Pattern & Anomaly Detection Lab

Experiment 7

Submitted By:

Dhruv Singhal

500075346

R177219074

AIML B3

Submitted To:

Dr. Gopal Phartiyal

Asst. Professor

SOCS

UPES

CODE:

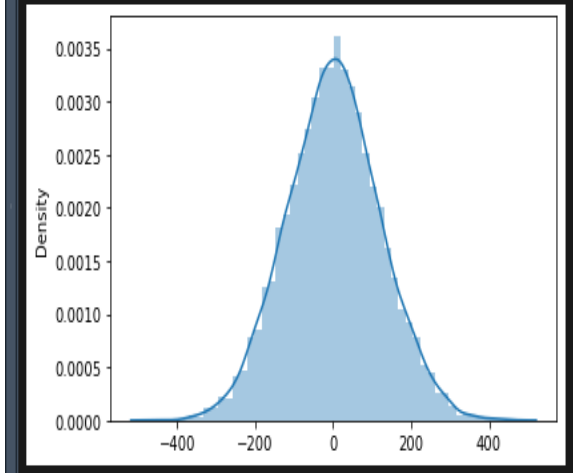
```
# -*- coding: utf-8 -*-
"""
Created on Mon Sep 20 15:33:41 2021

@author: Dhruv Singhal
"""

#%%
import numpy as np
import matplotlib.pyplot  as plt
#import pandas as pd
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
#%%
from sklearn.datasets import make_regression
x,y=make_regression(n_samples=10000,n_features=5,noise=30)
sns.distplot(y)

#%%
plt.hist(x)
#%%
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.15,random_state=42)
from sklearn.metrics import r2_score,mean_squared_error
#%%
from sklearn.linear_model import LinearRegression
lin=LinearRegression()
lin.fit(x_train,y_train)
y_pred=lin.predict(x_test)
print("r2 score without tuning",r2_score(y_test,y_pred))
print("RMSE without tuning",np.sqrt(mean_squared_error(y_test,y_pred)))

#%%  Hyperparameter Tuning From Here
#%%
from sklearn.model_selection import GridSearchCV
tuned_parameters = [{'fit_intercept': ['True'], 'normalize': ['True']},
                    {'fit_intercept': ['False'], 'normalize': ['True']},
                    {'fit_intercept': ['True'], 'normalize': ['False']},
                    {'fit_intercept': ['False'], 'normalize': ['False']}

                    ]
clf=GridSearchCV(LinearRegression(),tuned_parameters,scoring=('r2'))
clf.fit(x_train,y_train)
```

```
print("Best parameters set found on development set:")
print()
print(clf.best_params_)
print()
print("Best Score:",clf.best_score_)
z=clf.cv_results_

#%% kfold cross validation with hyperparameter tuning
from sklearn.model_selection import KFold
k = 5
kf = KFold(n_splits=k, random_state=None)
model = GridSearchCV(LinearRegression(),tuned_parameters,scoring=('r2'))

for train_index , test_index in kf.split(x):
    X_train , X_test = x[train_index,:],x[test_index,:]
    y_train , y_test = y[train_index] , y[test_index]

    model.fit(X_train,y_train)
    pred_values = model.predict(X_test)
print("Best parameters set found on development set:")
print()
print(model.best_params_)
print()
print("Best Score:",model.best_score_)
z2=model.cv_results_
```

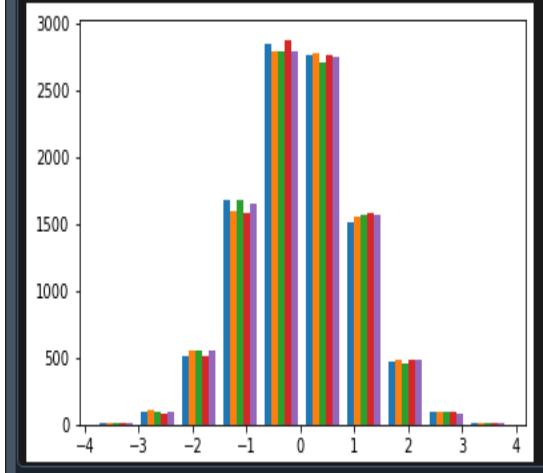
OUTPUT:

Console 8/A

```
Python 3.7.6 (default, Jan  8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.26.0 -- An enhanced Interactive Python.

In [1]: runcell(0, 'B:/3rd year/5th sem/P&AD/hyperparametertuning_linear_regression.py')

In [2]: runcell(1, 'B:/3rd year/5th sem/P&AD/hyperparametertuning_linear_regression.py')

In [3]: runcell(2, 'B:/3rd year/5th sem/P&AD/hyperparametertuning_linear_regression.py')
```




```
In [4]: runcell(3, 'B:/3rd year/5th sem/P&AD/hyperparametertuning_linear_regression.py')
```



IPython console History

```
In [5]: runcell(4, 'B:/3rd year/5th sem/P&AD/hyperparametertuning_linear_regression.py')

In [6]: runcell(5, 'B:/3rd year/5th sem/P&AD/hyperparametertuning_linear_regression.py')
r2 score without tuning 0.9378429885435289
RMSE without tuning 30.094973564046324

In [7]: runcell('Hyperparameter Tuning From Here', 'B:/3rd year/5th sem/P&AD/hyperparametertuning_linear_regression.py')
Nothing to execute, this cell is empty.


In [8]: runcell(7, 'B:/3rd year/5th sem/P&AD/hyperparametertuning_linear_regression.py')
Best parameters set found on development set:

{'fit_intercept': 'True', 'normalize': 'True'}

Best Score: 0.9354610495395299

In [9]: runcell('kfold cross validation with hyperparameter tuning', 'B:/3rd year/5th sem/P&AD/hyperparametertuning_linear_regression.py')
Best parameters set found on development set:

{'fit_intercept': 'True', 'normalize': 'True'}

Best Score: 0.9380426092738423

In [10]:
```