# we are tuning three hyperparameters right now, we are passing the different values for both parameters

grid\_param = {

"n\_estimators" : [90,100,115,130],

'criterion': ['gini', 'entropy'],

'max\_depth' : range(2,20,1),

'min\_samples\_leaf' : range(1,10,1),

'min\_samples\_split': range(2,10,1),

'max\_features' : ['auto','log2']

}

from sklearn.model\_selection import GridSearchCV

grid\_searh=GridSearchCV(estimator=model,param\_grid=grid\_param,cv=3,verbose=1)

grid\_searh.fit(X\_train,y\_train)

—------------------------------------------------------------------------------------------------------------------

from sklearn.svm import SVC

from sklearn.ensemble import BaggingClassifier

from sklearn.datasets import make\_classification

model\_bagging\_svc = BaggingClassifier(base\_estimator=SVC(),n\_estimators=50, random\_state=0).fit(X, y)

y\_predict\_bagging=model\_bagging\_svc.predict(X\_test)

accuracy\_score(y\_test,y\_predict\_bagging)

Rf\_model\_with\_best\_params=RandomForestClassifier(criterion='gini',max\_depth= 14,max\_features= 'log2',min\_samples\_leaf= 1,min\_samples\_split= 2,n\_estimators=115)

—--------------------------------------------------------------------------------------------------

**###use bagging classifier and regressor, extra tree classifier and regressor, voting classifier and regressor and random forest classifier and regressor on top of Household consumption data and census data**

**data link:** <https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption>

<https://archive.ics.uci.edu/ml/datasets/census+income>

**Sklearn\_link:**

<https://scikit-learn.org/stable/modules/classes.html#module-sklearn.ensemble>

Submission form:

<https://forms.gle/qnVhjQ9bkwFT4vax5>