In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
from sklearn.metrics import classification_report
```

In [4]:
```
df = pd.read_csv('../Machine Learning Project/online_shoppers_intention.csv')
```

In [5]:
```
df
```

Out[5]:

	Administrative	Administrative_Duration	Informational	Informational_Duration	ProductI
0	0	0.0	0	0.0	
1	0	0.0	0	0.0	
2	0	0.0	0	0.0	
3	0	0.0	0	0.0	
4	0	0.0	0	0.0	
...	...	...	...	...	
12325	3	145.0	0	0.0	
12326	0	0.0	0	0.0	
12327	0	0.0	0	0.0	
12328	4	75.0	0	0.0	
12329	0	0.0	0	0.0	

12330 rows × 18 columns

In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12330 entries, 0 to 12329
Data columns (total 18 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Administrative           12330 non-null  int64  
 1   Administrative_Duration  12330 non-null  float64
 2   Informational            12330 non-null  int64  
 3   Informational_Duration   12330 non-null  float64
 4   ProductRelated           12330 non-null  int64  
 5   ProductRelated_Duration  12330 non-null  float64
 6   BounceRates              12330 non-null  float64
 7   ExitRates                12330 non-null  float64
 8   PageValues               12330 non-null  float64
 9   SpecialDay               12330 non-null  float64
 10  Month                    12330 non-null  object 
 11  OperatingSystems         12330 non-null  int64  
 12  Browser                  12330 non-null  int64  
 13  Region                   12330 non-null  int64  
 14  TrafficType              12330 non-null  int64  
 15  VisitorType              12330 non-null  object 
```

```
 16  Weekend                  12330 non-null  bool
 17  Revenue                  12330 non-null  bool
dtypes: bool(2), float64(7), int64(7), object(2)
memory usage: 1.5+ MB
```

In [7]:
```
df.describe()
```

Out[7]:

	Administrative	Administrative_Duration	Informational	Informational_Duration	ProductI
count	12330.000000	12330.000000	12330.000000	12330.000000	12330.(
mean	2.315166	80.818611	0.503569	34.472398	31.
std	3.321784	176.779107	1.270156	140.749294	44.(
min	0.000000	0.000000	0.000000	0.000000	0.(
25%	0.000000	0.000000	0.000000	0.000000	7.(
50%	1.000000	7.500000	0.000000	0.000000	18.(
75%	4.000000	93.256250	0.000000	0.000000	38.(
max	27.000000	3398.750000	24.000000	2549.375000	705.(

In [8]:
```
df.isnull().sum() #no missing value
```

Out[8]:
```
Administrative             0
Administrative_Duration    0
Informational              0
Informational_Duration     0
ProductRelated             0
ProductRelated_Duration    0
BounceRates                0
ExitRates                  0
PageValues                 0
SpecialDay                 0
Month                      0
OperatingSystems           0
Browser                    0
Region                     0
TrafficType                0
VisitorType                0
Weekend                    0
Revenue                    0
dtype: int64
```

In [9]:
```
df['Revenue'] = df['Revenue'].astype(int) #clean data type: bool to int
```

In [10]:
```
df['Weekend'] = df['Weekend'].astype(int) #clean data type: bool to int
```

In [11]:
```
month = {'Feb':2, 'Mar':3,'May':5, 'June':6, 'Jul':7, 'Aug':8, 'Sep':9, 'Oct'
df['Month'] = df['Month'].map(month) #clean data type: str to int
```

In [12]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12330 entries, 0 to 12329
```

```
Data columns (total 18 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Administrative           12330 non-null  int64  
 1   Administrative_Duration  12330 non-null  float64
 2   Informational            12330 non-null  int64  
 3   Informational_Duration   12330 non-null  float64
 4   ProductRelated           12330 non-null  int64  
 5   ProductRelated_Duration  12330 non-null  float64
 6   BounceRates              12330 non-null  float64
 7   ExitRates                12330 non-null  float64
 8   PageValues               12330 non-null  float64
 9   SpecialDay               12330 non-null  float64
 10  Month                    12330 non-null  int64  
 11  OperatingSystems         12330 non-null  int64  
 12  Browser                  12330 non-null  int64  
 13  Region                   12330 non-null  int64  
 14  TrafficType              12330 non-null  int64  
 15  VisitorType              12330 non-null  object 
 16  Weekend                  12330 non-null  int64  
 17  Revenue                  12330 non-null  int64  
dtypes: float64(7), int64(10), object(1)
memory usage: 1.7+ MB
```

Encoding

In [13]:
```
df['VisitorType'] = df['VisitorType'].map({'Returning_Visitor':2, 'New_Visito
```

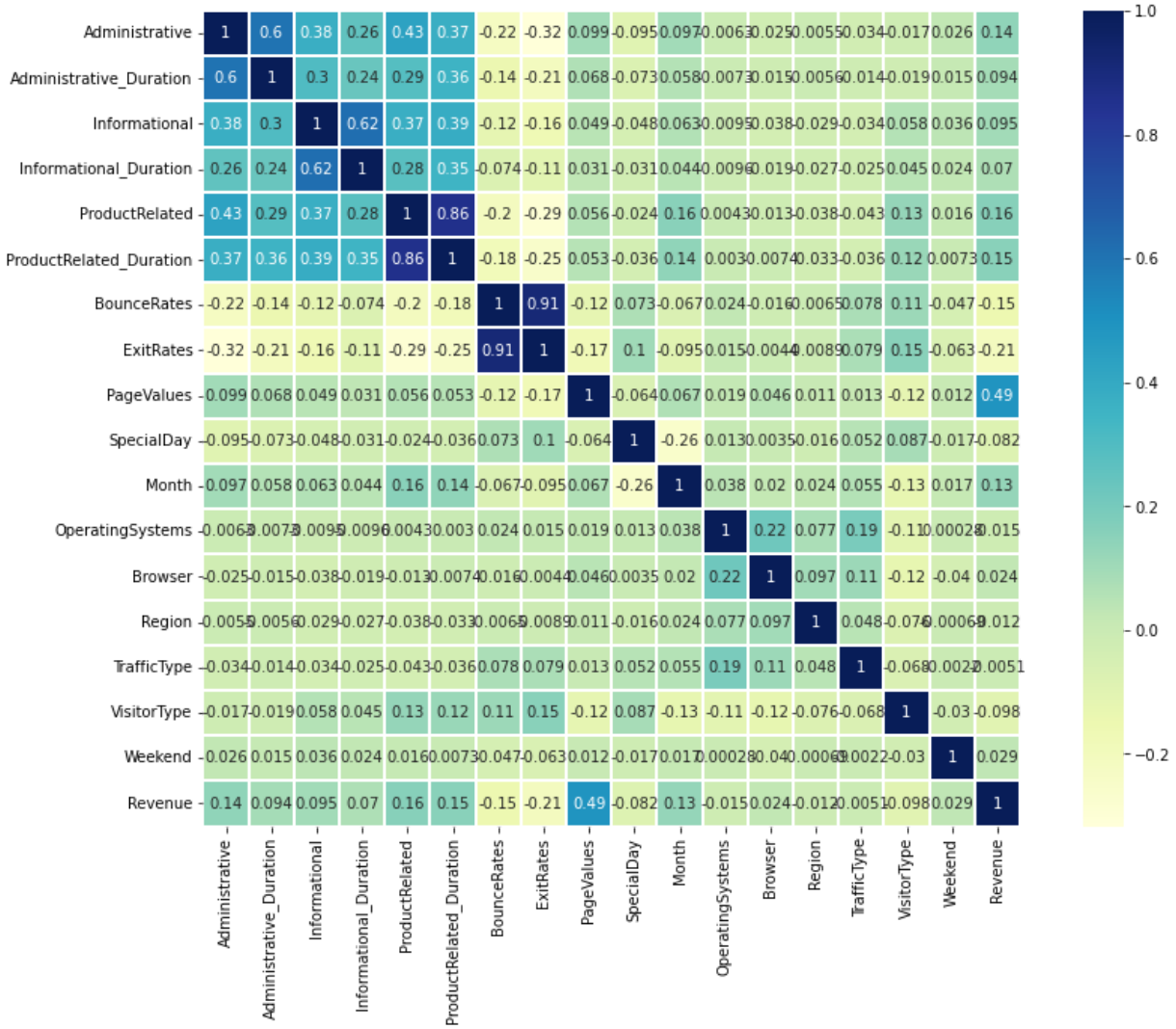
In [14]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12330 entries, 0 to 12329
Data columns (total 18 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Administrative           12330 non-null  int64  
 1   Administrative_Duration  12330 non-null  float64
 2   Informational            12330 non-null  int64  
 3   Informational_Duration   12330 non-null  float64
 4   ProductRelated           12330 non-null  int64  
 5   ProductRelated_Duration  12330 non-null  float64
 6   BounceRates              12330 non-null  float64
 7   ExitRates                12330 non-null  float64
 8   PageValues               12330 non-null  float64
 9   SpecialDay               12330 non-null  float64
 10  Month                    12330 non-null  int64  
 11  OperatingSystems         12330 non-null  int64  
 12  Browser                  12330 non-null  int64  
 13  Region                   12330 non-null  int64  
 14  TrafficType              12330 non-null  int64  
 15  VisitorType              12330 non-null  int64  
 16  Weekend                  12330 non-null  int64  
 17  Revenue                  12330 non-null  int64  
dtypes: float64(7), int64(11)
memory usage: 1.7 MB
```

In [15]:
```
plt.figure(figsize=(15,10))
sns.heatmap(df.corr(),
            linewidths=0.1,
            vmax=1.0,
```

```
            square=True,
            linecolor='white',
            cmap="YlGnBu",
            annot=True)
```

Out[15]: <AxesSubplot:>



In [16]:

```
a = df.corr()
b = a[['Revenue']]
b.sort_values(by='Revenue', ascending=False)
```

Out[16]:

	Revenue
Revenue	1.000000
PageValues	0.492569
ProductRelated	0.158538
ProductRelated_Duration	0.152373
Administrative	0.138917
Month	0.127372
Informational	0.095200
Administrative_Duration	0.093587
Informational_Duration	0.070345
Weekend	0.029295

	Revenue
Browser	0.023984
TrafficType	-0.005113
Region	-0.011595
OperatingSystems	-0.014668
SpecialDay	-0.082305
VisitorType	-0.098485
BounceRates	-0.150673
ExitRates	-0.207071

remove outliners

In [17]:
```
print('1º Quartile: ', df['ProductRelated_Duration'].quantile(q = 0.25))
print('2º Quartile: ', df['ProductRelated_Duration'].quantile(q = 0.50))
print('3º Quartile: ', df['ProductRelated_Duration'].quantile(q = 0.75))
print('4º Quartile: ', df['ProductRelated_Duration'].quantile(q = 1.00))
#Calculate the outliers:
  # Interquartile range, IQR = Q3 - Q1
  # lower 1.5*IQR whisker = Q1 - 1.5 * IQR
  # Upper 1.5*IQR whisker = Q3 + 1.5 * IQR

print('Duration above: ', df['ProductRelated_Duration'].quantile(q = 0.75) +
                      1.5*(df['ProductRelated_Duration'].quantile(q = 0.75) -
```

```
1º Quartile:  184.1375
2º Quartile:  598.9369047499999
3º Quartile:  1464.1572135000001
4º Quartile:  63973.52223
Duration above:  3384.1867837500004 are outliers
```

In [18]:
```
print('1º Quartile: ', df['Administrative_Duration'].quantile(q = 0.25))
print('2º Quartile: ', df['Administrative_Duration'].quantile(q = 0.50))
print('3º Quartile: ', df['Administrative_Duration'].quantile(q = 0.75))
print('4º Quartile: ', df['Administrative_Duration'].quantile(q = 1.00))
#Calculate the outliers:
  # Interquartile range, IQR = Q3 - Q1
  # lower 1.5*IQR whisker = Q1 - 1.5 * IQR
  # Upper 1.5*IQR whisker = Q3 + 1.5 * IQR

print('Duration above: ', df['Administrative_Duration'].quantile(q = 0.75) +
                      1.5*(df['Administrative_Duration'].quantile(q = 0.75) -
```

```
1º Quartile:  0.0
2º Quartile:  7.5
3º Quartile:  93.25625
4º Quartile:  3398.75
Duration above:  233.14062499999997 are outliers
```

In [19]:
```
print('1º Quartile: ', df['Informational_Duration'].quantile(q = 0.25))
print('2º Quartile: ', df['Informational_Duration'].quantile(q = 0.50))
print('3º Quartile: ', df['Informational_Duration'].quantile(q = 0.75))
print('4º Quartile: ', df['Informational_Duration'].quantile(q = 1.00))
#Calculate the outliers:
  # Interquartile range, IQR = Q3 - Q1
```

```
  # lower 1.5*IQR whisker = Q1 - 1.5 * IQR
  # Upper 1.5*IQR whisker = Q3 + 1.5 * IQR

print('Duration above: ', df['Informational_Duration'].quantile(q = 0.75) +
                      1.5*(df['Informational_Duration'].quantile(q = 0.75) - (
```

```
1º Quartile:  0.0
2º Quartile:  0.0
3º Quartile:  0.0
4º Quartile:  2549.375
Duration above:  0.0 are outliers
```

In [20]:

```
df = df[df.ProductRelated_Duration < 3384.18]
df = df[df.Administrative_Duration < 233.14]
```

In [21]:

```
df.describe()
```

Out[21]:

	Administrative	Administrative_Duration	Informational	Informational_Duration	ProductF
count	10467.000000	10467.000000	10467.000000	10467.000000	10467.(
mean	1.604662	36.259147	0.337346	20.959281	21
std	2.442930	56.192433	0.976302	102.611044	22.(
min	0.000000	0.000000	0.000000	0.000000	0.(
25%	0.000000	0.000000	0.000000	0.000000	6.(
50%	0.000000	0.000000	0.000000	0.000000	15.(
75%	3.000000	58.033333	0.000000	0.000000	30.(
max	19.000000	233.083333	16.000000	2252.033333	223.(

In [22]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 10467 entries, 0 to 12329
Data columns (total 18 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Administrative           10467 non-null  int64
 1   Administrative_Duration  10467 non-null  float64
 2   Informational            10467 non-null  int64
 3   Informational_Duration   10467 non-null  float64
 4   ProductRelated           10467 non-null  int64
 5   ProductRelated_Duration  10467 non-null  float64
 6   BounceRates              10467 non-null  float64
 7   ExitRates                10467 non-null  float64
 8   PageValues               10467 non-null  float64
 9   SpecialDay               10467 non-null  float64
 10  Month                    10467 non-null  int64
 11  OperatingSystems         10467 non-null  int64
 12  Browser                  10467 non-null  int64
 13  Region                   10467 non-null  int64
 14  TrafficType              10467 non-null  int64
 15  VisitorType              10467 non-null  int64
 16  Weekend                  10467 non-null  int64
 17  Revenue                  10467 non-null  int64
dtypes: float64(7), int64(11)
memory usage: 1.5 MB
```

train_test_split

In [23]:
```
X = df.drop(columns='Revenue', axis=1)
y = df['Revenue']
```

In [24]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,train_size=0.8,rando
print("Input Training:",X_train.shape)
print("Input Test:",X_test.shape)
print("Output Training:",y_train.shape)
print("Output Test:",y_test.shape)
```

```
Input Training: (8373, 17)
Input Test: (2094, 17)
Output Training: (8373,)
Output Test: (2094,)
```

modelling

In [33]:
```
def evaluate_model(model, x_test, y_test):
    from sklearn import metrics

    # Predict Test Data
    y_pred = model.predict(x_test)

    # Calculate accuracy, precision, recall, f1-score, and kappa score
    acc = metrics.accuracy_score(y_test, y_pred)
    prec = metrics.precision_score(y_test, y_pred)
    rec = metrics.recall_score(y_test, y_pred)
    f1 = metrics.f1_score(y_test, y_pred)
    kappa = metrics.cohen_kappa_score(y_test, y_pred)

    # Calculate area under curve (AUC)
    y_pred_proba = model.predict_proba(x_test)[::,1]
    fpr, tpr, _ = metrics.roc_curve(y_test, y_pred_proba)
    auc = metrics.roc_auc_score(y_test, y_pred_proba)

    # Display confussion matrix
    cm = metrics.confusion_matrix(y_test, y_pred)

    return {'acc': acc, 'prec': prec, 'rec': rec, 'f1': f1, 'kappa': kappa,
            'fpr': fpr, 'tpr': tpr, 'auc': auc, 'cm': cm}
```

In [34]:
```
from sklearn import tree

# Building Decision Tree model
dtc = tree.DecisionTreeClassifier(random_state=0)
dtc.fit(X_train, y_train)
```

Out[34]:
```
DecisionTreeClassifier(random_state=0)
```

In [35]:
```
# Evaluate Model
dtc_eval = evaluate_model(dtc, X_test, y_test)

# Print result
```

```
print('Accuracy:', dtc_eval['acc'])
print('Precision:', dtc_eval['prec'])
print('Recall:', dtc_eval['rec'])
print('F1 Score:', dtc_eval['f1'])
print('Cohens Kappa Score:', dtc_eval['kappa'])
print('Area Under Curve:', dtc_eval['auc'])
print('Confusion Matrix:\n', dtc_eval['cm'])
```

```
Accuracy: 0.8853868194842407
Precision: 0.5618729096989966
Recall: 0.6064981949458483
F1 Score: 0.5833333333333334
Cohens Kappa Score: 0.5170005132157871
Area Under Curve: 0.7672006659924618
Confusion Matrix:
 [[1686  131]
 [ 109  168]]
```

In [36]:

```
from sklearn.ensemble import RandomForestClassifier

# Building Random Forest model
rf = RandomForestClassifier(random_state=0)
rf.fit(X_train, y_train)
```

Out[36]:

```
RandomForestClassifier(random_state=0)
```

In [37]:

```
# Evaluate Model
rf_eval = evaluate_model(rf, X_test, y_test)

# Print result
print('Accuracy:', rf_eval['acc'])
print('Precision:', rf_eval['prec'])
print('Recall:', rf_eval['rec'])
print('F1 Score:', rf_eval['f1'])
print('Cohens Kappa Score:', rf_eval['kappa'])
print('Area Under Curve:', rf_eval['auc'])
print('Confusion Matrix:\n', rf_eval['cm'])
```

```
Accuracy: 0.9221585482330468
Precision: 0.7567567567567568
Recall: 0.6064981949458483
F1 Score: 0.6733466933867736
Cohens Kappa Score: 0.629769675828002
Area Under Curve: 0.9322175840288968
Confusion Matrix:
 [[1763   54]
 [ 109  168]]
```

In [38]:

```
from sklearn.naive_bayes import GaussianNB

# Building Naive Bayes model
nb = GaussianNB()
nb.fit(X_train, y_train)
```

Out[38]:

```
GaussianNB()
```

In [39]:

```
# Evaluate Model
nb_eval = evaluate_model(nb, X_test, y_test)

# Print result
```

```
print('Accuracy:', nb_eval['acc'])
print('Precision:', nb_eval['prec'])
print('Recall:', nb_eval['rec'])
print('F1 Score:', nb_eval['f1'])
print('Cohens Kappa Score:', nb_eval['kappa'])
print('Area Under Curve:', nb_eval['auc'])
print('Confusion Matrix:\n', nb_eval['cm'])
```

```
Accuracy: 0.8428844317096467
Precision: 0.43333333333333335
Recall: 0.6101083032490975
F1 Score: 0.5067466266866567
Cohens Kappa Score: 0.4164799032387573
Area Under Curve: 0.84550941866726
Confusion Matrix:
 [[1596  221]
 [ 108  169]]
```

In [40]:

```
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression(random_state=0)
lr.fit(X_train, y_train)
```

```
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/sklearn/linear_model/_
logistic.py:814: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regress
ion
  n_iter_i = _check_optimize_result(
```

Out[40]:

```
LogisticRegression(random_state=0)
```

In [41]:

```
# Evaluate Model
lr_eval = evaluate_model(lr, X_test, y_test)

# Print result
print('Accuracy:', lr_eval['acc'])
print('Precision:', lr_eval['prec'])
print('Recall:', lr_eval['rec'])
print('F1 Score:', lr_eval['f1'])
print('Cohens Kappa Score:', lr_eval['kappa'])
print('Area Under Curve:', lr_eval['auc'])
print('Confusion Matrix:\n', lr_eval['cm'])
```

```
Accuracy: 0.9083094555873925
Precision: 0.7741935483870968
Recall: 0.4332129963898917
F1 Score: 0.5555555555555556
Cohens Kappa Score: 0.5089418104448553
Area Under Curve: 0.8392597787840075
Confusion Matrix:
 [[1782   35]
 [ 157  120]]
```

In [42]:

```
import xgboost
xgb = xgboost.XGBClassifier()
xgb.fit(X_train, y_train)
```

```
[15:21:45] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
```

```
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
```

```
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
```

Out[42]:
```
XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
              colsample_bynode=1, colsample_bytree=1, enable_categorical=Fals
e,
              gamma=0, gpu_id=-1, importance_type=None,
              interaction_constraints='', learning_rate=0.300000012,
              max_delta_step=0, max_depth=6, min_child_weight=1, missing=nan,
              monotone_constraints='()', n_estimators=100, n_jobs=8,
              num_parallel_tree=1, predictor='auto', random_state=0,
              reg_alpha=0, reg_lambda=1, scale_pos_weight=1, subsample=1,
              tree_method='exact', validate_parameters=1, verbosity=None)
```

In [43]:
```
# Evaluate Model
xgb_eval = evaluate_model(xgb, X_test, y_test)

# Print result
print('Accuracy:', xgb_eval['acc'])
print('Precision:', xgb_eval['prec'])
print('Recall:', xgb_eval['rec'])
print('F1 Score:', xgb_eval['f1'])
print('Cohens Kappa Score:', xgb_eval['kappa'])
print('Area Under Curve:', xgb_eval['auc'])
print('Confusion Matrix:\n', xgb_eval['cm'])
```

```
Accuracy: 0.9221585482330468
Precision: 0.7478260869565218
Recall: 0.6209386281588448
F1 Score: 0.6785009861932939
Cohens Kappa Score: 0.6346519837557454
Area Under Curve: 0.9331345157746036
Confusion Matrix:
 [[1759   58]
 [ 105  172]]
```

In [44]:
```
from sklearn.ensemble import AdaBoostClassifier
ada = AdaBoostClassifier(n_estimators=100, random_state=0)
ada.fit(X_train, y_train)
```

Out[44]:
```
AdaBoostClassifier(n_estimators=100, random_state=0)
```

In [45]:
```
# Evaluate Model
ada_eval = evaluate_model(ada, X_test, y_test)

# Print result
print('Accuracy:', ada_eval['acc'])
print('Precision:', ada_eval['prec'])
print('Recall:', ada_eval['rec'])
print('F1 Score:', ada_eval['f1'])
print('Cohens Kappa Score:', ada_eval['kappa'])
print('Area Under Curve:', ada_eval['auc'])
print('Confusion Matrix:\n', ada_eval['cm'])
```

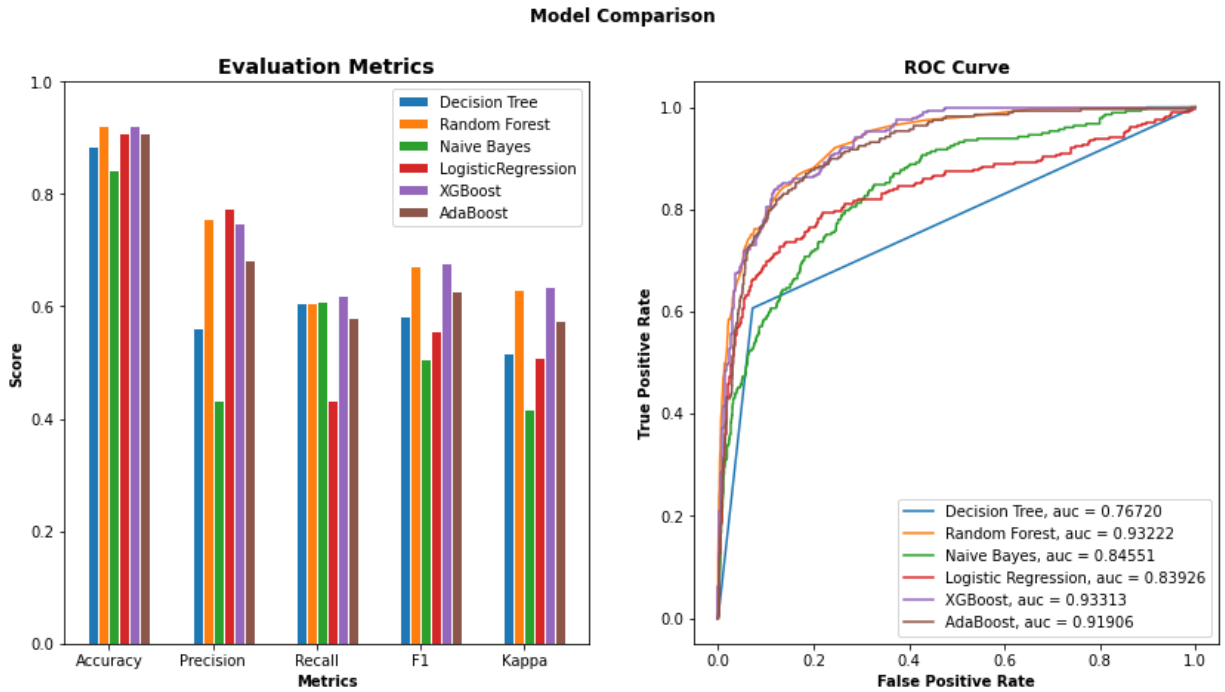
```
Accuracy: 0.9087870105062082
Precision: 0.6822033898305084
Recall: 0.5812274368231047
F1 Score: 0.6276803118908382
Cohens Kappa Score: 0.5760855049084346
Area Under Curve: 0.919062643425808
Confusion Matrix:
 [[1742   75]
 [ 116  161]]
```

initial result

In [46]:

```
# Intitialize figure with two plots
fig, (ax1, ax2) = plt.subplots(1, 2)
fig.suptitle('Model Comparison', fontsize=12, fontweight='bold')
fig.set_figheight(7)
fig.set_figwidth(14)
fig.set_facecolor('white')

# First plot
## set bar size
barWidth = 0.1
dtc_score = [dtc_eval['acc'], dtc_eval['prec'], dtc_eval['rec'], dtc_eval['f1
rf_score = [rf_eval['acc'], rf_eval['prec'], rf_eval['rec'], rf_eval['f1'], r
nb_score = [nb_eval['acc'], nb_eval['prec'], nb_eval['rec'], nb_eval['f1'], nb
lr_score = [lr_eval['acc'], lr_eval['prec'], lr_eval['rec'], lr_eval['f1'], lr
xgb_score = [xgb_eval['acc'], xgb_eval['prec'], xgb_eval['rec'], xgb_eval['f1
ada_score = [ada_eval['acc'], ada_eval['prec'], ada_eval['rec'], ada_eval['f1

## Set position of bar on X axis
r1 = np.arange(len(dtc_score))
r2 = [x + barWidth for x in r1]
r3 = [x + barWidth for x in r2]
r4 = [x + barWidth for x in r3]
r5 = [x + barWidth for x in r4]
r6 = [x + barWidth for x in r5]

## Make the plot
ax1.bar(r1, dtc_score, width=barWidth, edgecolor='white', label='Decision Tre
ax1.bar(r2, rf_score, width=barWidth, edgecolor='white', label='Random Forest
ax1.bar(r3, nb_score, width=barWidth, edgecolor='white', label='Naive Bayes')
ax1.bar(r4, lr_score, width=barWidth, edgecolor='white', label='LogisticRegre
ax1.bar(r5, xgb_score, width=barWidth, edgecolor='white', label='XGBoost')
ax1.bar(r6, ada_score, width=barWidth, edgecolor='white', label='AdaBoost')

## Configure x and y axis
ax1.set_xlabel('Metrics', fontweight='bold')
labels = ['Accuracy', 'Precision', 'Recall', 'F1', 'Kappa']
ax1.set_xticks([r + (barWidth * 1.5) for r in range(len(dtc_score))], )
ax1.set_xticklabels(labels)
ax1.set_ylabel('Score', fontweight='bold')
ax1.set_ylim(0, 1)

## Create legend & title
ax1.set_title('Evaluation Metrics', fontsize=14, fontweight='bold')
ax1.legend()

# Second plot
## Comparing ROC Curve
ax2.plot(dtc_eval['fpr'], dtc_eval['tpr'], label='Decision Tree, auc = {:0.5f
ax2.plot(rf_eval['fpr'], rf_eval['tpr'], label='Random Forest, auc = {:0.5f}'
```

```
ax2.plot(nb_eval['fpr'], nb_eval['tpr'], label='Naive Bayes, auc = {:0.5f}'.f
ax2.plot(lr_eval['fpr'], lr_eval['tpr'], label='Logistic Regression, auc = {:
ax2.plot(xgb_eval['fpr'], xgb_eval['tpr'], label='XGBoost, auc = {:0.5f}'.for
ax2.plot(ada_eval['fpr'], ada_eval['tpr'], label='AdaBoost, auc = {:0.5f}'.fo

## Configure x and y axis
ax2.set_xlabel('False Positive Rate', fontweight='bold')
ax2.set_ylabel('True Positive Rate', fontweight='bold')

## Create legend & title
ax2.set_title('ROC Curve', fontsize=12, fontweight='bold')
ax2.legend(loc=4)

plt.show()
```

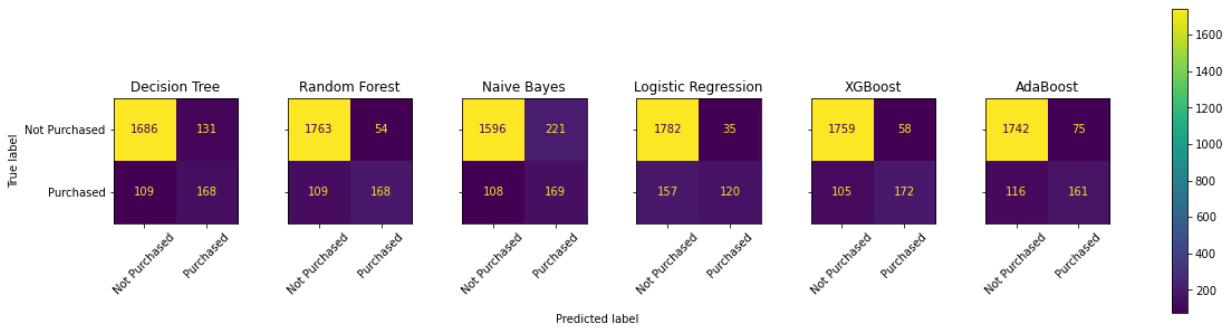


In [47]:

```
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay, accurac
classifiers = {
    "Decision Tree": tree.DecisionTreeClassifier(random_state=0),
    "Random Forest": RandomForestClassifier(random_state=0),
    "Naive Bayes": GaussianNB(),
    "Logistic Regression": LogisticRegression(),
    "XGBoost": xgboost.XGBClassifier(),
    "AdaBoost": AdaBoostClassifier(n_estimators=100, random_state=0),
    }

f, axes = plt.subplots(1, 6, figsize=(20, 5), sharey='row')

for i, (key, classifier) in enumerate(classifiers.items()):
    y_pred = classifier.fit(X_train, y_train).predict(X_test)
    cf_matrix = confusion_matrix(y_test, y_pred)
    print(key, " \n Accuracy:",accuracy_score(y_test,y_pred),"\n F-score",f1_
    disp = ConfusionMatrixDisplay(cf_matrix,
                                  display_labels=["Not Purchased","Purchased"
    disp.plot(ax=axes[i], xticks_rotation=45)
    disp.ax_.set_title(key)
    disp.im_.colorbar.remove()
    disp.ax_.set_xlabel('')
    if i!=0:
        disp.ax_.set_ylabel('')
```

```
f.text(0.4, 0.1, 'Predicted label', ha='left')
plt.subplots_adjust(wspace=0.40, hspace=0.1)


f.colorbar(disp.im_, ax=axes)
plt.show()
```

```
Decision Tree
 Accuracy: 0.8853868194842407
 F-score 0.5833333333333334
Random Forest
 Accuracy: 0.9221585482330468
 F-score 0.6733466933867736
Naive Bayes
 Accuracy: 0.8428844317096467
 F-score 0.5067466266866567
Logistic Regression
 Accuracy: 0.9083094555873925
 F-score 0.5555555555555556
[15:21:47] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/sklearn/linear_model/_
logistic.py:814: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regress
ion
  n_iter_i = _check_optimize_result(
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
XGBoost
 Accuracy: 0.9221585482330468
 F-score 0.6785009861932939
AdaBoost
 Accuracy: 0.9087870105062082
 F-score 0.6276803118908382
```



SMOTE

In [48]:

```
from imblearn.combine import SMOTEENN
```

In [49]:
```
sm = SMOTEENN()
X_resampled1, y_resampled1 = sm.fit_resample(X,y)
```

In [50]:
```
X_train, X_test, y_train, y_test=train_test_split(X_resampled1, y_resampled1,
```

In [51]:
```
dtc = tree.DecisionTreeClassifier(random_state=0)
dtc.fit(X_train, y_train)
```

Out[51]:
```
DecisionTreeClassifier(random_state=0)
```

In [52]:
```
# Evaluate Model
dtc_eval = evaluate_model(dtc, X_test, y_test)

# Print result
print('Accuracy:', dtc_eval['acc'])
print('Precision:', dtc_eval['prec'])
print('Recall:', dtc_eval['rec'])
print('F1 Score:', dtc_eval['f1'])
print('Cohens Kappa Score:', dtc_eval['kappa'])
print('Area Under Curve:', dtc_eval['auc'])
print('Confusion Matrix:\n', dtc_eval['cm'])
```

```
Accuracy: 0.949010399194901
Precision: 0.9557843731072078
Recall: 0.9523234761617381
F1 Score: 0.9540507859733979
Cohens Kappa Score: 0.8967791670292551
Area Under Curve: 0.9485937622500533
Confusion Matrix:
 [[1251   73]
 [  79 1578]]
```

In [53]:
```
rf = RandomForestClassifier(random_state=0)
rf.fit(X_train, y_train)
```

Out[53]:
```
RandomForestClassifier(random_state=0)
```

In [54]:
```
# Evaluate Model
rf_eval = evaluate_model(rf, X_test, y_test)

# Print result
print('Accuracy:', rf_eval['acc'])
print('Precision:', rf_eval['prec'])
print('Recall:', rf_eval['rec'])
print('F1 Score:', rf_eval['f1'])
print('Cohens Kappa Score:', rf_eval['kappa'])
print('Area Under Curve:', rf_eval['auc'])
print('Confusion Matrix:\n', rf_eval['cm'])
```

```
Accuracy: 0.9647769204964777
Precision: 0.9743276283618582
Recall: 0.9619794809897405
F1 Score: 0.9681141815973276
Cohens Kappa Score: 0.9287771782970008
Area Under Curve: 0.994492603930592
Confusion Matrix:
```

```
[[1282   42]
[  63 1594]]
```

In [55]:
```
nb = GaussianNB()
nb.fit(X_train, y_train)
```

Out[55]:
```
GaussianNB()
```

In [56]:
```
# Evaluate Model
nb_eval = evaluate_model(nb, X_test, y_test)

# Print result
print('Accuracy:', nb_eval['acc'])
print('Precision:', nb_eval['prec'])
print('Recall:', nb_eval['rec'])
print('F1 Score:', nb_eval['f1'])
print('Cohens Kappa Score:', nb_eval['kappa'])
print('Area Under Curve:', nb_eval['auc'])
print('Confusion Matrix:\n', nb_eval['cm'])
```

```
Accuracy: 0.8782287822878229
Precision: 0.8655367231638418
Recall: 0.9245624622812312
F1 Score: 0.8940764517070324
Cohens Kappa Score: 0.7512468084118391
Area Under Curve: 0.9475797996962442
Confusion Matrix:
 [[1086  238]
 [ 125 1532]]
```

In [57]:
```
lr = LogisticRegression(random_state=0)
lr.fit(X_train, y_train)
```

```
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/sklearn/linear_model/_
logistic.py:814: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regress
ion
  n_iter_i = _check_optimize_result(
```

Out[57]:
```
LogisticRegression(random_state=0)
```

In [58]:
```
# Evaluate Model
lr_eval = evaluate_model(lr, X_test, y_test)

# Print result
print('Accuracy:', lr_eval['acc'])
print('Precision:', lr_eval['prec'])
print('Recall:', lr_eval['rec'])
print('F1 Score:', lr_eval['f1'])
print('Cohens Kappa Score:', lr_eval['kappa'])
print('Area Under Curve:', lr_eval['auc'])
print('Confusion Matrix:\n', lr_eval['cm'])
```

```
Accuracy: 0.9050654142905066
Precision: 0.9673469387755103
Recall: 0.8581774290887145
```

```
F1 Score: 0.9094979213303486
Cohens Kappa Score: 0.8104220959652423
Area Under Curve: 0.9515353704051474
Confusion Matrix:
 [[1276   48]
 [ 235 1422]]
```

In [59]:

```
xgb = xgboost.XGBClassifier()
xgb.fit(X_train, y_train)
```

```
[15:21:55] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_1643227205751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluation metric used with the objective 'binary:logistic' was changed from 'error' to 'logloss'. Explicitly set eval_metric if you'd like to restore the old behavior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:1224: UserWarning: The use of label encoder in XGBClassifier is deprecated and will be removed in a future release. To remove this warning, do the following: 1) Pass option use_label_encoder=False when constructing XGBClassifier object; and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ..., [num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
```

Out[59]:

```
XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
              colsample_bynode=1, colsample_bytree=1, enable_categorical=False,
              gamma=0, gpu_id=-1, importance_type=None,
              interaction_constraints='', learning_rate=0.300000012,
              max_delta_step=0, max_depth=6, min_child_weight=1, missing=nan,
              monotone_constraints='()', n_estimators=100, n_jobs=8,
              num_parallel_tree=1, predictor='auto', random_state=0,
              reg_alpha=0, reg_lambda=1, scale_pos_weight=1, subsample=1,
              tree_method='exact', validate_parameters=1, verbosity=None)
```

In [60]:

```
# Evaluate Model
xgb_eval = evaluate_model(xgb, X_test, y_test)

# Print result
print('Accuracy:', xgb_eval['acc'])
print('Precision:', xgb_eval['prec'])
print('Recall:', xgb_eval['rec'])
print('F1 Score:', xgb_eval['f1'])
print('Cohens Kappa Score:', xgb_eval['kappa'])
print('Area Under Curve:', xgb_eval['auc'])
print('Confusion Matrix:\n', xgb_eval['cm'])
```

```
Accuracy: 0.9718215363971822
Precision: 0.9775349119611415
Recall: 0.971635485817743
F1 Score: 0.9745762711864406
Cohens Kappa Score: 0.942974211728997
Area Under Curve: 0.9938638058442897
Confusion Matrix:
 [[1287   37]
 [  47 1610]]
```

In [61]:

```
ada = AdaBoostClassifier(n_estimators=100, random_state=0)
ada.fit(X_train, y_train)
```

Out[61]:

```
AdaBoostClassifier(n_estimators=100, random_state=0)
```

In [62]:

```
# Evaluate Model
ada_eval = evaluate_model(ada, X_test, y_test)

# Print result
print('Accuracy:', ada_eval['acc'])
print('Precision:', ada_eval['prec'])
print('Recall:', ada_eval['rec'])
print('F1 Score:', ada_eval['f1'])
print('Cohens Kappa Score:', ada_eval['kappa'])
print('Area Under Curve:', ada_eval['auc'])
print('Confusion Matrix:\n', ada_eval['cm'])
```

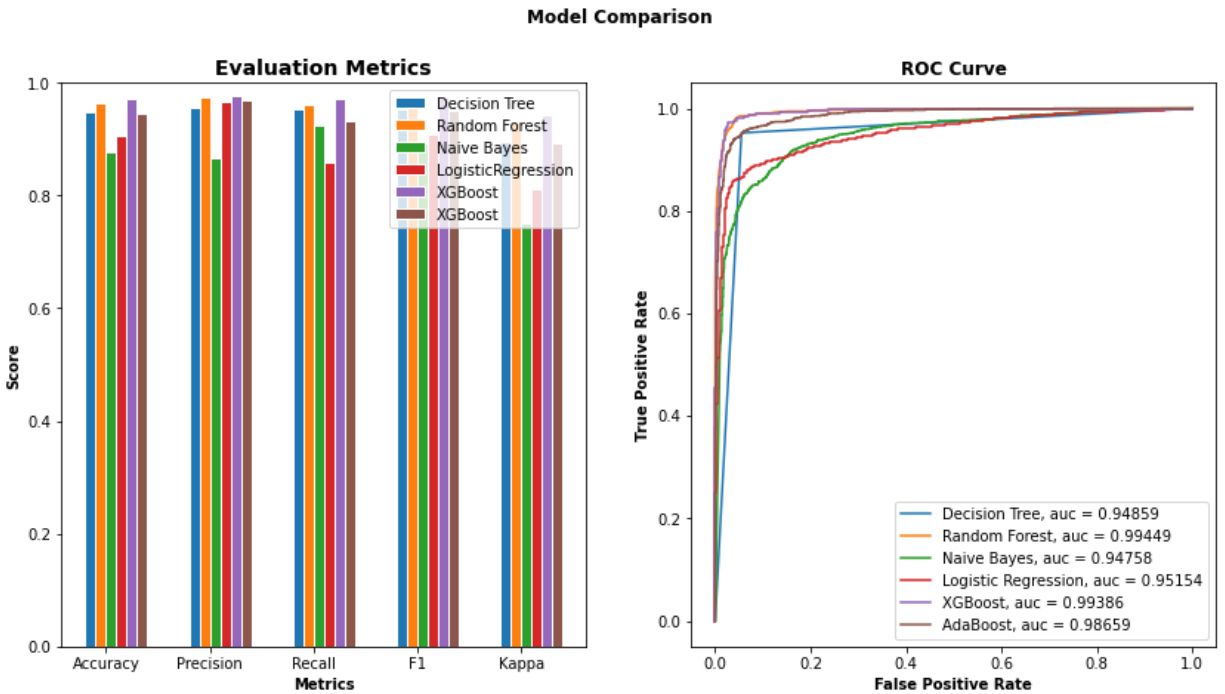
```
Accuracy: 0.9466621938946662
Precision: 0.9698870765370138
Recall: 0.9330114665057333
F1 Score: 0.9510919717010151
Cohens Kappa Score: 0.892490442226363
Area Under Curve: 0.9865946811749841
Confusion Matrix:
 [[1276   48]
 [ 111 1546]]
```

results after SMOTE

In [63]:

```
# Intitialize figure with two plots
fig, (ax1, ax2) = plt.subplots(1, 2)
fig.suptitle('Model Comparison', fontsize=12, fontweight='bold')
fig.set_figheight(7)
fig.set_figwidth(14)
fig.set_facecolor('white')

# First plot
## set bar size
barWidth = 0.1
dtc_score = [dtc_eval['acc'], dtc_eval['prec'], dtc_eval['rec'], dtc_eval['f1
rf_score = [rf_eval['acc'], rf_eval['prec'], rf_eval['rec'], rf_eval['f1'], r
nb_score = [nb_eval['acc'], nb_eval['prec'], nb_eval['rec'], nb_eval['f1'], nb
lr_score = [lr_eval['acc'], lr_eval['prec'], lr_eval['rec'], lr_eval['f1'], l
xgb_score = [xgb_eval['acc'], xgb_eval['prec'], xgb_eval['rec'], xgb_eval['f1
ada_score = [ada_eval['acc'], ada_eval['prec'], ada_eval['rec'], ada_eval['f1

## Set position of bar on X axis
r1 = np.arange(len(dtc_score))
r2 = [x + barWidth for x in r1]
r3 = [x + barWidth for x in r2]
r4 = [x + barWidth for x in r3]
r5 = [x + barWidth for x in r4]
r6 = [x + barWidth for x in r5]

## Make the plot
ax1.bar(r1, dtc_score, width=barWidth, edgecolor='white', label='Decision Tre
ax1.bar(r2, rf_score, width=barWidth, edgecolor='white', label='Random Forest
ax1.bar(r3, nb_score, width=barWidth, edgecolor='white', label='Naive Bayes')
ax1.bar(r4, lr_score, width=barWidth, edgecolor='white', label='LogisticRegre
ax1.bar(r5, xgb_score, width=barWidth, edgecolor='white', label='XGBoost')
ax1.bar(r6, ada_score, width=barWidth, edgecolor='white', label='XGBoost')

## Configure x and y axis
ax1.set_xlabel('Metrics', fontweight='bold')
labels = ['Accuracy', 'Precision', 'Recall', 'F1', 'Kappa']
ax1.set_xticks([r + (barWidth * 1.5) for r in range(len(dtc_score))], )
ax1.set_xticklabels(labels)
```

```
ax1.set_ylabel('Score', fontweight='bold')
ax1.set_ylim(0, 1)

## Create legend & title
ax1.set_title('Evaluation Metrics', fontsize=14, fontweight='bold')
ax1.legend()

# Second plot
## Comparing ROC Curve
ax2.plot(dtc_eval['fpr'], dtc_eval['tpr'], label='Decision Tree, auc = {:0.5f
ax2.plot(rf_eval['fpr'], rf_eval['tpr'], label='Random Forest, auc = {:0.5f}'
ax2.plot(nb_eval['fpr'], nb_eval['tpr'], label='Naive Bayes, auc = {:0.5f}'.f
ax2.plot(lr_eval['fpr'], lr_eval['tpr'], label='Logistic Regression, auc = {:
ax2.plot(xgb_eval['fpr'], xgb_eval['tpr'], label='XGBoost, auc = {:0.5f}'.for
ax2.plot(ada_eval['fpr'], ada_eval['tpr'], label='AdaBoost, auc = {:0.5f}'.fo

## Configure x and y axis
ax2.set_xlabel('False Positive Rate', fontweight='bold')
ax2.set_ylabel('True Positive Rate', fontweight='bold')

## Create legend & title
ax2.set_title('ROC Curve', fontsize=12, fontweight='bold')
ax2.legend(loc=4)

plt.show()
```

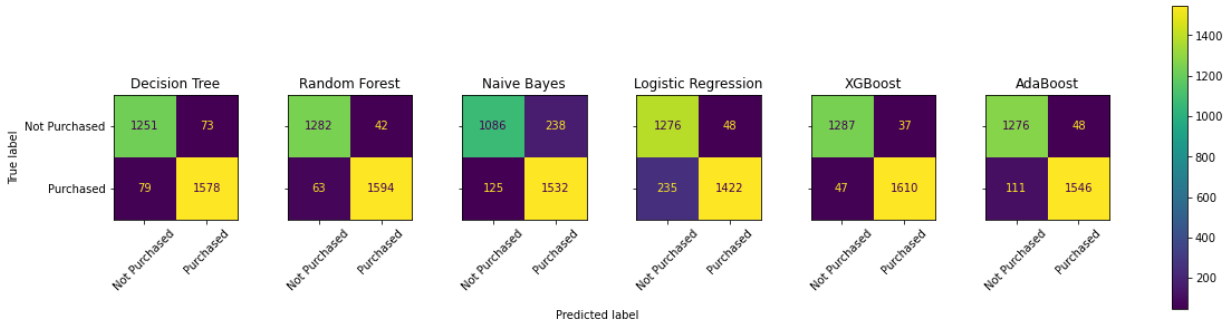


In [64]:

```
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay, accuracy
classifiers = {
    "Decision Tree": tree.DecisionTreeClassifier(random_state=0),
    "Random Forest": RandomForestClassifier(random_state=0),
    "Naive Bayes": GaussianNB(),
    "Logistic Regression": LogisticRegression(),
    "XGBoost": xgboost.XGBClassifier(),
    "AdaBoost": AdaBoostClassifier(n_estimators=100, random_state=0),
    }

f, axes = plt.subplots(1, 6, figsize=(20, 5), sharey='row')

for i, (key, classifier) in enumerate(classifiers.items()):
    y_pred = classifier.fit(X_train, y_train).predict(X_test)
```

```
    cf_matrix = confusion_matrix(y_test, y_pred)
    print(key, " \n Accuracy:",accuracy_score(y_test,y_pred),"\n F-score",f1_
    disp = ConfusionMatrixDisplay(cf_matrix,
                                  display_labels=["Not Purchased","Purchased"
    disp.plot(ax=axes[i], xticks_rotation=45)
    disp.ax_.set_title(key)
    disp.im_.colorbar.remove()
    disp.ax_.set_xlabel('')
    if i!=0:
        disp.ax_.set_ylabel('')

f.text(0.4, 0.1, 'Predicted label', ha='left')
plt.subplots_adjust(wspace=0.40, hspace=0.1)


f.colorbar(disp.im_, ax=axes)
plt.show()
```

```
Decision Tree
 Accuracy: 0.949010399194901
 F-score 0.9540507859733979
Random Forest
 Accuracy: 0.9647769204964777
 F-score 0.9681141815973276
Naive Bayes
 Accuracy: 0.8782287822878229
 F-score 0.8940764517070324
Logistic Regression
 Accuracy: 0.9050654142905066
 F-score 0.9094979213303486
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/sklearn/linear_model/_
logistic.py:814: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regress
ion
  n_iter_i = _check_optimize_result(
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:21:59] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
XGBoost
 Accuracy: 0.9718215363971822
 F-score 0.9745762711864406
AdaBoost
 Accuracy: 0.9466621938946662
 F-score 0.9510919717010151
```



hypertuning

In [65]:
```
from sklearn.model_selection import RandomizedSearchCV

classifier_smote_hpo=xgboost.XGBClassifier()
```

In [66]:
```
def timer(start_time=None):
    if not start_time:
        start_time = datetime.now()
        return start_time
    elif start_time:
        thour, temp_sec = divmod((datetime.now() - start_time).total_seconds(
        tmin, tsec = divmod(temp_sec, 60)
        print('\n Time taken: %i hours %i minutes and %s seconds.' % (thour,
```

In [82]:
```
## Hyper Parameter Optimization

params={"learning_rate"    : [0.05, 0.10, 0.15, 0.20, 0.25, 0.30 ] ,
 "max_depth"        : [ 3, 4, 5, 6, 8, 10, 12, 15],
 "min_child_weight" : [ 1, 3, 5, 7 ],
 "gamma"            : [ 0.0, 0.1, 0.2 , 0.3, 0.4 ],
 "colsample_bytree" : [ 0.3, 0.4, 0.5 , 0.7 ] }
```

In [83]:
```
random_search=RandomizedSearchCV(classifier_smote_hpo,param_distributions=par
                                 n_iter=5,scoring='roc_auc',n_jobs=-1,cv=20,v

from datetime import datetime

start_time = timer(None)
random_search.fit(X_resampled1, y_resampled1)
timer(start_time)
```

```
Fitting 20 folds for each of 5 candidates, totalling 100 fits
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
```

```
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
```

```
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
```

```
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
```

```
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
```

```
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
```

```
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
```

```
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
```

```
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
```

```
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
```

```
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
```

```
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:33:52] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.

 Time taken: 0 hours 1 minutes and 49.46 seconds.
```

In [84]:

```
random_search.best_estimator_
```

Out[84]:
```
XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
              colsample_bynode=1, colsample_bytree=0.5,
              enable_categorical=False, gamma=0.1, gpu_id=-1,
              importance_type=None, interaction_constraints='',
              learning_rate=0.25, max_delta_step=0, max_depth=15,
              min_child_weight=1, missing=nan, monotone_constraints='()',
              n_estimators=100, n_jobs=8, num_parallel_tree=1, predictor='aut
o',
              random_state=0, reg_alpha=0, reg_lambda=1, scale_pos_weight=1,
              subsample=1, tree_method='exact', validate_parameters=1,
              verbosity=None)
```

In [85]:
```
random_search.best_params_
```

Out[85]:
```
{'min_child_weight': 1,
 'max_depth': 15,
 'learning_rate': 0.25,
 'gamma': 0.1,
 'colsample_bytree': 0.5}
```

In [86]:
```
classifier=xgboost.XGBClassifier(base_score=0.5, booster='gbtree', colsample_
              colsample_bynode=1, colsample_bytree=0.5,
              enable_categorical=False, gamma=0.1, gpu_id=-1,
              importance_type=None, interaction_constraints='',
              learning_rate=0.25, max_delta_step=0, max_depth=15,
              min_child_weight=1, monotone_constraints='()',
              n_estimators=100, n_jobs=8, num_parallel_tree=1, predictor='aut
              random_state=0, reg_alpha=0, reg_lambda=1, scale_pos_weight=1,
              subsample=1, tree_method='exact', validate_parameters=1,
              verbosity=None)
```

In [94]:
```
from sklearn.model_selection import cross_val_score
score=cross_val_score(classifier,X,y,cv=10)
print(score)
```

```
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:50] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:51] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
```

```
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:52] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:53] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:54] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:54] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:55] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
```

```
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:56] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:57] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:35:57] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
[0.913085   0.91786055 0.9417383  0.90639924 0.90830946 0.9025788
 0.8739255  0.90630975 0.91108987 0.90917782]
```

In [95]:

```
classifier = classifier.fit(X_train, y_train)
```

```
/Users/ingrid/opt/anaconda3/lib/python3.9/site-packages/xgboost/sklearn.py:122
4: UserWarning: The use of label encoder in XGBClassifier is deprecated and wi
ll be removed in a future release. To remove this warning, do the following:
1) Pass option use_label_encoder=False when constructing XGBClassifier object;
and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ...,
[num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)
[15:36:03] WARNING: /Users/runner/miniforge3/conda-bld/xgboost-split_164322720
5751/work/src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluati
on metric used with the objective 'binary:logistic' was changed from 'error' t
o 'logloss'. Explicitly set eval_metric if you'd like to restore the old behav
ior.
```

In [96]:

```
y_pred_new = classifier.predict(X_test)
```

final output

In [97]:

```
result = confusion_matrix(y_test, y_pred_new)
print("Confusion Matrix:")
print(result)
result1 = classification_report(y_test, y_pred_new)
```

```
print("Classification Report:",)
print (result1)
result2 = accuracy_score(y_test, y_pred_new)
print("Accuracy:",result2)
```

```
Confusion Matrix:
[[1282   42]
 [  41 1616]]
Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.97      0.97      1324
           1       0.97      0.98      0.97      1657

    accuracy                           0.97      2981
   macro avg       0.97      0.97      0.97      2981
weighted avg       0.97      0.97      0.97      2981

Accuracy: 0.9721569942972157
```

prediction

In [99]:

```
print(X_test[0:5])
```

```
       Administrative  Administrative_Duration  Informational  \
11055               0                 3.931724              0
7328                0                 0.000000              2
7219                1                15.800000              3
11698               1                11.390306              0
11490               1                23.847481              0

       Informational_Duration  ProductRelated  ProductRelated_Duration  \
11055                0.000000               7               118.759623
7328               594.133333              39               922.400606
7219                97.800000              32              1213.650000
11698                0.000000              12               221.140419
11490                0.000000               0                 0.000000

       BounceRates  ExitRates  PageValues  SpecialDay  Month  \
11055     0.012463   0.030819    0.000000         0.0      7
7328      0.000000   0.001709  154.095539         0.0      9
7219      0.005405   0.024324    1.273317         0.0      8
11698     0.001504   0.038240    0.000000         0.0      8
11490     0.000000   0.066667    0.000000         0.0     11

       OperatingSystems  Browser  Region  TrafficType  VisitorType  Weekend
11055                 2        2       1            2            1        0
7328                  3        2       1            2            1        0
7219                  2        2       3            2            2        0
11698                 2        2       2            5            1        0
11490                 1        1       1            4            1        0
```

In [91]:

```
print(y_pred_new[0:10])
```

```
[1 1 1 1 1 0 0 0 1 1]
```

In [92]:

```
print(y_test[0:10])
```

```
11055    1
7328     1
```

```
7219     1
11698    1
11490    1
5321     0
5012     0
6530     0
10152    1
7939     1
Name: Revenue, dtype: int64
```