SPRINGBOARD - DATA SCIENCE CAREER TRACK

PROJECT REPORT ON HOUSING PRICES-ADVANCED REGRESSION TECHNIQUES

TABLE OF CONTENTS

IN- DEPTH ANALYSIS (MACHINE LEARNING)

PROJECT PROPOSAL

PROBLEM

Prediction of sale prices.

Let's say that a buyer is interested in purchasing a house. He has an estimate of the price of the house and has an offer in mind. The price estimation might have been based on few factors or external sources such as real estate agencies. The problem for the buyer is knowing the exact amount for the purchase price of the house. For a real estate company, which can also pose as a buyer or broker, the problem is to negotiate for the best deal. This dataset has several factors. It becomes crucial to know the levers that drive the price and develop a model to predict them with best accuracy.

CLIENT

Model for price prediction of a house can be a valuable tool for buyer/seller of a house, real estate agent/company, builders or tax departments.

USES OF THE PRICE PREDICTION MODEL

1. The client will be able to predict the sale price of a house.
2. Various aspects or features that have a strong influence on price can be known.
3. The client can be in an advantageous position while negotiating.
4. The model can be useful to real estate agents and online companies as it would save additional costs and time in further examination and research.
5. Having an idea of the most influential features would enable the client to plan and effect changes in the property vis a vis the cost and expected return from investment. One can also decide what features need to be included for the house construction / renovation as per budget.

DELIVERABLES

The analysis will be done using a python program in Jupyter notebook. A final presentation for the same will be prepared to highlight the findings.

DATA ACQUISITION

The Ames Housing dataset part of Kaggle competitions is used. The dataset has 79 explanatory variables describing every aspect of residential homes in Ames, Iowa.

PROCESS

1. Data cleaning to handle Nan values and outliers.
2. Data observation to understand the scope challenges, estimators data types and central tendencies.
3. Data visualisation to look at the distribution and other aspects of features and target variable.
4. Exploratory Data Analysis to understand the correlations of features among itself and with the target variable.
5. Fit the Linear Model and predict prices.

METRIC

Model will be evaluated on Root mean squared error(RMSE) between logarithm of predicted value and logarithm of sale price.

DATA WRANGLING

In this section we will clean the data, classify the variables, deal with missing values and outliers.

Step1- Classifying dtypes

There are 1460 observations. There are 81 variables of int,float and object dtypes.
We classify data types into 2 classes i.e numeric and categorical. We also need to identify ordinal variables and decide which class we can place them in. Following lists are created:
NUMERIC ('numcollist')
CATEGORICAL('categlist')

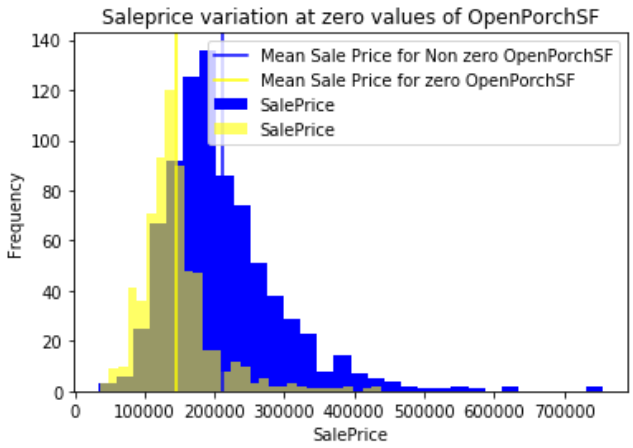
Step- 2 Imputation of Nan and Zero values

Nan values in Categorical variables:

For the Nan values we create a separate category 'Absent'.This would help to capture any new information such as average Sale Price in case a feature is absent. Variables like PoolQC, Fence, MiscFeature and Alley have a very high Nan. These may not be useful at all. However, we cannot make similar assumptions about population data. So we retain them for whatever little information they may provide.

Nan values and Zeros in Numeric variables:

More than the problem of Nan, we have a problem of zero values, as they were more in numbers. This could mean that a feature is not present. We tried to see the impact of a numerical feature being absent, on the 'SalePrice' variable by plotting the histogram of 'Sale Price'.
For example: In the histogram we found that Average SalePrice is much lower where OpenPorchSF is not present.



We treat Nan's and zero's as follows:

1. We create a seperate variable for each numerical column and label the absence of feature i.e 0 value as 1, and presence of a feature as 0.
2. We replace zeros with Nan and impute Nan's by linear interpolation.

Step-3 Identifying Outliers

1. We observed many values above z score of 3 in the target variable i.e Sales Price.
2. There were 22 such observations.
3. Before making a decision to remove or transform these outliers we looked at the linear relationship between Sale Price and GrLivArea.
4. Out of these 22 values most of the values are following a linear trend with GrLiveArea.But these values are high. If we normalize the data, this could be taken care of. However 2 points at bottom right are not following the trend. So these 2 points are outliers.
5. We later applied graphical tools to identify outliers such as residual plot.

EXPLORATORY DATA ANALYSIS

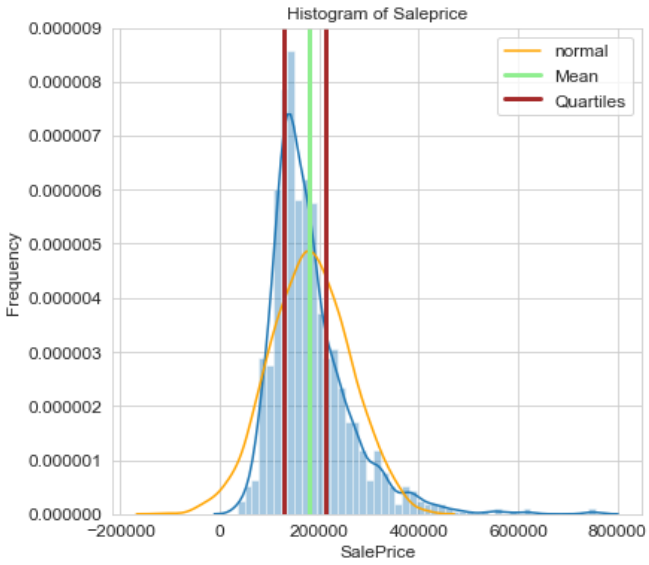
INTRODUCTION

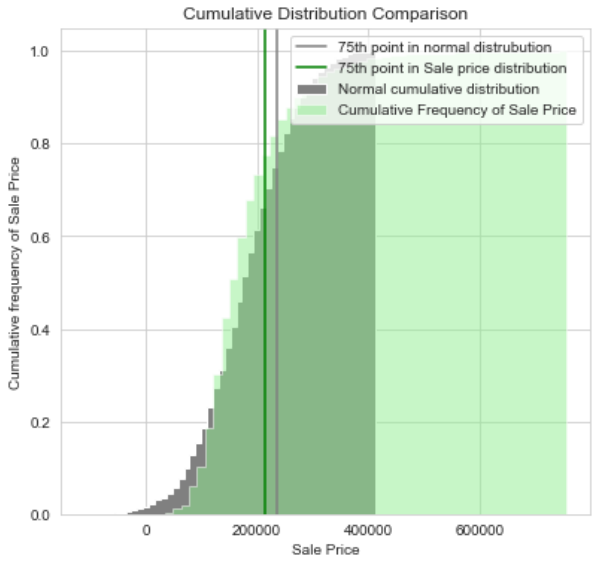
In this section, we will analyse the target variable and its relationship with independent variables. This will help in selecting predictive variables.

TARGET VARIABLE - Sale Price

We will look at the following:

1. Distribution plot of SalePrice alongside kde plot of a normal distribution.
2. Develop a cumulative distribution of Sale Price alongside cumulative normal distribution.





Observations:

1.The distribution is not normal. Mean Sale Price and median Sale Price, both have lower probabilities of occurrence than the mode.

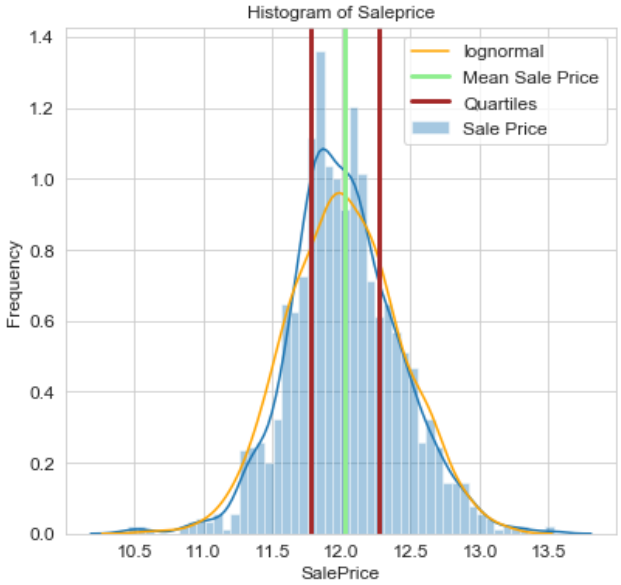
2. Distribution of SalePrice is leptokurtic. An example of a leptokurtic distribution is the Laplace distribution, which has tails that asymptotically approach zero more slowly than a Gaussian.

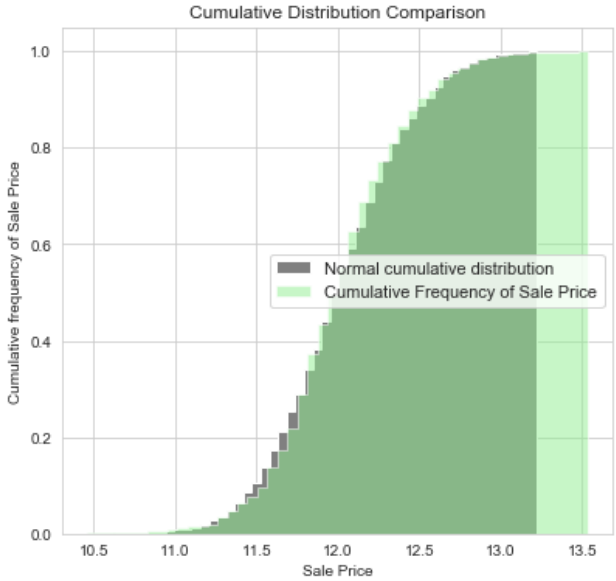
3. The distribution is right skewed. The range of upper 25% of data is around 60000 which is 3 times more than the Interquartile range.

4. Mean Sale Price is not a good representation and there are quite a number of outliers.

5. The cumulative distribution comparison makes it very clear that SalePrice does not conform to a normal pattern. There is also a greater chance of SalePrice in the range 100000 to 300000 occuring in comparison to what a normal distribution should have. We can also see that the range of upper 25% of Sale Price is abnormally greater than normal pattern.

LOG TRANSFORMATION OF TARGET VARIABLE 'SALE PRICE'





Observations:

The distribution of 'SalePrice' is very close to lognormal distribution.

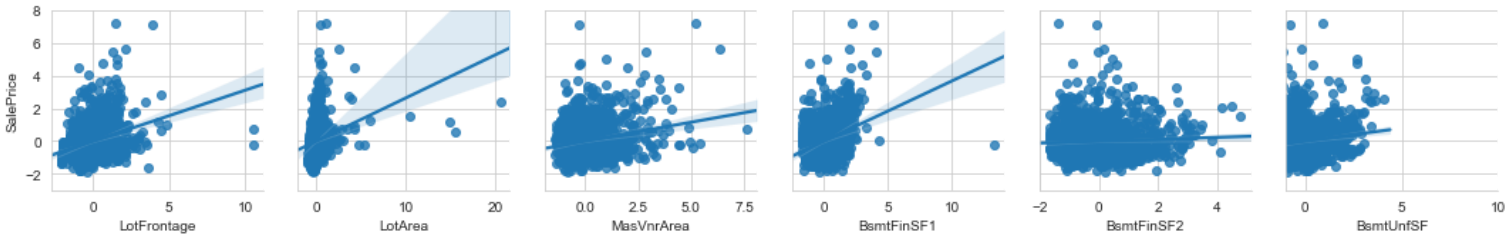
The tails are matching, though 'SalePrice' appears to be bimodal.

The range above the upper quartile has normalised to quite an extent.

The cumulative distribution of Sale Price is very close to cumulative lognormal distribution.

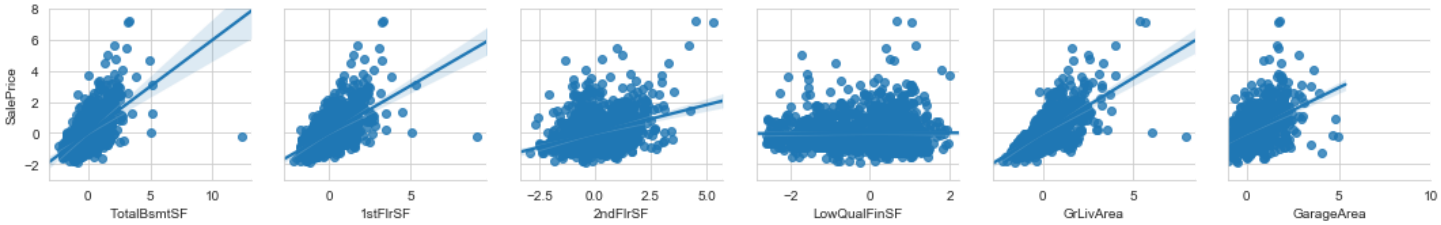
'SALE PRICE' RELATIONSHIP WITH INDEPENDENT NUMERICAL VARIABLES

PAIR GRID OF SCATTER PLOTS OF NUMERIC VARIABLE VS SALE PRICE:



Observations:

1. SalePrice has non constant variance with Variables LotFrontage, LotArea, MasVnrArea and BsmtSF1. Linear relationships cannot be clearly established.
2. BsmtUnfSF and BsmtFinSF2 both are weakly correlated with SalePrice.
3. SalePrice shows a weak linear relationship with BsmtFinSF1 and has non constant variance.

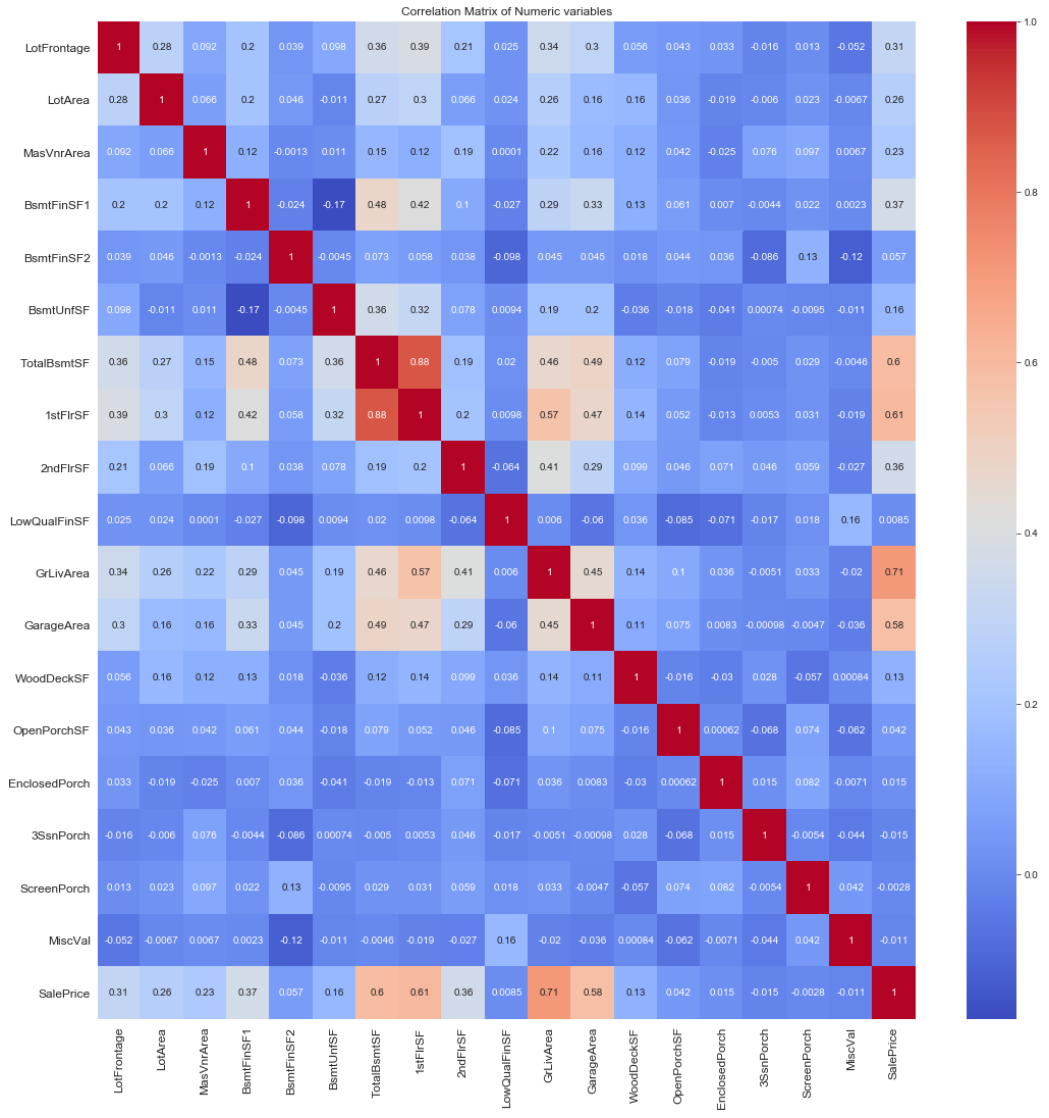


Observations

1. TotalBsmtSF and FirstFlrSF have a correlation of .6 with SalePrice though the relationship is somewhat linear. Variance in SalePrice increases as these independent variables increase , for eg we see a range from 100000 to 400000 with both these variables.Both these variables impact the average SalePrice. There is an outlier in both cases.
2. The variance in SalePrice is non constant with SecondFlrSF with a weak linear relationship. It has a .3 correlation with SalePrice.

3. GrLivArea and GarageArea have .7 and .58 correlation with SalePrice. SalePrice variance increases with higher values of GrLivArea and eventually spreads out of the group at few very high values. SalePrice has a constant variance, with GarageArea and here too it has outliers. GrLivArea shows some linearity with SalePrice.

MULTICOLLINEARITY BETWEEN NUMERIC VARIABLES AS OBSERVED FROM CORRELATION HEATMAP:



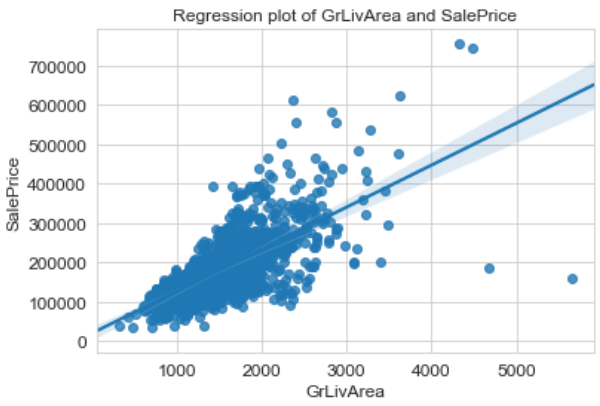
Observations on correlation:

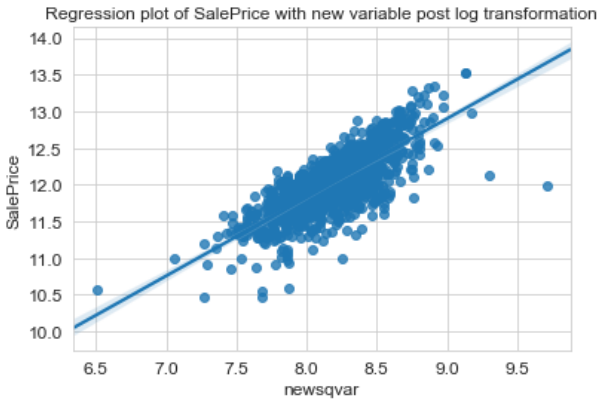
There are a lot of variables which have a correlation with each other.

1. TotalBsmtSF nad 1stFlrSF have a high correlation with each other i.e .88.
2. GrLivArea and TotalBsmtSF have correlation of .61.
3. There is correlation between multiple variables- MasVnrArea, TotalBsmtSF, 1stFlrSF, GrLivArea and GarageArea

INDEPENDENT VARIABLE GRLIVAREA AND ITS TRANSFORMATION:

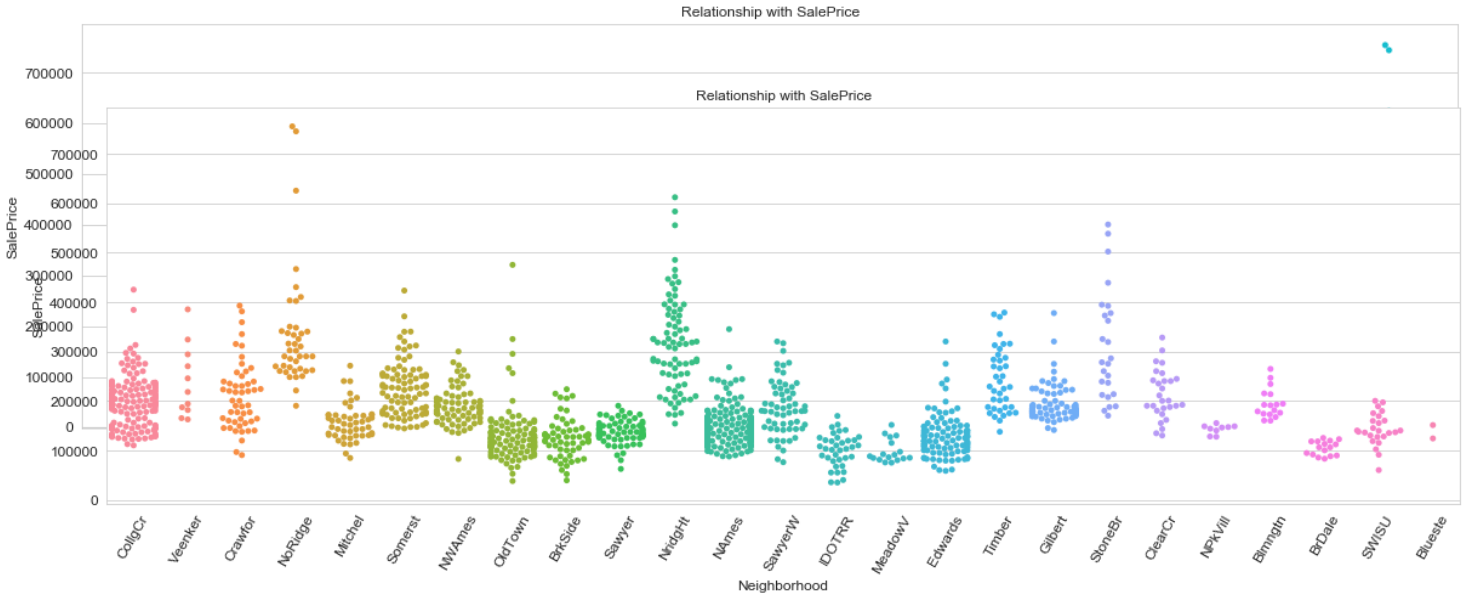
We have seen earlier that some variables have high correlation with each other and linear relation with SalePrice is also similar. On this basis we create a new variable- 'newsqvar' which is a combination of 'TotalBsmtSF','1stFlrSF' and 'GrLivArea'. Let's take the log value of this variable and see the regression plot with SalePrice. Great! This new variable has a strong linear relation with SalePrice and a constant variance as well. The correlation also has improved from .70 to .76





'SALE PRICE' RELATIONSHIP WITH INDEPENDENT CATEGORICAL VARIABLES

SWARMPLOTS OF CATEGORICAL VARIABLES WITH SALE PRICE:



Observations with some categorical variables of interest:

Neighborhood: This variable has 25 categories. Overall it looks like each category has a differentiated range in SalePrice.Most categories also have an almost equally distributed count. Few categories look similar in spread and average SalePrice like OldTown, BrkSide and Edwards.

ExterQual: The categories TA and Gd almost have an equal count. The average SalePrice of Gd is higher than TA.

ExterCond: Category TA dominates in counts and has a higher average SalePrice, compared to Gd.

Foundation: The range in SalePrice and average SalePrice is differentiated in the 3 categories-PConc, CBlock, and BrkTil. The counts vary, with PConc count leading and followed by CBlock.

BsmtQual:The SalePrice range in the 3 categories- Gd,TA and Ex differs. The count of Gd category is the highest, followed by TA. The average SalePrice of Ex category is highest, followed by Gd category.

BsmtFinType1: The Average SalePrice and range differs in the 3 categories GLQ, ALQ and Unf, with GLQ leading and having almost equal counts with Unf.

KitchenQual: The categories Gd, TA and Ex have differentiated SalePrice range and Average price with Ex having the highest Average SalePrice and Gd having the highest counts.

GarageFinish:The 3 categories RfN, Unf and Fin have differentiated Avg Sale price, different price range and almost equal in counts.

OverallQual: The Average SalePrice and its range is increasing as the parameters increase from 1 to 10.Counts are mainly spread in categories 4 to 8.

FullBath: This variable has a different Sale Price range and average SalePrice in 3 categories- 1,2, and 3, with 3 having the highest Avg SalePrice followed by category 2. Counts are highest for category 1 followed by category 2.

HalfBath: This variable has the highest category count of 0 and average Sale Price is higher for 1.

BedroomAbvGr: This variable has maximum counts spread between categories 2,3,4 with 3 being highest in count. Spread in Sale Price and Average SalePrice varies slightly between these categories.

TotRmsAbvGrd: The categories in this variable are differentiated in terms of range in Sale Price and Average Sale Price.

GarageCars: The categories 0,1,2,3 are well differentiated in terms of Average Sale Price and range in SalePrice. Category 2 has the highest counts followed by 1.

YearBuilt: There is a steady increase in Average Sale Price, as well as counts from the middle half of YearBuilt.

YearRemodAdd: There is an increasing trend in Average SalePrice over the years. ZeroMasVnrArea, ZeroWoodDeckSF, ZeroBsmtFinSF1, ZeroOpenPorchSF and ZeroScreenPorch show lower Average Sale price with category '1'. Whereas ZeroEnclosedPorch and ZeroBsmtFinSF2 have higher Average Sale Price with category '1'.

TRANSFORMING LABELS OF CATEGORICAL VARIABLES.

Based on our observations we find that there are categories, sensitive to Average Sale Price. For including these variables in the prediction models we need to assign them numerical labels. Here, I have assigned the labels based on average SalePrice. There are also easy options available with OneHotEcoder and LabelEncoder for this job.

IN- DEPTH ANALYSIS (MACHINE LEARNING)

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.

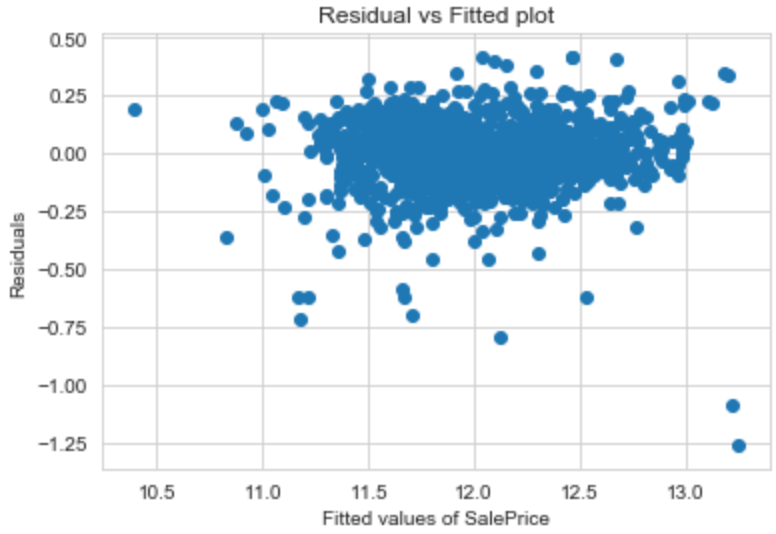
	Features	Coefficients
1	newsqvar	0.38
2	OverallQual	0.29
3	Neighborhood	0.18
4	HalfBath	0.12
5	CentralAir	0.11
6	Functional	0.11
7	YearRemodAdd	0.11
8	BsmtExposure	0.09
9	MSZoning	0.08
10	TotRmsAbvGrd	0.08
11	KitchenQual	0.08
12	GarageCars	0.08
13	Condition1	0.06
14	LotArea	0.06
15	FireplaceQu	0.06
16	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.