**Problem Statement**

In this assignment students will build the random forest model after normalizing the variable to house pricing from boston data set. Following the code to get data into the environment:

**#importing libraries**

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.model\_selection

import train\_test\_split

from sklearn.preprocessing

import StandardScaler from sklearn

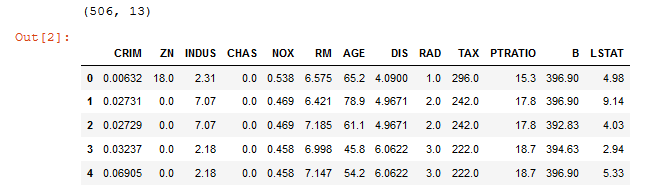
import datasets boston = datasets.load\_boston()

features = pd.DataFrame(boston.data, columns=boston.feature\_names)

targets = boston.target

print(features.shape)

features.head()



**# Now we have to standardize the data. We will use StandardScaler from preprocessing**

X\_train, X\_test, y\_train, y\_test = train\_test\_split(features, targets, train\_size=0.8, random\_state=43)

from sklearn import preprocessing

min\_max\_scaler = preprocessing.MinMaxScaler()

features\_scaled = min\_max\_scaler.fit\_transform(features)

features\_scaled = pd.DataFrame(features\_scaled)

targets = targets.reshape(506,1)

**# We can even check if we can reduce dimensionality of dataset using PCA, although here its only 13 features.**

%matplotlib inline

from sklearn.decomposition import PCA

pca=PCA()

pca.fit(X\_train)

foo=pd.DataFrame(pca.transform(X\_train))

x\_axis=np.arange(1,pca.n\_components\_ + 1)

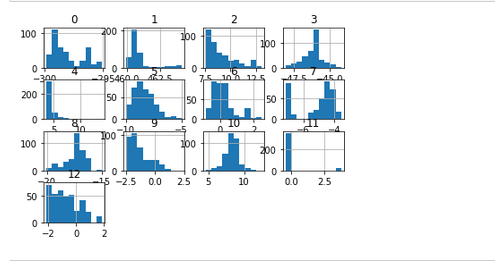
pca\_scaled=PCA()

pca\_scaled.fit(X\_train\_scaled)

foo\_scaled=pd.DataFrame(pca.transform(X\_train\_scaled))

foo\_scaled.hist()

plt.show()



**# Import ,Instantiate,Fit**

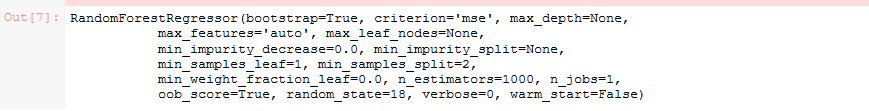
from sklearn.ensemble import RandomForestRegressor

forest=RandomForestRegressor(n\_estimators=1000,oob\_score=True,random\_state=18)

**# Oob\_True :Whether To Use Out-Of-Bag Samples To Estimate**

**# The R^2 On Unseen Data.Is Kind Of Cross-Validation**

forest.fit(X\_train,y\_train)



from sklearn.metrics import r2\_score

preds=forest.predict(X\_test)

test\_score=r2\_score(y\_test,preds)

print(' R squared score:',test\_score)

op4.PNG