Practical 8

Aim: To implement Adaboost learning algorithm.

Code:

*import* pandas *as* pd

*from* sklearn.preprocessing *import* StandardScaler

*from* sklearn.preprocessing *import* LabelEncoder

*from* sklearn.feature\_extraction *import* DictVectorizer

*from* sklearn.ensemble *import* AdaBoostClassifier

data=pd.read\_csv('data.csv')

cols\_to\_retain=['Alt','Bar','Fri','Hun','Pat','Price','Rain','Res','Type','Est']

X\_feature=data[cols\_to\_retain]

X\_dict=X\_feature.T.to\_dict().values()

vect=DictVectorizer(sparse=False)

X\_vector=vect.fit\_transform(X\_dict)

print(X\_vector)

X\_Train=X\_vector[:-1]

X\_Test=X\_vector[-1:]

print('Train set')

print(X\_Train)

print('Test set')

print(X\_Test)

le=LabelEncoder()

y\_Train=le.fit\_transform(data['Goal'][:-1])

scaler=StandardScaler()

scaler.fit(X\_Test)

X\_Train=scaler.transform(X\_Train)

X\_Test=scaler.transform(X\_Test)

model=AdaBoostClassifier(random\_state=1)

model.fit(X\_Train, y\_Train)

print(le.inverse\_transform(model.predict(X\_Test)))

Output:

