SPRINGBOARD - DATA SCIENCE CAREER TRACK

IN-DEPTH ANALYSIS (MACHINE LEARNING)

HOUSING PRICES-ADVANCED REGRESSION TECHNIQUES

INTRODUCTION

In this section we will apply the Machine Learning Models on the train set, after all the pre-processing steps and then predict the Sale Price for the test set.

PREPROCESSING AND FEATURE ENGINEERING STEPS:

We would clean the data from the train set and create new variables as follows.

1. Creating a list of categories and numeric columns.
2. Creation of new variable 'newsqavr'.
3. Creating new variables out of numeric variables that reflect absence or presence of zero.
4. Filling NaN and zero with interpolation
5. Categorical Nan's filled with 'Absent'
6. All categorical variables have been converted into numeric values that represent mean Sale Price of each label.

LASSO REGULARISATION FOR FEATURE SELECTION.

Now we have a train set which is cleaned and transformed. We need to select a few relevant features. This helps in reducing cost. For this we will apply Lasso Regularisation. With alpha of .001 we were able to extract the following features.

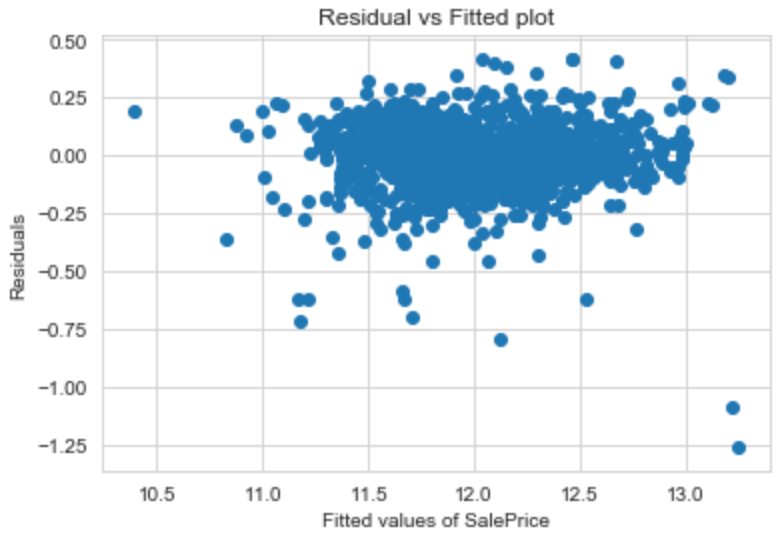
	Features	Coefficients
1	newsqvar	0.38
2	OverallQual	0.29
3	Neighborhood	0.18
4	HalfBath	0.12
5	CentralAir	0.11
6	Functional	0.11
7	YearRemodAdd	0.11
8	BsmtExposure	0.09
9	MSZoning	0.08
10	TotRmsAbvGrd	0.08
11	KitchenQual	0.08
12	GarageCars	0.08
13	Condition1	0.06
14	LotArea	0.06
15	FireplaceQu	0.06
16	GarageQual	0.05
17	MSSubClass	0.05
18	HeatingQC	0.05
19	FullBath	0.05
20	Fireplaces	0.05
21	GarageCond	0.04
22	SaleCondition	0.04
23	YearBuilt	0.02
24	BsmtQual	0.02
25	WoodDeckSF	0.01
26	Exterior1st	0.01
27	GarageYrBlt	0.01
28	MasVnrArea	0.01
29	2ndFlrSF	0.01
30	GarageFinish	0.01
31	EnclosedPorch	0.01
32	BsmtFinSF1	0.00
33	BedroomAbvGr	0.00
34	zeroLowQualFinSF	0.00
35	PoolQC	0.00
36	zeroScreenPorch	0.00
37	Street	0.00

Not all the features above are statistically significant, so I selected features whose p-values were less or equal to 5%. This list came out as follows:

['newsqvar', 'OverallQual', 'Neighborhood', 'HalfBath',
'CentralAir', 'Functional', 'YearRemodAdd', 'BsmtExposure',
'MSZoning', 'TotRmsAbvGrd', 'KitchenQual', 'GarageCars',
'Condition1', 'LotArea', 'HeatingQC', 'FullBath',
'SaleCondition', 'WoodDeckSF', 'BedroomAbvGr', 'BsmtUnfSF']

RESIDUAL PLOT FOR IDENTIFYING OUTLIERS

Linear regression was fitted and we calculated the residuals. We then plot predictions from the model with the residuals. This shows us the outliers. Outliers have been identified as the residual points above .25 and below -.25. We then delete these points from the train data. We use the same definition of outliers for the test set as well.



TRAIN MACHINE LEARNING MODELS

LINEAR REGRESSION MODEL- Train set

We create a Linear Regression Object and fit to train data. The score for evaluating performance is Root Mean Square Error between log of Sale Price and log of Predictions. We will use 5 fold cross validation on the train set. The score is as follows:
The cross validated RMSE of Linear Regression Model- lm1, on train set is: 0.096

RIDGE REGRESSION MODEL- Train set

We create a Ridge Regression Object and fit to train data. The score for evaluating performance is Root Mean Square Error between log of Sale Price and log of Predictions.
First we will use Grid Search cross validation on the train set in order to tune alpha.

The findings from the Grid Search CV is as follows:

Optimum value of alpha for Ridge Regression Model is: {'alpha': 50}
The cross validated RMSE for Ridge Regression Model, using optimum value for alpha is : 0.112

We create a Ridge Regression object using optimum value for alpha and fit on the train set. The scores are as follows:
The RMSE of Ridge Model- ridg, on train set is: 0.112

RANDOM FOREST MODEL- Train Set

Finding the optimum hyperparameters:
We use Randomised Grid Search CV for finding optimum parameters. The parameters selected are:

param_grid={'max_features':[5,10,15,25], 'max_samples':[200, 300, 500, 700, 1000, 1168],'n_estimators':[100, 500, 1000,1500],'min_samples_split':[2,6,8,10,12,15]}

Optimum values of hyperparameters for RandomForest model are: {'n_estimators': 500, 'min_samples_split': 6, 'max_samples': 300, 'max_features': 5}

We then Create and fit Random Forest with optimum hyperparameters and arrive at the following score:

The cross validated RMSE for Ridge Regression Model, is : 0.14

We extract the important features using the _feature_importances method and convert it into a dataframe. We then list the top 20 features as follows :

	Feature_name	Feature_Importance
1	newsqvar	0.08
2	OverallQual	0.06
3	Neighborhood	0.06
4	YearBuilt	0.05
5	ExterQual	0.04
6	BsmtQual	0.04
7	GarageYrBlt	0.04
8	KitchenQual	0.04
9	YearRemodAdd	0.03
10	FullBath	0.03
11	FireplaceQu	0.03
12	GarageCars	0.03
13	GarageArea	0.03
14	TotRmsAbvGrd	0.02
15	GarageType	0.02
16	BsmtFinSF1	0.02
17	GarageFinish	0.02
18	Foundation	0.02
19	Fireplaces	0.02
20	MSSubClass	0.02

PREDICT TARGET VARIABLE FOR TEST SET

PREPROCESSING AND FEATURE ENGINEERING STEPS- TEST DATA:

1. Reading Test data.
2. Creating New variable'newsqvar' and amending the data frame.
3. Creating new variables out of numeric variables that reflect absence or presence of zero.
4. Nan Imputation
5. Perform numeric labelling of categorical variables for test data.
   For this we applied the test_category_conversion(col) function to all categorical columns.

```
def test_category_conversion(col):
    a=test.groupby(col)['SalePrice'].mean()
    a=round((a/10000),2)
    index=a.index.values.tolist()
    weight=a.values.tolist()
    zipped=list(zip(index,weight))
    dict1=dict(zipped)
    test[col]=test[col].map(dict1)
```

6. Identifying and deleting outliers for Linear Regression Model.

CREATING X (FEATURES) AND Y (TARGET VARIABLE) FOR TEST SET.

We transform the Test data to its logarithmic equivalent using the log transformation object created.

APPLY LINEAR REGRESSION MODEL- (LM1) ON TEST SET AND PREDICT.

The RMSE i.e root mean squared error between log of actual Sale Price and Predicted Sale Price is: 0.12

APPLY RANDOM FOREST MODEL- (RF) ON TEST SET AND PREDICT¶

The RMSE i.e root mean squared error between log of actual Sale Price and Predicted Sale Price is: 0.086

Reversing log transformation of predicted Sale Price:

We can reverse the log transformation object and obtain the Sale Price values.