PREDICTING HOUSE PRICES USING MACHINE LEARNING

INTRODUCTION

GIVEN DATA SET

https://www.kaggle.com/datasets/vedavyasv/usa-housing

CODING

import pandas as pd

import numpy as np

Import matplotlib as mpl

Import matplotlib. Pyplot as plt

```
%matplotlib inline

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

data=pd.read_csv(‘data.csv’)

data=pd.read_csv(‘usa housing.csv’)

 data.dropna()
```

▪ Remove rows

```
 data.fillna(value)
```

▪ Replace missing value with our specific value

```
 data.drop_duplicate()
```

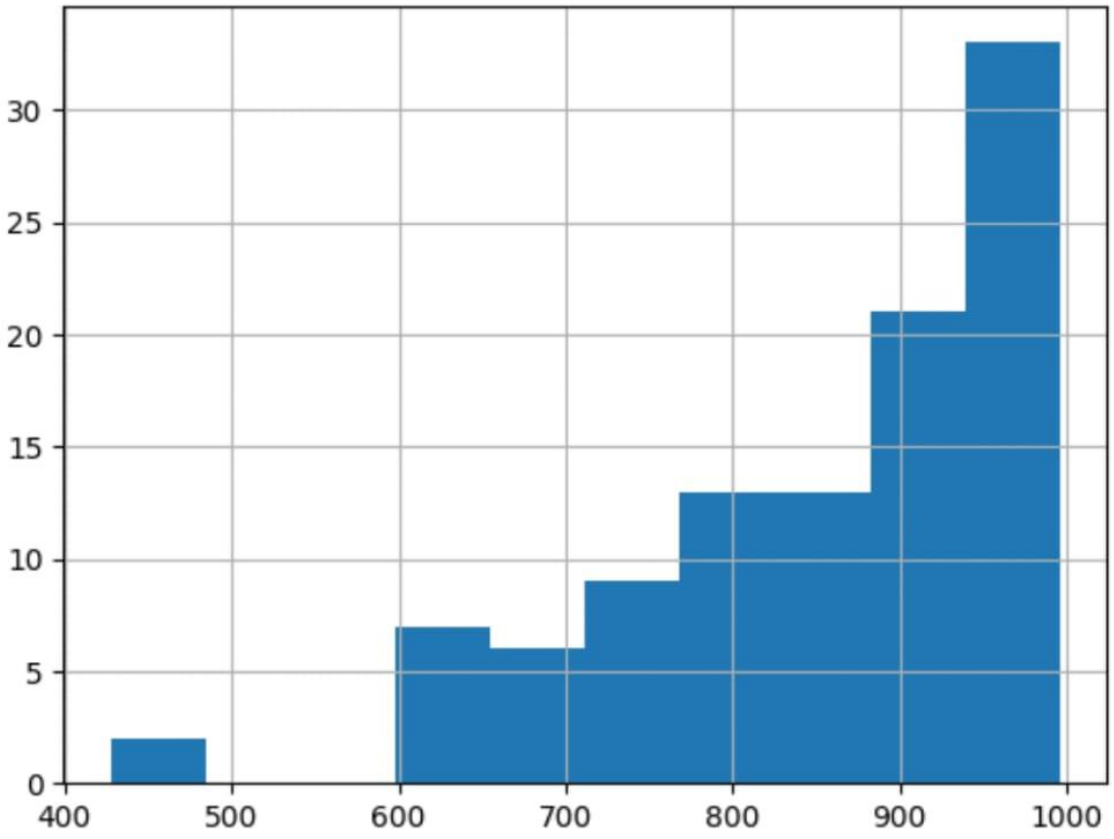
▪ Remove duplicate data

```
data[‘column_name’].replace(old_value, new_value, inplace=True)
```

▪ Replace the value of old value which is no more needed,to a value with newly update

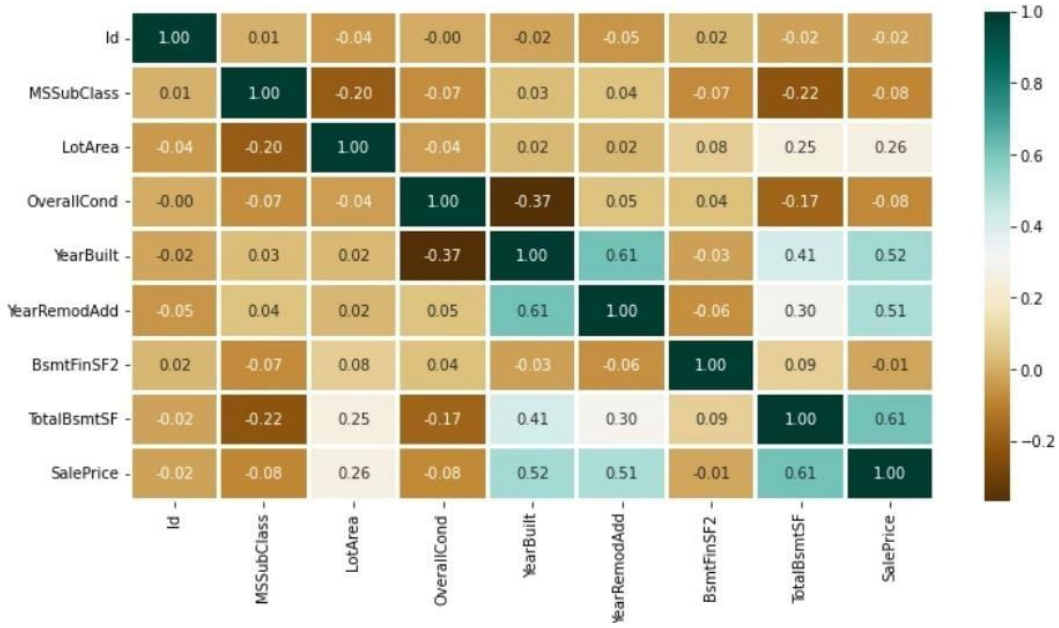
```
pd.merge()

df = pd.merge(df1, df2, on = ‘common attribute ‘)

df..head(10)

#Sort by name

sorted = df.sort_values(by=['name'])

display(sorted)

#Filter rows

just_students = df.query('is_student==True')

display(just_students)
```

```
#Filter columns
no_birthday = df.filter(['name','is_student','target'])
display(no_birthday)
#Rename column
renamed = df.rename(columns={'target':'target_score'})
display(renamed)
#Splitting
splitnames = df.copy()
split = splitnames['name'].str.split(' ', expand = True)
splitnames['first'] = split[0]
splitnames['last'] = split[1]
display(splitnames)
#Data Value transforms
df['target'].hist()
```

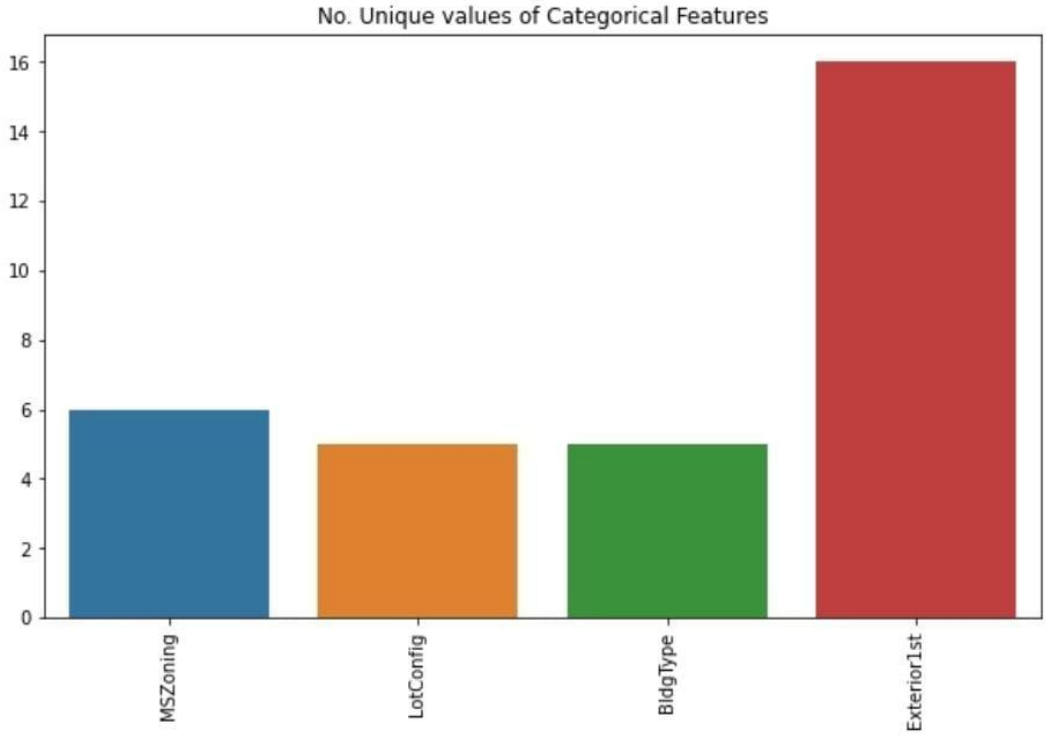



```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import r2_score,
mean_absolute_error,mean_squared_error
from sklearn.linear_model import LinearRegression
from sklearn.linear_model import Lasso
```

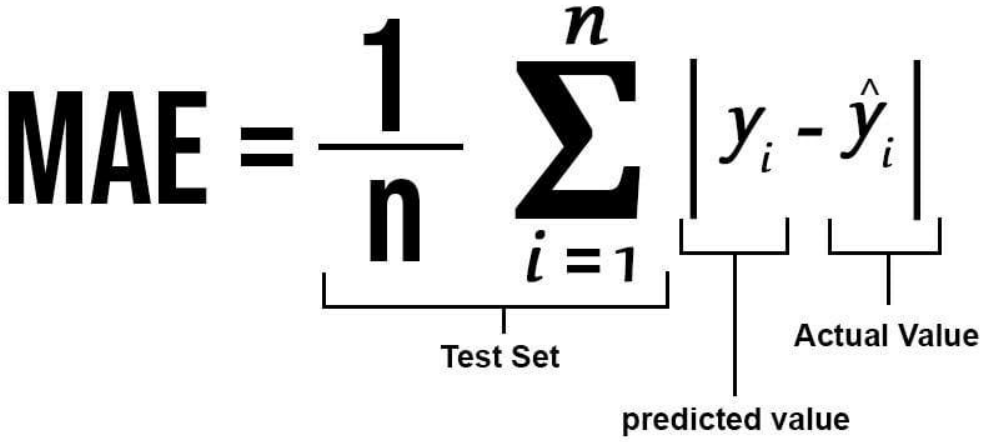
```
from sklearn.ensemble import RandomForestRegressor
from sklearn.svm import SVR
import xgboost as xg
Warnings.filterwarnings(“ignore”)
Sns.histplot(dataset, x=’Price’, bins=50, color=’y’)
Out:
<Axes:xlabel=’Price’, ylabel=’Count’>
Plt.figure(figsize=(12, 6))
Sns.heatmap(dataset.corr(), Cmap = ‘BrBG’ Fmt = ‘.2f’
Linewidths = 2, Annot = True)
```




```
unique_values = []
for col in object_cols:
unique_values.append(dataset[col].unique().size)
plt.figure(figsize=(10,6))
plt.title('No. Unique values of Categorical Features')
```


No. Unique values of Categorical Features
16
14
12
10
8
6
4
2
0
MSZoning
LotConfig
BldgType
Exterior1st

Mean absolute percentage Error:



```
from sklearn.linear_model import LinearRegression

model_LR = LinearRegression()

model_LR.fit(X_train, Y_train)

y_pred = model_LR.predict(X_valid)

print(mean_absolute_percentage_error(Y_valid, Y_pred))
```