```
##Experiment 7:Ensemble Learning
##Level 1: Implement Ensemble Learning algorithms
such as Bagging,Pasting and Out-of Bag Evaluation.
##Level 2: Random Patches and Random Subspace Method.
```

```
  File "<ipython-input-2-33a5945471e0>", line 3
    such as Bagging,Pasting and Out-of Bag Evaluation.
         ^
SyntaxError: invalid syntax
```

SEARCH STACK OVERFLOW

```
!pip install matplotlib-venn
```

```
Requirement already satisfied: matplotlib-venn in /usr/local/lib/python3.10/dist-packages (0.11.9)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-packages (from matplotlib-venn) (3.7.1)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from matplotlib-venn) (1.23.5)
Requirement already satisfied: scipy in /usr/local/lib/python3.10/dist-packages (from matplotlib-venn) (1.11.3)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib->matplotlib-venn) (1.2.0)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib->matplotlib-venn) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib->matplotlib-venn) (4.44.3)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib->matplotlib-venn) (1.4.5)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib->matplotlib-venn) (23.2)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib->matplotlib-venn) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib->matplotlib-venn) (3.1.1)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib->matplotlib-venn) (2.8.2
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib->matplotlib-ve
```

```
!apt-get -qq install -y libfluidsynth1
```

```
E: Package 'libfluidsynth1' has no installation candidate
```

```
!pip install cartopy
import cartopy
```

```
Requirement already satisfied: cartopy in /usr/local/lib/python3.10/dist-packages (0.22.0)
Requirement already satisfied: numpy>=1.21 in /usr/local/lib/python3.10/dist-packages (from cartopy) (1.23.5)
Requirement already satisfied: matplotlib>=3.4 in /usr/local/lib/python3.10/dist-packages (from cartopy) (3.7.1)
Requirement already satisfied: shapely>=1.7 in /usr/local/lib/python3.10/dist-packages (from cartopy) (2.0.2)
Requirement already satisfied: packaging>=20 in /usr/local/lib/python3.10/dist-packages (from cartopy) (23.2)
Requirement already satisfied: pyshp>=2.1 in /usr/local/lib/python3.10/dist-packages (from cartopy) (2.3.1)
Requirement already satisfied: pyproj>=3.1.0 in /usr/local/lib/python3.10/dist-packages (from cartopy) (3.6.1)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.4->cartopy) (1.2.0)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.4->cartopy) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.4->cartopy) (4.44.3)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.4->cartopy) (1.4.5)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.4->cartopy) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.4->cartopy) (3.1.1)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib>=3.4->cartopy) (2.8.2)
Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-packages (from pyproj>=3.1.0->cartopy) (2023.7.22)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib>=3.4->cartopy)
```

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import BaggingClassifier
from sklearn.metrics import classification_report
import matplotlib.pyplot as plt
import cartopy
```

```
# Load winequality dataset
df = pd.read_csv("https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-red.csv", delimiter=";")
```

```
# Split data into training and testing sets
X = df.drop(columns=['quality'])
y = df['quality']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,random_state=42)
```

```
# Train decision tree model
dt = DecisionTreeClassifier(random_state=42)
dt.fit(X_train, y_train)
```

```
▾       DecisionTreeClassifier
DecisionTreeClassifier(random_state=42)
```

```
# Print classification report for decision tree model
y_pred = dt.predict(X_test)
print("Decision Tree")
print(classification_report(y_test, y_pred))
```

```
Decision Tree
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.00      0.00      0.00        10
           5       0.63      0.68      0.65       130
           6       0.55      0.53      0.54       132
           7       0.51      0.50      0.51        42
           8       0.00      0.00      0.00         5

    accuracy                           0.56       320
   macro avg       0.28      0.28      0.28       320
weighted avg       0.55      0.56      0.55       320
```

```
# Train Bagging classifier
bagging = BaggingClassifier(base_estimator=DecisionTreeClassifier(),
                            n_estimators=100, random_state=42)
bagging.fit(X_train, y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_base.py:166: FutureWarning: `b
  warnings.warn(
▸          BaggingClassifier
▸ base_estimator: DecisionTreeClassifier
        ▸ DecisionTreeClassifier
```

```
# Print classification report for Bagging classifier
y_pred = bagging.predict(X_test)
print("Bagging")
print(classification_report(y_test, y_pred))
```

```
Bagging
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.00      0.00      0.00        10
           5       0.72      0.78      0.75       130
           6       0.63      0.68      0.66       132
           7       0.59      0.48      0.53        42
           8       0.00      0.00      0.00         5

    accuracy                           0.66       320
   macro avg       0.32      0.32      0.32       320
weighted avg       0.63      0.66      0.64       320

/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetricWarning: Precision and F-score are ill-d
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetricWarning: Precision and F-score are ill-d
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetricWarning: Precision and F-score are ill-d
  _warn_prf(average, modifier, msg_start, len(result))
```

```
# Train Pasting classifier
pasting = BaggingClassifier(base_estimator=DecisionTreeClassifier(),
                            n_estimators=100, bootstrap=False, random_state=42)
pasting.fit(X_train, y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_base.py:166: FutureWarning: `b
  warnings.warn(
```

▸ BaggingClassifier

▸ base_estimator: DecisionTreeClassifier

▸ DecisionTreeClassifier

```
# Print classification report for Pasting classifier
y_pred = pasting.predict(X_test)
print("Pasting")
print(classification_report(y_test, y_pred))
```

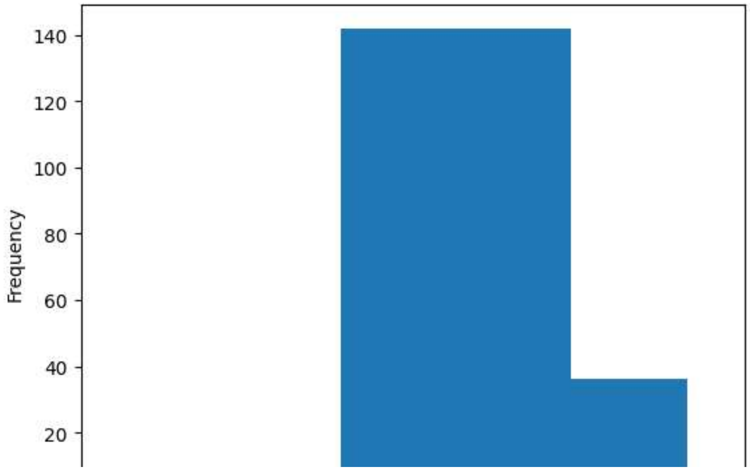
```
Pasting
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.00      0.00      0.00        10
           5       0.64      0.66      0.65       130
           6       0.56      0.54      0.55       132
           7       0.53      0.55      0.54        42
           8       0.00      0.00      0.00         5

    accuracy                           0.56       320
   macro avg       0.29      0.29      0.29       320
weighted avg       0.56      0.56      0.56       320
```

```
# Evaluate Out-of-Bag score for Bagging classifier
bagging_oob = BaggingClassifier(base_estimator=DecisionTreeClassifier(),
                                n_estimators=100, oob_score=True,
                                random_state=42)
bagging_oob.fit(X_train, y_train)
oob_score = bagging_oob.oob_score_
print("Out-of-Bag Score")
print("OOB Score:", oob_score)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_base.py:166: FutureWarning: `base_estimator` was renamed to `estimator` in ver
  warnings.warn(
Out-of-Bag Score
OOB Score: 0.6942924159499609
```

```
# Plot confusion matrix for Bagging classifier
(bagging, X_test, y_test)
plt.show()
```

```
# Plot histogram of predicted quality values for Bagging classifier
y_pred = bagging.predict(X_test)
plt.hist(y_pred, bins=range(3, 9), align='left')
plt.xticks(range(3, 9))
plt.xlabel("Predicted Quality")
plt.ylabel("Frequency")
plt.show()
```


140
120
100
80
60
40
20
Frequency