houseprice-advanced

July 2, 2025

1 House Prices – Advanced Regression Dataset

2 Import Libraries

[1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error
```

3 Load Dataset

[18]:
```
df = pd.read_csv('train.csv')
df = pd.read_csv('test.csv')
```

[19]:
```
print("Train Shape: ",df.shape)
print("Test Shape: ",df.shape)
df.head()
```

```
Train Shape:  (1459, 80)
Test Shape:  (1459, 80)
```

[19]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	\
0	1461	20	RH	80.0	11622	Pave	NaN	Reg	
1	1462	20	RL	81.0	14267	Pave	NaN	IR1	
2	1463	60	RL	74.0	13830	Pave	NaN	IR1	
3	1464	60	RL	78.0	9978	Pave	NaN	IR1	
4	1465	120	RL	43.0	5005	Pave	NaN	IR1	

	LandContour	Utilities	…	ScreenPorch	PoolArea	PoolQC	Fence	MiscFeature	\
0	Lvl	AllPub	…	120	0	NaN	MnPrv	NaN	
1	Lvl	AllPub	…	0	0	NaN	NaN	Gar2	

```
2         Lvl    AllPub  …           0        0    NaN  MnPrv         NaN
3         Lvl    AllPub  …           0        0    NaN    NaN         NaN
4         HLS    AllPub  …         144        0    NaN    NaN         NaN

  MiscVal MoSold  YrSold  SaleType  SaleCondition
0       0      6    2010        WD         Normal
1   12500      6    2010        WD         Normal
2       0      3    2010        WD         Normal
3       0      6    2010        WD         Normal
4       0      1    2010        WD         Normal

[5 rows x 80 columns]
```

4 Check Missing Values

[20]:
```
missing = df.isnull().sum()
missing = missing[missing > 0].sort_values(ascending=False)
print(missing)
```

```
PoolQC          1456
MiscFeature     1408
Alley           1352
Fence           1169
MasVnrType       894
FireplaceQu      730
LotFrontage      227
GarageCond        78
GarageYrBlt       78
GarageQual        78
GarageFinish      78
GarageType        76
BsmtCond          45
BsmtExposure      44
BsmtQual          44
BsmtFinType1      42
BsmtFinType2      42
MasVnrArea        15
MSZoning           4
BsmtFullBath       2
BsmtHalfBath       2
Functional         2
Utilities          2
GarageCars         1
GarageArea         1
TotalBsmtSF        1
KitchenQual        1
```

```
BsmtUnfSF          1
BsmtFinSF2         1
BsmtFinSF1         1
Exterior2nd        1
Exterior1st        1
SaleType           1
dtype: int64
```

5 Check for NULL values

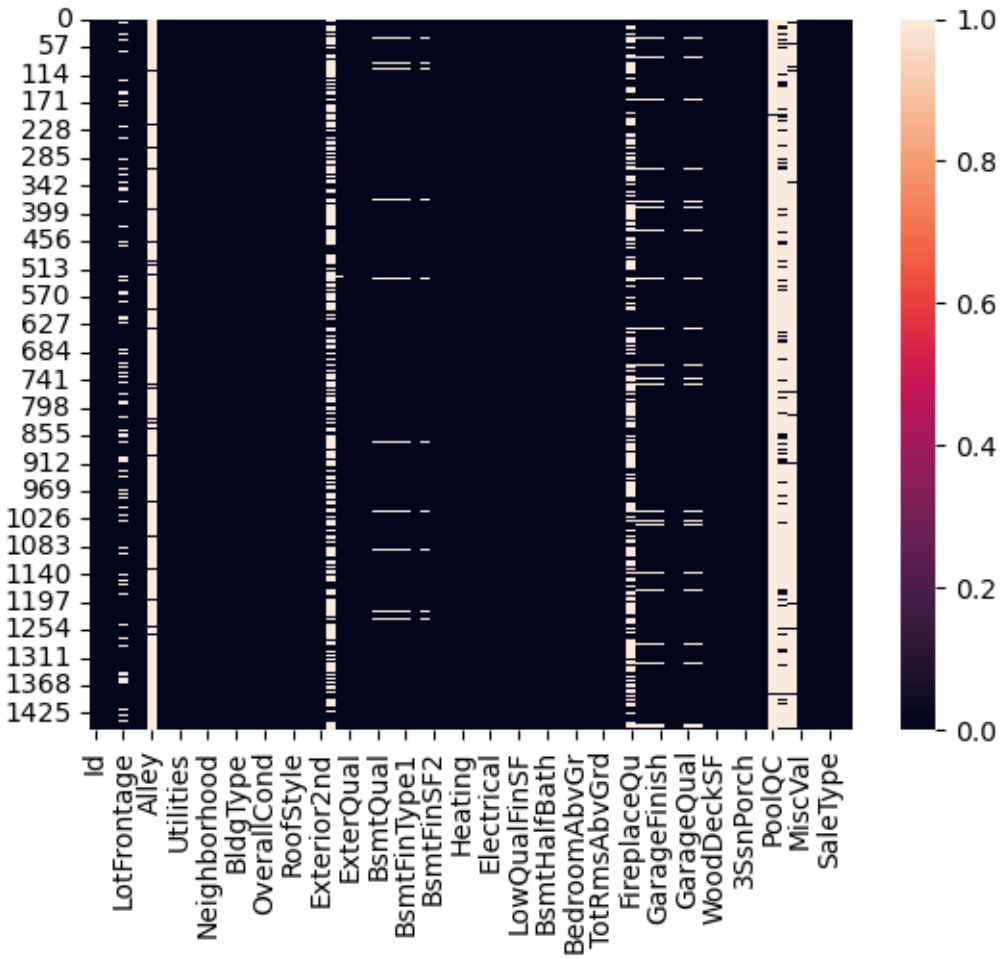
6 Train Data

[7]:
```
print(train.isnull().sum())
```

```
Id                 0
MSSubClass         0
MSZoning           0
LotFrontage      259
LotArea            0
                  …
MoSold             0
YrSold             0
SaleType           0
SaleCondition      0
SalePrice          0
Length: 81, dtype: int64
```

[8]:
```
sns.heatmap(train.isnull())
```

[8]:
```
<Axes: >
```



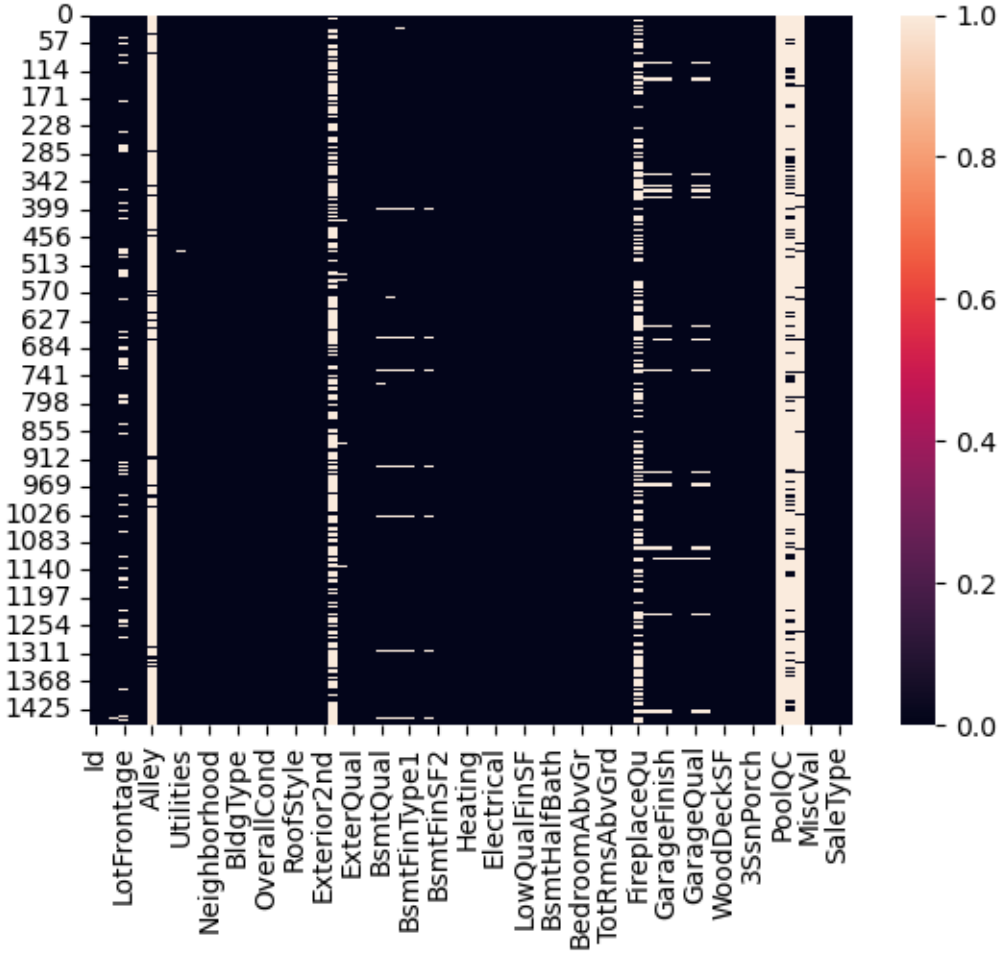
7 Test Data

[9]:
```
print(test.isnull().sum())
```

```
Id                0
MSSubClass        0
MSZoning          4
LotFrontage     227
LotArea           0
                 ..
MiscVal           0
MoSold            0
YrSold            0
SaleType          1
SaleCondition     0
Length: 80, dtype: int64
```

[10]:
```
sns.heatmap(test.isnull())
```

[10]: <Axes: >



8 Handling NULL data

9 For train data

[11]:
```
cat_col_train =␣
 ↪['FireplaceQu','GarageType','GarageFinish','MasVnrType','BsmtQual',
          ␣
 ↪'BsmtCond','BsmtExposure','BsmtFinType1','BsmtFinType2','FireplaceQu',
          'GarageQual','GarageCond']

ncat_col_train = ['LotFrontage','GarageYrBlt','MasVnrArea']
```

[12]:
```
for i in cat_col_train:
    train[i] = train[i].fillna(train[i].mode()[0])

for j in ncat_col_train:
    train[j] = train[j].fillna(train[j].mean())
```

10 For test data

[13]:
```
cat_col_test =␣
 ↪['FireplaceQu','GarageType','GarageFinish','MasVnrType','BsmtQual',
          ␣
 ↪'BsmtCond','BsmtExposure','BsmtFinType1','BsmtFinType2','FireplaceQu',
         ␣
 ↪'GarageQual','GarageCond','MSZoning','Utilities','Exterior1st','Exterior2nd','KitchenQual',

ncat_col_test =␣
 ↪['LotFrontage','GarageYrBlt','MasVnrArea','BsmtFinSF1','BsmtFinSF2','BsmtUnfSF','TotalBsmtS
                'BsmtHalfBath','GarageCars','GarageArea']
```

[14]:
```
for i in cat_col_test:
    test[i] = test[i].fillna(test[i].mode()[0])

for j in ncat_col_test:
    test[j] = test[j].fillna(test[j].mean())
```

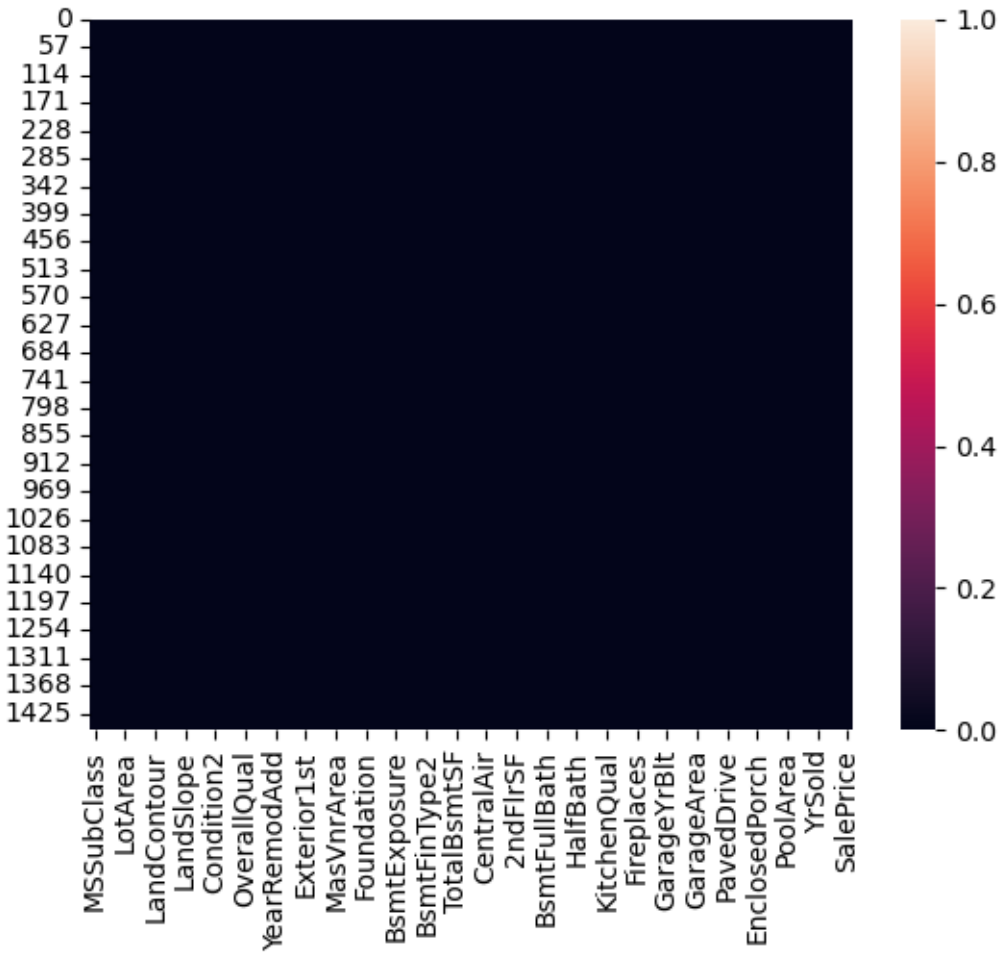
11 Drop Columns

[15]:
```
to_drop = ['Id','Alley','PoolQC','Fence','MiscFeature']

for k in to_drop:
    train.drop([k], axis = 1, inplace = True)
    test.drop([k], axis = 1, inplace = True)
```

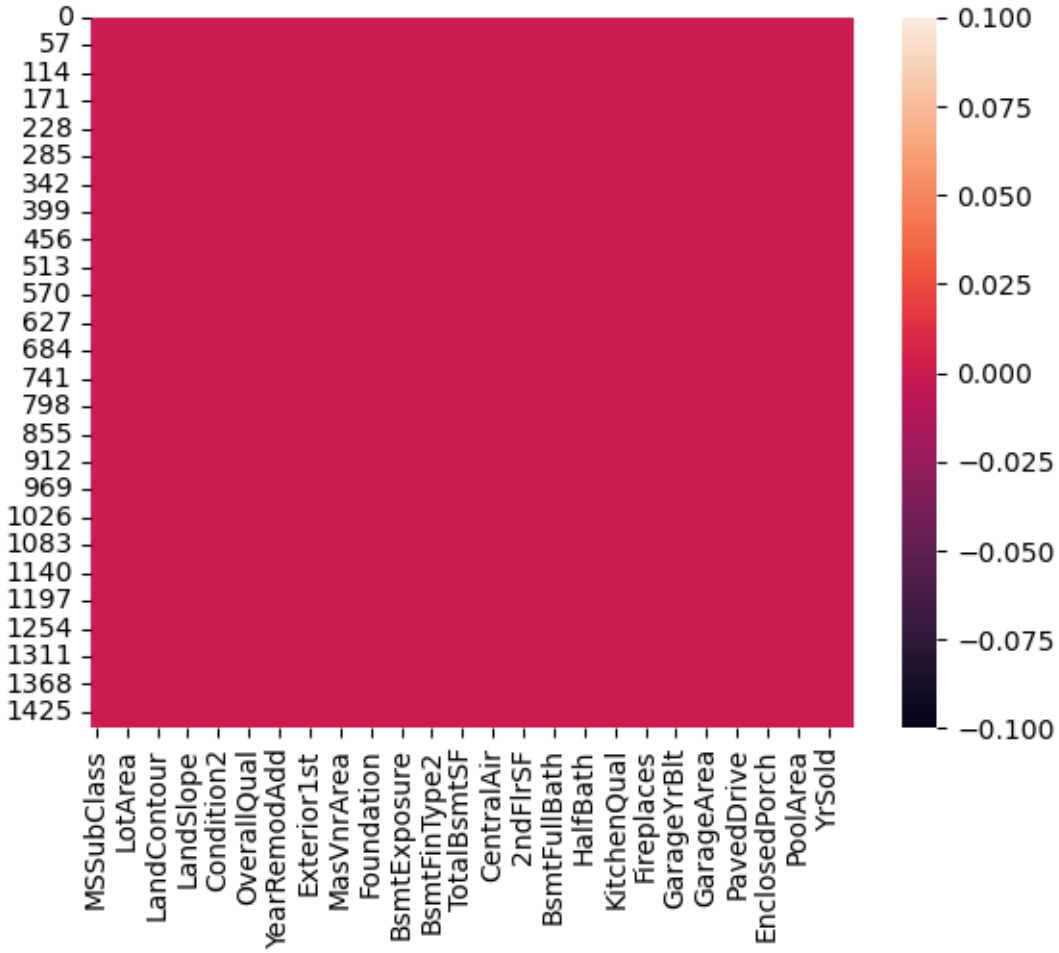
[16]:
```
sns.heatmap(train.isnull())
```

[16]: <Axes: >



[17]:
```
sns.heatmap(test.isnull())
```

[17]: <Axes: >


0
57
114
171
228
285
342
399
456
513
570
627
684
741
798
855
912
969
1026
1083
1140
1197
1254
1311
1368
1425
MSSubClass
LotArea
LandContour
LandSlope
Condition2
OverallQual
YearRemodAdd
Exterior1st
MasVnrArea
Foundation
BsmtExposure
BsmtFinType2
TotalBsmtSF
CentralAir
2ndFlrSF
BsmtFullBath
HalfBath
KitchenQual
Fireplaces
GarageYrBlt
GarageArea
PavedDrive
EnclosedPorch
PoolArea
YrSold
0.100
0.075
0.050
0.025
0.000
−0.025
−0.050
−0.075
−0.100

[ ]: