- Retraining an Image Classifier: Build a Keras model on top of a pre-trained image classifier to distinguish flowers.
- Text Classification: Classify IMDB movie reviews as either *positive* or *negative*.
- Style Transfer: Use deep learning to transfer style between images.
- Multilingual Universal Sentence Encoder Q&A: Use a machine learning model to answer questions from the SQuAD dataset.
- Video Interpolation: Predict what happened in a video between the first and the last frame.

DataFrame: data

View

DataFrame with shape (11055, 31)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
data = pd.read_csv("/content/Phishingwebsites_Data.csv")
data.head()
```

	id	having_IP_Address	URL_Length	Shortining_Service	having_At_Symbol	double_s
0	1	-1	1	1	1	
1	2	1	1	1	1	
2	3	1	0	1	1	
3	4	1	0	1	1	
4	5	1	0	-1	1	

```
data.drop(["id"],axis=1,inplace=True)
data.columns
```

```
Index(['having_IP_Address', 'URL_Length', 'Shortining_Service',
       'having_At_Symbol', 'double_slash_redirecting', 'Prefix_Suffix',
       'having_Sub_Domain', 'SSLfinal_State', 'Domain_registeration_length',
       'Favicon', 'port', 'HTTPS_token', 'Request_URL', 'URL_of_Anchor',
       'Links_in_tags', 'SFH', 'Submitting_to_email', 'Abnormal_URL',
       'Redirect', 'on_mouseover', 'RightClick', 'popUpWidnow', 'Iframe',
       'age_of_domain', 'DNSRecord', 'web_traffic', 'Page_Rank',
       'Google_Index', 'Links_pointing_to_page', 'Statistical_report',
       'Result'],
      dtype='object')
```

```
data.shape
```

```
(11055, 31)
```

```
data.isnull().values.any()
```

```
False
```

```
from sklearn.ensemble import AdaBoostClassifier
```

```
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('having_IP_Address',axis=1)
y=data['having_IP_Address']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

DataFrame: data
View
DataFrame with shape (11055, 31)

```
train set accuracy:  84.96155585707824
Test set accuracy :  84.71279963817278
              precision    recall  f1-score   support

          -1       0.83      0.70      0.76       762
           1       0.85      0.93      0.89      1449

    accuracy                           0.85      2211
   macro avg       0.84      0.81      0.82      2211
weighted avg       0.85      0.85      0.84      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('URL_Length',axis=1)
y=data['URL_Length']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  85.2894617819991
Test set accuracy :  84.62234283129806
              precision    recall  f1-score   support

          -1       0.87      0.95      0.91      1800
           0       0.24      0.20      0.22        30
           1       0.68      0.40      0.50       381

    accuracy                           0.85      2211
   macro avg       0.60      0.52      0.54      2211
weighted avg       0.83      0.85      0.83      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
```

```
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Shortining_Service',axis=1)
y=data['Shortining_Service']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

DataFrame: data
View
DataFrame with shape (11055, 31)

```
train set accuracy:  97.42198100407056
Test set accuracy :  97.60289461781998
              precision    recall  f1-score   support

          -1       0.95      0.86      0.90       287
           1       0.98      0.99      0.99      1924

    accuracy                           0.98      2211
   macro avg       0.96      0.93      0.94      2211
weighted avg       0.98      0.98      0.98      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('having_At_Symbol',axis=1)
y=data['having_At_Symbol']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  90.06105834464043
Test set accuracy :  89.68792401628222
              precision    recall  f1-score   support

          -1       0.77      0.47      0.59       342
           1       0.91      0.97      0.94      1869

    accuracy                           0.90      2211
   macro avg       0.84      0.72      0.76      2211
weighted avg       0.89      0.90      0.89      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
```

```
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('double_slash_redirecting',axis=1)
y=data['double_slash_redirecting']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
train set accuracy:  97.85165083672547
Test set accuracy :  97.3767526006332
              precision    recall  f1-score   support

          -1       0.91      0.88      0.89       278
           1       0.98      0.99      0.99      1933

    accuracy                           0.97      2211
   macro avg       0.95      0.93      0.94      2211
weighted avg       0.97      0.97      0.97      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Prefix_Suffix',axis=1)
y=data['Prefix_Suffix']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  87.05336951605608
Test set accuracy :  88.42152872003618
              precision    recall  f1-score   support

          -1       0.89      0.99      0.94      1930
           1       0.66      0.18      0.28       281

    accuracy                           0.88      2211
   macro avg       0.78      0.58      0.61      2211
weighted avg       0.86      0.88      0.85      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
```

```
x=data.drop('having_Sub_Domain',axis=1)
y=data['having_Sub_Domain']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  55.461329715061055
Test set accuracy :  55.992763455450024
              precision    recall  f1-score   support

          -1       0.45      0.33      0.38       650
           0       0.54      0.56      0.55       712
           1       0.63      0.73      0.68       849

    accuracy                           0.56      2211
   macro avg       0.54      0.54      0.54      2211
weighted avg       0.55      0.56      0.55      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('SSLfinal_State',axis=1)
y=data['SSLfinal_State']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  79.69244685662596
Test set accuracy :  78.01899592944369
              precision    recall  f1-score   support

          -1       0.65      0.75      0.70       710
           0       0.48      0.34      0.40       247
           1       0.91      0.88      0.90      1254

    accuracy                           0.78      2211
   macro avg       0.68      0.66      0.66      2211
weighted avg       0.78      0.78      0.78      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
```

```
x=data.drop('Domain_registeration_length',axis=1)
y=data['Domain_registeration_length']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  80.68747173224786
Test set accuracy :  80.05427408412483
              precision    recall  f1-score   support

          -1       0.86      0.82      0.84      1436
           1       0.70      0.76      0.73       775

    accuracy                           0.80      2211
   macro avg       0.78      0.79      0.79      2211
weighted avg       0.81      0.80      0.80      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Favicon',axis=1)
y=data['Favicon']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  98.30393487109905
Test set accuracy :  98.10040705563094
              precision    recall  f1-score   support

          -1       0.96      0.94      0.95       426
           1       0.99      0.99      0.99      1785

    accuracy                           0.98      2211
   macro avg       0.97      0.97      0.97      2211
weighted avg       0.98      0.98      0.98      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('port',axis=1)
y=data['port']
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  98.73360470375395
Test set accuracy :  98.28132066938036
              precision    recall  f1-score   support

          -1       0.92      0.95      0.94       295
           1       0.99      0.99      0.99      1916

    accuracy                           0.98      2211
   macro avg       0.96      0.97      0.96      2211
weighted avg       0.98      0.98      0.98      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('HTTPS_token',axis=1)
y=data['HTTPS_token']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  94.4708276797829
Test set accuracy :  93.441881501583
              precision    recall  f1-score   support

          -1       0.88      0.70      0.78       368
           1       0.94      0.98      0.96      1843

    accuracy                           0.93      2211
   macro avg       0.91      0.84      0.87      2211
weighted avg       0.93      0.93      0.93      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Request_URL',axis=1)
y=data['Request_URL']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
```

```
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  81.70511080958842
Test set accuracy :  81.63726820443237
              precision    recall  f1-score   support

          -1       0.84      0.71      0.77       950
           1       0.80      0.90      0.85      1261

    accuracy                           0.82      2211
   macro avg       0.82      0.80      0.81      2211
weighted avg       0.82      0.82      0.81      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('URL_of_Anchor',axis=1)
y=data['URL_of_Anchor']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  66.7910447761194
Test set accuracy :  67.20940750791497
              precision    recall  f1-score   support

          -1       0.75      0.84      0.80       694
           0       0.65      0.72      0.68      1043
           1       0.53      0.33      0.41       474

    accuracy                           0.67      2211
   macro avg       0.65      0.63      0.63      2211
weighted avg       0.66      0.67      0.66      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Links_in_tags',axis=1)
y=data['Links_in_tags']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
```

```
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  47.8629579375848
Test set accuracy :  46.675712347354136
              precision    recall  f1-score   support

          -1       0.49      0.47      0.48       785
           0       0.47      0.59      0.53       869
           1       0.40      0.26      0.31       557

    accuracy                           0.47      2211
   macro avg       0.45      0.44      0.44      2211
weighted avg       0.46      0.47      0.46      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('SFH',axis=1)
y=data['SFH']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  78.3355947535052
Test set accuracy :  78.24513794663048
              precision    recall  f1-score   support

          -1       0.81      0.96      0.88      1712
           0       0.00      0.00      0.00       151
           1       0.49      0.24      0.33       348

    accuracy                           0.78      2211
   macro avg       0.43      0.40      0.40      2211
weighted avg       0.71      0.78      0.73      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Submitting_to_email',axis=1)
y=data['Submitting_to_email']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
```

```
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  95.16056083220262
Test set accuracy :  93.35142469470827
              precision    recall  f1-score   support

          -1       0.93      0.70      0.80       420
           1       0.93      0.99      0.96      1791

    accuracy                           0.93      2211
   macro avg       0.93      0.85      0.88      2211
weighted avg       0.93      0.93      0.93      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Abnormal_URL',axis=1)
y=data['Abnormal_URL']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  95.88421528720036
Test set accuracy :  95.61284486657621
              precision    recall  f1-score   support

          -1       0.88      0.81      0.85       327
           1       0.97      0.98      0.97      1884

    accuracy                           0.96      2211
   macro avg       0.92      0.90      0.91      2211
weighted avg       0.96      0.96      0.96      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Redirect',axis=1)
y=data['Redirect']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
```

```
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  93.13658977838082
Test set accuracy :  92.67299864314789
              precision    recall  f1-score   support

           0       0.94      0.98      0.96      1951
           1       0.76      0.55      0.64       260

    accuracy                           0.93      2211
   macro avg       0.85      0.76      0.80      2211
weighted avg       0.92      0.93      0.92      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('on_mouseover',axis=1)
y=data['on_mouseover']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  95.16056083220262
Test set accuracy :  93.9846223428313
              precision    recall  f1-score   support

          -1       0.74      0.76      0.75       260
           1       0.97      0.96      0.97      1951

    accuracy                           0.94      2211
   macro avg       0.85      0.86      0.86      2211
weighted avg       0.94      0.94      0.94      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('RightClick',axis=1)
y=data['RightClick']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  96.40434192672998
Test set accuracy :  96.11035730438715
              precision    recall  f1-score   support

          -1       0.56      0.47      0.51        96
           1       0.98      0.98      0.98      2115

    accuracy                           0.96      2211
   macro avg       0.77      0.73      0.75      2211
weighted avg       0.96      0.96      0.96      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('popUpWidnow',axis=1)
y=data['popUpWidnow']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  98.37177747625509
Test set accuracy :  98.05517865219358
              precision    recall  f1-score   support

          -1       0.96      0.94      0.95       439
           1       0.98      0.99      0.99      1772

    accuracy                           0.98      2211
   macro avg       0.97      0.96      0.97      2211
weighted avg       0.98      0.98      0.98      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Iframe',axis=1)
y=data['Iframe']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  96.42695612844867
```

```
Test set accuracy :  96.47218453188603
              precision    recall  f1-score   support

          -1       0.81      0.78      0.80       194
           1       0.98      0.98      0.98      2017

    accuracy                           0.96      2211
   macro avg       0.89      0.88      0.89      2211
weighted avg       0.96      0.96      0.96      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('age_of_domain',axis=1)
y=data['age_of_domain']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  69.2672998643148
Test set accuracy :  68.475802804161
              precision    recall  f1-score   support

          -1       0.66      0.63      0.65      1010
           1       0.70      0.73      0.71      1201

    accuracy                           0.68      2211
   macro avg       0.68      0.68      0.68      2211
weighted avg       0.68      0.68      0.68      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('DNSRecord',axis=1)
y=data['DNSRecord']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
train set accuracy:  85.87743102668476
Test set accuracy :  86.883763003166
              precision    recall  f1-score   support
```

```
              -1       0.83      0.72      0.77       674
               1       0.88      0.93      0.91      1537

        accuracy                           0.87      2211
       macro avg       0.86      0.83      0.84      2211
    weighted avg       0.87      0.87      0.87      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('web_traffic',axis=1)
y=data['web_traffic']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
    train set accuracy:  59.55450022614202
    Test set accuracy :  60.741745816372685
                  precision    recall  f1-score   support

              -1       0.52      0.40      0.45       553
               0       0.45      0.32      0.37       501
               1       0.68      0.83      0.74      1157

        accuracy                           0.61      2211
       macro avg       0.55      0.52      0.52      2211
    weighted avg       0.58      0.61      0.59      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Page_Rank',axis=1)
y=data['Page_Rank']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
    train set accuracy:  75.62189054726367
    Test set accuracy :  75.7123473541384
                  precision    recall  f1-score   support
```

```
                -1       0.78      0.94      0.85      1636
                 1       0.58      0.25      0.34       575

          accuracy                           0.76      2211
         macro avg       0.68      0.59      0.60      2211
      weighted avg       0.73      0.76      0.72      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Google_Index',axis=1)
y=data['Google_Index']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
     train set accuracy:  86.94029850746269
     Test set accuracy :  86.83853459972863
                  precision    recall  f1-score   support

               -1       0.63      0.15      0.24       310
                1       0.88      0.99      0.93      1901

         accuracy                           0.87      2211
        macro avg       0.75      0.57      0.59      2211
     weighted avg       0.84      0.87      0.83      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Links_pointing_to_page',axis=1)
y=data['Links_pointing_to_page']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  75.02261420171868
     Test set accuracy :  74.94346449570331
                  precision    recall  f1-score   support

               -1       0.17      0.05      0.07       107
                0       0.78      0.82      0.80      1238
```

```
                1       0.73      0.73      0.73       866

         accuracy                           0.75      2211
        macro avg       0.56      0.53      0.53      2211
     weighted avg       0.73      0.75      0.74      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Statistical_report',axis=1)
y=data['Statistical_report']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

DataFrame: data

View

DataFrame with shape (11055, 31)

```
     train set accuracy:  89.80099502487562
     Test set accuracy :  89.91406603346903
                  precision    recall  f1-score   support

               -1       0.74      0.40      0.52       300
                1       0.91      0.98      0.94      1911

         accuracy                           0.90      2211
        macro avg       0.82      0.69      0.73      2211
     weighted avg       0.89      0.90      0.89      2211
```

```
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn import metrics
from sklearn.model_selection import train_test_split
x=data.drop('Result',axis=1)
y=data['Result']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
adc=AdaBoostClassifier(random_state=0)
adc.fit(x_train,y_train)
y_predict=adc.predict(x_test)
print("train set accuracy: ",100*adc.score(x_train,y_train))
print("Test set accuracy : ",100*adc.score(x_test,y_test))
print(metrics.classification_report(y_test,y_predict))
```

```
     train set accuracy:  93.8150158299412
     Test set accuracy :  93.26096788783356
                  precision    recall  f1-score   support

               -1       0.94      0.91      0.93      1014
                1       0.93      0.95      0.94      1197
```

```
              accuracy                           0.93      2211
             macro avg       0.93      0.93      0.93      2211
          weighted avg       0.93      0.93      0.93      2211
```

DataFrame: data

View

DataFrame with shape (11055, 31)

✓ 0s completed at 10:29 AM