L1 L2 regularization for overfiting

May 11, 2023

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

[10]:
```
df=pd.read_csv(r'C:\Users\Rakesh\Downloads\melb_data.csv')
df.head(3)
```

[10]:
```
       Suburb          Address  Rooms Type    Price Method SellerG  \
0  Abbotsford     85 Turner St      2    h  1480000      S  Biggin
1  Abbotsford  25 Bloomburg St      2    h  1035000      S  Biggin
2  Abbotsford     5 Charles St      3    h  1465000     SP  Biggin

         Date  Distance  Postcode  …  Bathroom  Car  Landsize  BuildingArea  \
0  03-12-2016       2.5      3067  …         1  1.0       202           NaN
1  04-02-2016       2.5      3067  …         1  0.0       156          79.0
2  04-03-2017       2.5      3067  …         2  0.0       134         150.0

   YearBuilt  CouncilArea Lattitude  Longtitude             Regionname  \
0        NaN        Yarra  -37.7996    144.9984  Northern Metropolitan
1     1900.0        Yarra  -37.8079    144.9934  Northern Metropolitan
2     1900.0        Yarra  -37.8093    144.9944  Northern Metropolitan

  Propertycount
0          4019
1          4019
2          4019

[3 rows x 21 columns]
```

[14]:
```
df.isnull().sum().count
```

[14]:
```
<bound method Series.count of Suburb             0
Address             0
Rooms               0
```

```
Type              0
Price             0
Method            0
SellerG           0
Date              0
Distance          0
Postcode          0
Bedroom2          0
Bathroom          0
Car              62
Landsize          0
BuildingArea   6450
YearBuilt      5375
CouncilArea    1369
Lattitude         0
Longtitude        0
Regionname        0
Propertycount     0
dtype: int64>
```

[15]:
```
df.nunique()
```

[15]:
```
Suburb           314
Address        13378
Rooms              9
Type               3
Price           2204
Method             5
SellerG          268
Date              58
Distance         202
Postcode         198
Bedroom2          12
Bathroom           9
Car               11
Landsize        1448
BuildingArea     602
YearBuilt        144
CouncilArea       33
Lattitude       6503
Longtitude      7063
Regionname         8
Propertycount    311
dtype: int64
```

[16]:
```
df.shape
```

[16]:
```
(13580, 21)
```

[27]:
```
col_useful=['Suburb','Rooms','Type','Method','SellerG','Distance','Bedroom2','Bathroom','Car'
␣
↪,'Landsize','CouncilArea','BuildingArea','Regionname','Propertycount',
          'Price']
```

[28]:
```
df.columns
```

[28]:
```
Index(['Suburb', 'Address', 'Rooms', 'Type', 'Price', 'Method', 'SellerG',
       'Date', 'Distance', 'Postcode', 'Bedroom2', 'Bathroom', 'Car',
       'Landsize', 'BuildingArea', 'YearBuilt', 'CouncilArea', 'Lattitude',
       'Longtitude', 'Regionname', 'Propertycount'],
      dtype='object')
```

[29]:
```
df1=df[col_useful]
df1.head(2)
```

[29]:

	Suburb	Rooms	Type	Method	SellerG	Distance	Bedroom2	Bathroom	Car	Landsize	CouncilArea	BuildingArea	Regionname	Propertycount	Price
0	Abbotsford	2	h	S	Biggin	2.5	2	1	1.0	202	Yarra	NaN	Northern Metropolitan	4019	1480000
1	Abbotsford	2	h	S	Biggin	2.5	2	1	0.0	156	Yarra	79.0	Northern Metropolitan	4019	1035000

[30]:
```
df1.shape
```

[30]:
```
(13580, 15)
```

[32]:
```
df1.isna().sum()
```

[32]:
```
Suburb             0
Rooms              0
Type               0
Method             0
SellerG            0
Distance           0
Bedroom2           0
Bathroom           0
Car               62
Landsize           0
CouncilArea     1369
BuildingArea    6450
```

```
Regionname          0
Propertycount       0
Price               0
dtype: int64
```

[33]:
```
df1.isnull().sum()
```

[33]:
```
Suburb              0
Rooms               0
Type                0
Method              0
SellerG             0
Distance            0
Bedroom2            0
Bathroom            0
Car                62
Landsize            0
CouncilArea      1369
BuildingArea     6450
Regionname          0
Propertycount       0
Price               0
dtype: int64
```

[34]:
```
col_fillna=['Car']
df1[col_fillna]=df1[col_fillna].fillna(0)
```

[35]:
```
df1.isnull().sum()
```

[35]:
```
Suburb              0
Rooms               0
Type                0
Method              0
SellerG             0
Distance            0
Bedroom2            0
Bathroom            0
Car                 0
Landsize            0
CouncilArea      1369
BuildingArea     6450
Regionname          0
Propertycount       0
Price               0
dtype: int64
```

[37]:
```
df1['BuildingArea']=df1['BuildingArea'].fillna(df1.BuildingArea.mean())
```

[38]:
```
df1.isnull().sum()
```

[38]:
```
Suburb             0
Rooms              0
Type               0
Method             0
SellerG            0
Distance           0
Bedroom2           0
Bathroom           0
Car                0
Landsize           0
CouncilArea     1369
BuildingArea       0
Regionname         0
Propertycount      0
Price              0
dtype: int64
```

[39]:
```
df1.shape
```

[39]:
```
(13580, 15)
```

[40]:
```
df1.dropna(inplace=True)
```

[41]:
```
df1.isnull().sum()
```

[41]:
```
Suburb          0
Rooms           0
Type            0
Method          0
SellerG         0
Distance        0
Bedroom2        0
Bathroom        0
Car             0
Landsize        0
CouncilArea     0
BuildingArea    0
Regionname      0
Propertycount   0
Price           0
dtype: int64
```

[42]:
```
df1.shape
```

[42]:
```
(12211, 15)
```

[43]:
```
df1=pd.get_dummies(df1,drop_first=True)
df1.head(3)
```

[43]:

	Rooms	Distance	Bedroom2	Bathroom	Car	Landsize	BuildingArea	\
0	2	2.5	2	1	1.0	202	151.96765	
1	2	2.5	2	1	0.0	156	79.00000	
2	3	2.5	3	2	0.0	134	150.00000	

	Propertycount	Price	Suburb_Aberfeldie	…	CouncilArea_Wyndham	\
0	4019	1480000	0	…	0	
1	4019	1035000	0	…	0	
2	4019	1465000	0	…	0	

	CouncilArea_Yarra	CouncilArea_Yarra Ranges	Regionname_Eastern Victoria	\
0	1	0	0	
1	1	0	0	
2	1	0	0	

	Regionname_Northern Metropolitan	Regionname_Northern Victoria	\
0	1	0	
1	1	0	
2	1	0	

	Regionname_South-Eastern Metropolitan	Regionname_Southern Metropolitan	\
0	0	0	
1	0	0	
2	0	0	

	Regionname_Western Metropolitan	Regionname_Western Victoria
0	0	0
1	0	0
2	0	0

[3 rows x 613 columns]

[45]:
```
x=df1.drop('Price',axis=1)
y=df1.Price
```

[46]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y, test_size=0.
  ↪3,random_state=2)
x_train.shape
```

[46]: (8547, 612)

[48]:
```
from sklearn.linear_model import LinearRegression
reg=LinearRegression()
```

```
reg.fit(x_train,y_train)
```

[48]:
```
LinearRegression()
```

[49]:
```
reg.score(x_test,y_test)
```

[49]:
```
-3765962962688.7295
```

[50]:
```
reg.score(x_train,y_train)
```

[50]:
```
0.7120927409656957
```

[51]:
```
from sklearn import linear_model

lasso_L1 = linear_model.Lasso(alpha=50,max_iter=100,tol=0.1)

lasso_L1.fit(x_train,y_train)
```

[51]:
```
Lasso(alpha=50, max_iter=100, tol=0.1)
```

[52]:
```
lasso_L1.score(x_test,y_test)
```

[52]:
```
0.6635361138790516
```

[53]:
```
lasso_L1.score(x_train,y_train)
```

[53]:
```
0.7074303619393785
```

[ ]:
```

```