1   CustomerId       10000 non-null  int64
 2   Surname          10000 non-null  object
 3   CreditScore      10000 non-null  int64
 4   Geography        10000 non-null  object
 5   Gender           10000 non-null  object
 6   Age              10000 non-null  int64
 7   Tenure           10000 non-null  int64
 8   Balance          10000 non-null  float64
 9   NumOfProducts    10000 non-null  int64
 10  HasCrCard        10000 non-null  int64
 11  IsActiveMember   10000 non-null  int64
 12  EstimatedSalary  10000 non-null  float64
 13  Exited           10000 non-null  int64
dtypes: float64(2), int64(9), object(3)
memory usage: 1.1+ MB
```

```
[ ] data.isnull().sum()
```

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          0
Gender             0
Age                0
Tenure             0
Balance            0
NumOfProducts      0
```

```
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

+ Code + Text

## Dropping of columns

```
[ ] data.columns
```

```
Index(['RowNumber', 'CustomerId', 'Surname', 'CreditScore', 'Geography',
       'Gender', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'HasCrCard',
       'IsActiveMember', 'EstimatedSalary', 'Exited'],
      dtype='object')
```

```
[ ] data = data.drop(['RowNumber', 'CustomerId', 'Surname'],axis=1)
    data
```

| | CreditScore | Geography | Gender | Age | Tenure | Balance | NumOfProducts | HasCrCard | IsActiveMember | EstimatedSalary | Exited |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 619 | France | Female | 42 | 2 | 0.00 | 1 | 1 | 1 | 101348.88 | 1 |
| 1 | 608 | Spain | Female | 41 | 1 | 83807.86 | 1 | 0 | 1 | 112542.58 | 0 |
| 2 | 502 | France | Female | 42 | 8 | 159660.80 | 3 | 1 | 0 | 113931.57 | 1 |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 9996 | 516 | France | Male | 35 | 10 | 57369.61 | 1 | 1 | 1 | 101699.77 | 0 |
| 9997 | 709 | France | Female | 36 | 7 | 0.00 | 1 | 0 | 1 | 42085.58 | 1 |
| 9998 | 772 | Germany | Male | 42 | 3 | 75075.31 | 2 | 1 | 0 | 92888.52 | 1 |
| 9999 | 792 | France | Female | 28 | 4 | 130142.79 | 1 | 1 | 0 | 38190.78 | 0 |

10000 rows × 11 columns

## changing categorical values to numerical values

```
[ ] data = pd.get_dummies(data,drop_first = True)
    data.head()
    data = data.astype(int)
    data
```

| | CreditScore | Age | Tenure | Balance | NumOfProducts | HasCrCard | IsActiveMember | EstimatedSalary | Exited | Geography_Germany | Geography_Spain | Gender_I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 619 | 42 | 2 | 0 | 1 | 1 | 1 | 101348 | 1 | 0 | 0 | |
| 1 | 608 | 41 | 1 | 83807 | 1 | 0 | 1 | 112542 | 0 | 0 | 1 | |
| 2 | 502 | 42 | 8 | 159660 | 3 | 1 | 0 | 113931 | 1 | 0 | 0 | |
| 3 | 699 | 39 | 1 | 0 | 2 | 0 | 0 | 93826 | 0 | 0 | 0 | |
| 4 | 850 | 43 | 2 | 125510 | 1 | 1 | 1 | 79084 | 0 | 0 | 1 | |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| 9995 | 771 | 39 | 5 | 0 | 2 | 1 | 0 | 96270 | 0 | 0 | 0 | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 9995 | 771 | 39 | 5 | 0 | 2 | 1 | 0 | 96270 | 0 | 0 | 0 |
| 9996 | 516 | 35 | 10 | 57369 | 1 | 1 | 1 | 101699 | 0 | 0 | 0 |
| 9997 | 709 | 36 | 7 | 0 | 1 | 0 | 1 | 42085 | 1 | 0 | 0 |
| 9998 | 772 | 42 | 3 | 75075 | 2 | 1 | 0 | 92888 | 1 | 1 | 0 |
| 9999 | 792 | 28 | 4 | 130142 | 1 | 1 | 0 | 38190 | 0 | 0 | 0 |

10000 rows × 12 columns

## Data visualization

```
[ ] data['Exited'].value_counts()
```

```
Exited
0    7963
1    2037
Name: count, dtype: int64
```

```
[ ] plt.figure(figsize =(8,6))
    sns.countplot(x='Exited',data = data)
```

```
<Axes: xlabel='Exited', ylabel='count'>
```

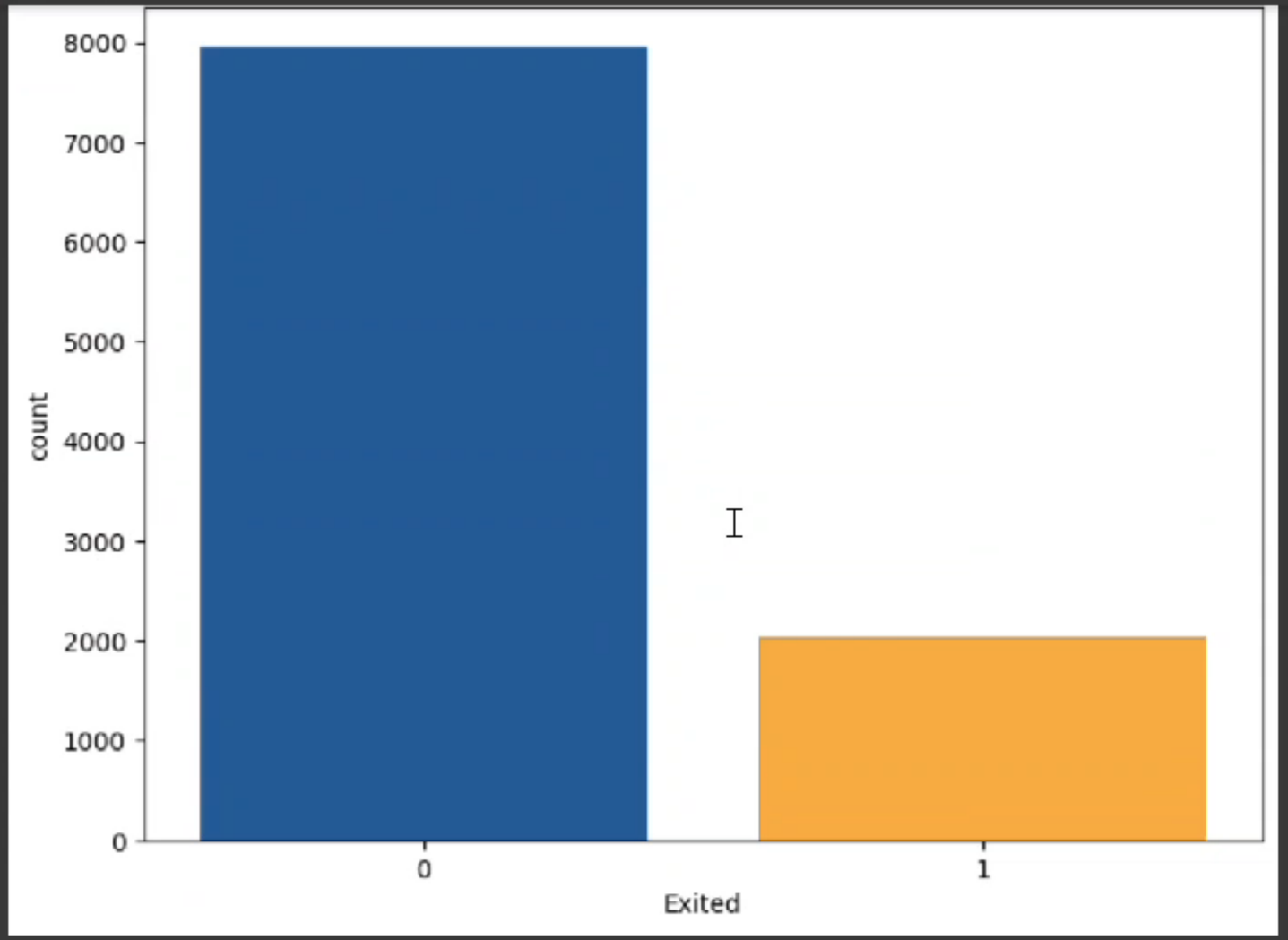
CUSTOMER CHURN PREDICTION

File Edit View Insert Runtime Tools Help All changes saved

Share

+ Code + Text

Connect Gemini




```
X = data.drop('Exited',axis=1)
y = data['Exited']
```

CUSTOMER CHURN PREDICTION

File Edit View Insert Runtime Tools Help All changes saved

Share

+ Code + Text Connect Gemini

```
X = data.drop('Exited',axis=1)
y = data['Exited']
```

```
!pip install imblearn
```

```
Requirement already satisfied: imblearn in c:\users\chatt\anaconda3\lib\site-packages (0.0)
Requirement already satisfied: imbalanced-learn in c:\users\chatt\anaconda3\lib\site-packages (from imblearn) (0.10.1)
Requirement already satisfied: numpy>=1.17.3 in c:\users\chatt\anaconda3\lib\site-packages (from imbalanced-learn->imblearn) (1.24.3)
Requirement already satisfied: scipy>=1.3.2 in c:\users\chatt\anaconda3\lib\site-packages (from imbalanced-learn->imblearn) (1.11.1)
Requirement already satisfied: scikit-learn>=1.0.2 in c:\users\chatt\anaconda3\lib\site-packages (from imbalanced-learn->imblearn) (1.3.0)
Requirement already satisfied: joblib>=1.1.1 in c:\users\chatt\anaconda3\lib\site-packages (from imbalanced-learn->imblearn) (1.2.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\chatt\anaconda3\lib\site-packages (from imbalanced-learn->imblearn) (2.2.0)
```

```
[ ] from sklearn.model_selection import train_test_split
    from sklearn.preprocessing import StandardScaler
    from sklearn.model_selection import train_test_split
    from sklearn.linear_model import LinearRegression , LogisticRegression
    from sklearn.metrics import r2_score
    from sklearn.tree import DecisionTreeClassifier
    from sklearn.ensemble import RandomForestClassifier, GradientBoostingRegressor
    from sklearn.neighbors import KNeighborsClassifier
    from sklearn.metrics import mean_squared_error
    from sklearn.preprocessing import StandardScaler
    from sklearn.preprocessing import LabelEncoder
    from sklearn.metrics import accuracy_score
    from sklearn.metrics import confusion_matrix
    from sklearn.metrics import precision_score, recall_score, f1_score
```

```
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import precision_score, recall_score, f1_score
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1, random_state=42)
print('Training Shape: ', X_train.shape)
print('Testing Shape: ', X_test.shape)
```

```
Training Shape:  (9000, 11)
Testing Shape:  (1000, 11)
```

```
[ ] scaler = StandardScaler()
    X_train_scaled = scaler.fit_transform(X_train)
    X_test_scaled = scaler.transform(X_test)
```

```
[ ] X_train_scaled
```

```
array([[-0.47944328,  0.19687202, -0.00234647, ..., -0.5761528 ,
        -0.57700814,  0.91105005],
       [ 1.04580853,  1.33803657,  1.03625698, ..., -0.5761528 ,
        -0.57700814, -1.09763453],
       [-0.85297437, -0.08841912,  1.03625698, ..., -0.5761528 ,
         1.73307782,  0.91105005],
       ...,
       [ 0.86941896, -0.08841912, -1.38715108, ..., -0.5761528 ,
        -0.57700814, -1.09763453],
       [ 0.16386025,  0.38706611,  1.03625698, ..., -0.5761528 ,
        -0.57700814,  0.91105005],
       [ 0.47513615,  1.14784248, -1.38715108, ...,  1.73565068,
        -0.57700814,  0.91105005]])
```

```
[ ] LR = LogisticRegression()
    LR.fit(X_train_scaled, y_train_classified)
```

```
LogisticRegression()
```

```
y_test_classified = [1 if value > threshold else 0 for value in y_test]
accuracy1 = LR.score(X_test_scaled, y_test_classified)
print("Model Accuracy:", accuracy1)
```

```
Model Accuracy: 0.809
```

## SVM

```
[ ] from sklearn import svm
    threshold = 0.5
    y_train_classified = [1 if value > threshold else 0 for value in y_train]
    svm = svm.SVC()
    svm.fit(X_train_scaled, y_train_classified)
```

```
SVC()
```

```
[ ] y_test_classified = [1 if value > threshold else 0 for value in y_test]
    accuracy2 = svm.score(X_test_scaled, y_test_classified)
```