superMarket_regression

April 24, 2018

In [272]:
```
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression,Ridge,Lasso
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
```

In [273]:
```
df = pd.read_csv('train.csv')
df.columns
```

Out[273]:
```
Index([u'Item_Identifier', u'Item_Weight', u'Item_Fat_Content',
       u'Item_Visibility', u'Item_Type', u'Item_MRP', u'Outlet_Identifier',
       u'Outlet_Establishment_Year', u'Outlet_Size', u'Outlet_Location_Type',
       u'Outlet_Type', u'Item_Outlet_Sales'],
      dtype='object')
```

In [274]:
```
df.head()
```

Out[274]:

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Location_Type	Outlet_Type	Item_Outlet_Sales
0	FDA15	9.30	Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium	Tier 1	Supermarket Type1	3735.1380
1	DRC01	5.92	Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium	Tier 3	Supermarket Type2	443.4228
2	FDN15	17.50	Low Fat	0.016760	Meat	141.6180	OUT049	1999	Medium	Tier 1	Supermarket Type1	2097.2700
3	FDX07	19.20	Regular	0.000000	Fruits and Vegetables	182.0950	OUT010	1998	NaN	Tier 3	Grocery Store	732.3800
4	NCD19	8.93	Low Fat	0.000000	Household	53.8614	OUT013	1987	High	Tier 3	Supermarket Type1	994.7052

```
          Outlet_Type  Item_Outlet_Sales
0  Supermarket Type1          3735.1380
1  Supermarket Type2           443.4228
2  Supermarket Type1          2097.2700
3      Grocery Store           732.3800
4  Supermarket Type1           994.7052
```

In [275]:
```
# preprocessing fillna
df['Item_Identifier'] = df['Item_Identifier'].fillna(df['Item_Identifier'].max())
df['Item_Weight'] = df['Item_Weight'].fillna(df['Item_Weight'].mean())
df['Item_Fat_Content']=df['Item_Fat_Content'].fillna(df['Item_Fat_Content'].max())
df['Item_Visibility']=df['Item_Visibility'].fillna(df['Item_Visibility'].mean())
df['Item_Type']=df['Item_Type'].fillna(df['Item_Type'].max())
df['Item_MRP']=df['Item_MRP'].fillna(df['Item_MRP'].mean())
df['Outlet_Identifier']=df['Outlet_Identifier'].fillna(df['Outlet_Identifier'].max())
df['Outlet_Establishment_Year']=df['Outlet_Establishment_Year'].fillna(df['Outlet_Esta
df['Outlet_Size']=df['Outlet_Size'].fillna(df['Outlet_Size'].max())
df['Outlet_Location_Type']=df['Outlet_Location_Type'].fillna(df['Outlet_Location_Type'
df['Outlet_Type']=df['Outlet_Type'].fillna(df['Outlet_Type'].max())
df['Item_Outlet_Sales']=df['Item_Outlet_Sales'].fillna(df['Item_Outlet_Sales'].mean())
```

In [276]:
```
# replace and format
df['Item_Fat_Content'] = df['Item_Fat_Content'].replace('low fat', 'Low Fat')
df['Item_Fat_Content'] = df['Item_Fat_Content'].replace('LF', 'Low Fat')
df['Item_Fat_Content'] = df['Item_Fat_Content'].replace('reg', 'Regular')
```

In [277]:
```
strData = [df['Item_Identifier'],df['Item_Fat_Content'],df['Item_Type'], df['Outlet_Id
f = pd.DataFrame(strData)
f = f.T
```

In [278]:
```
df['Item_Identifier'] = pd.get_dummies(f['Item_Identifier'])
df['Item_Type'] = pd.get_dummies(f['Item_Type'])
df['Item_Fat_Content']= pd.get_dummies(f['Item_Fat_Content'])
df['Outlet_Identifier']= pd.get_dummies(f['Outlet_Identifier'])
df['Outlet_Size']= pd.get_dummies(f['Outlet_Size'])
df['Outlet_Location_Type']= pd.get_dummies(f['Outlet_Location_Type'])
df['Outlet_Type']= pd.get_dummies(f['Outlet_Type'])
```

In [279]:
```
df['Outlet_Establishment_Year'] = 2018 - df['Outlet_Establishment_Year']
df['Outlet_Establishment_Year'].head()
```

Out[279]:
```
0    19
1     9
2    19
3    20
4    31
Name: Outlet_Establishment_Year, dtype: int64
```

In [280]:
```
x = df
x = x.drop('Item_Outlet_Sales',axis=1)
y = df['Item_Outlet_Sales']
```

In [281]:
```
train_x,test_x,train_y,test_y = train_test_split(x,y)
```

In [282]:
```
train_x.head()
```

Out[282]:

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	\
6758	0	18.600000	1	0.152295	
4441	0	12.300000	0	0.064619	
1172	0	11.100000	1	0.033160	
8340	0	7.390000	1	0.120468	
3039	0	12.857645	1	0.053148	

	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	\
6758	0	98.9358	0	19	
4441	0	89.8804	0	21	
1172	0	119.6124	0	19	
8340	0	145.1470	0	19	
3039	0	36.3874	0	33	

	Outlet_Size	Outlet_Location_Type	Outlet_Type
6758	0	1	0
4441	0	1	0
1172	0	1	0
8340	0	1	0
3039	0	1	1

In [289]:
```
model =Ridge()
# model.fit(train_x['Item_MRP'].values.reshape(-1,1),train_y)
model.fit(train_x,train_y)
```

Out[289]:
```
Ridge(alpha=1.0, copy_X=True, fit_intercept=True, max_iter=None,
   normalize=False, random_state=None, solver='auto', tol=0.001)
```

In [291]:
```
predicted = model.predict(test_x)
```

In [292]:
```
model.coef_
```

Out[292]:
```
array([ 1.47193588e+02, -1.24447234e+00, -3.81002388e+01, -1.82130437e+02,
        6.97566415e+00,  1.55859286e+01,  3.31220397e+02,  6.51624071e+01,
       -1.19010128e+03, -4.48651007e+02, -2.74120892e+03])
```

In [293]:
```
model.score(test_x,test_y)
```

Out[293]:
```
0.5488582331709895
```

In [287]:
```
mean_squared_error(predicted,test_y)
```

Out[287]:
```
1312019.5174309174
```

In [288]:
```
error = 0
mse = np.mean((predicted - test_y)**2)
mse
```

Out[288]:
```
1312019.5174309174
```