superMarket_2_Regression

April 25, 2018

In [162]:
```
import pandas as pd
import numpy as np
from sklearn import linear_model
from sklearn import metrics
from sklearn.model_selection import cross_validate, cross_val_predict
```

In [88]:
```
df1 = pd.read_csv('train.csv')
df2 = pd.read_csv('test.csv')
```

In [98]:
```
# df = df1
# df = df2
```

In [99]:
```
# handle null/missing data
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_Estab
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 [100]:
```
# handle unformated/ data
df['Item_Fat_Content'] = df['Item_Fat_Content'].replace('reg', 'Regular')
df['Item_Fat_Content'] = df['Item_Fat_Content'].replace('LF', 'Low Fat')
df['Item_Fat_Content'] = df['Item_Fat_Content'].replace('low fat', 'Low Fat')
```

In [101]:
```
# df1 = df
# df2 = df
# del df
```

In [102]:
```
all_data = pd.concat((df1,df2))
for column in all_data.select_dtypes(include=[np.object]).columns:
    df1[column] = df1[column].astype('category', categories = all_data[column].unique(
    df2[column] = df2[column].astype('category', categories = all_data[column].unique(
```

In [109]:
```
# df1 = df1.drop(["value"], axis=1)
df1 = pd.get_dummies(df1)
df2 = pd.get_dummies(df2)
```

In [110]:
```
df1.head()
```

Out[110]:

	Item_Weight	Item_Visibility	Item_MRP	Outlet_Establishment_Year	\
0	9.30	0.016047	249.8092	1999	
1	5.92	0.019278	48.2692	2009	
2	17.50	0.016760	141.6180	1999	
3	19.20	0.000000	182.0950	1998	
4	8.93	0.000000	53.8614	1987	

	Item_Outlet_Sales	Item_Identifier_FDA15	Item_Identifier_DRC01	\
0	3735.1380	1	0	
1	443.4228	0	1	
2	2097.2700	0	0	
3	732.3800	0	0	
4	994.7052	0	0	

	Item_Identifier_FDN15	Item_Identifier_FDX07	Item_Identifier_NCD19	\
0	0	0	0	
1	0	0	0	
2	1	0	0	
3	0	1	0	
4	0	0	1	

	...	Outlet_Size_Medium	Outlet_Size_Small	\
0	...	1	0	
1	...	1	0	
2	...	1	0	
3	...	0	1	
4	...	0	0	

	Outlet_Size_High	Outlet_Location_Type_Tier 1	Outlet_Location_Type_Tier 3	\
0	0	1	0	
1	0	0	1	
2	0	1	0	
3	0	0	1	
4	1	0	1	

	Outlet_Location_Type_Tier 2	Outlet_Type_Supermarket Type1	\
0	0	1	
1	0	0	
2	0	1	
3	0	0	
4	0	1	

```
   Outlet_Type_Supermarket Type2  Outlet_Type_Grocery Store  \
0                              0                          0
1                              1                          0
2                              0                          0
3                              0                          1
4                              0                          0

   Outlet_Type_Supermarket Type3
0                              0
1                              0
2                              0
3                              0
4                              0

[5 rows x 1602 columns]
```

In [111]:
```
df2.head()
```

Out[111]:
```
   Item_Weight  Item_Visibility  Item_MRP  Outlet_Establishment_Year  \
0    20.750000         0.007565  107.8622                       1999
1     8.300000         0.038428   87.3198                       2007
2    14.600000         0.099575  241.7538                       1998
3     7.315000         0.015388  155.0340                       2007
4    12.695633         0.118599  234.2300                       1985

   Item_Identifier_FDA15  Item_Identifier_DRC01  Item_Identifier_FDN15  \
0                      0                      0                      0
1                      0                      0                      0
2                      0                      0                      0
3                      0                      0                      0
4                      0                      0                      0

   Item_Identifier_FDX07  Item_Identifier_NCD19  Item_Identifier_FDP36  \
0                      0                      0                      0
1                      0                      0                      0
2                      0                      0                      0
3                      0                      0                      0
4                      0                      0                      0

               ...                Outlet_Size_Medium  Outlet_Size_Small  \
0              ...                                 1                  0
1              ...                                 0                  1
2              ...                                 0                  1
3              ...                                 0                  1
4              ...                                 1                  0

   Outlet_Size_High  Outlet_Location_Type_Tier 1  Outlet_Location_Type_Tier 3  \
0                 0                            1                            0
```

```
1                 0                                0                                0
2                 0                                0                                1
3                 0                                0                                0
4                 0                                0                                1

   Outlet_Location_Type_Tier 2  Outlet_Type_Supermarket Type1  \
0                            0                             1
1                            1                             1
2                            0                             0
3                            1                             1
4                            0                             0

   Outlet_Type_Supermarket Type2  Outlet_Type_Grocery Store  \
0                              0                           0
1                              0                           0
2                              0                           1
3                              0                           0
4                              0                           0

   Outlet_Type_Supermarket Type3
0                              0
1                              0
2                              0
3                              0
4                              1

[5 rows x 1601 columns]
```

```
In [113]: x_train = df1
          x_train = x_train.drop('Item_Outlet_Sales',axis=1)
          y_train = df1['Item_Outlet_Sales']
          x_test = df2

In [119]: len(x_train) == len(x_test)
          # len(x_test)

Out[119]: False

In [75]: y_train.head()

Out[75]: 0    3735.1380
         1     443.4228
         2    2097.2700
         3     732.3800
         4     994.7052
         Name: Item_Outlet_Sales, dtype: float64

In [136]: model = linear_model.ElasticNet()
          model.fit(x_train,y_train)
```

```
Out[136]: ElasticNet(alpha=1.0, copy_X=True, fit_intercept=True, l1_ratio=0.5,
                     max_iter=1000, normalize=False, positive=False, precompute=False,
                     random_state=None, selection='cyclic', tol=0.0001, warm_start=False)

In [139]: predicted = model.predict(x_test)
          model.score(x_train,y_train)

Out[139]: 0.46635591769119483

In [147]: model.intercept_

Out[147]: 23001.071446376955

In [148]: # LR 0.643
          # Ridge 0.640
          # Lasso 0.564
          # ElasticNet 0.466

In [173]: scores = cross_validate(model, x_train, y_train, cv=5, return_train_score=False)
          predicted = cross_val_predict(model,x_train,y_train,cv=5)
          scores

Out[173]: {'fit_time': array([0.57317805, 0.58499718, 0.61211205, 0.56701994, 0.54454398]),
           'score_time': array([0.01032996, 0.00862288, 0.01089811, 0.01187897, 0.01045299]),
           'test_score': array([0.47685585, 0.46340778, 0.45871856, 0.46907352, 0.45482636])}
```