HOUSING PROJECT

Submitted by-

KAUSHIK VEER

ACKNOWLEDGMENT

All thanks to fliprobo technologies for providing me the opportunity to work on this project. I leaned alot from this project.

INTRODUCTION

- Business Problem Framing

Houses are one of the necessary need of each and every person around the globe and therefore housing and real estate market is one of the markets which is one of the major contributors in the world's economy. It is a very large market and there are various companies working in the domain. Data science comes as a very important tool to solve problems in the domain to help the companies increase their overall revenue, profits, improving their marketing strategies and focusing on changing trends in house sales and purchases. Predictive modelling, Market mix modelling, recommendation systems are some of the machine learning techniques used for achieving the business goals for housing companies. Our problem is related to one such housing company.

We are required to model the price of houses with the available independent variables. This model will then be used by the management to understand how exactly the prices vary with the variables. They can accordingly manipulate the strategy of the firm and concentrate on areas that will yield high returns. Further, the model will be a good way for the management to understand the pricing dynamics of a new market.

- Conceptual Background of the Domain Problem

A US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia.

The company is looking at prospective properties to buy houses to enter the market. We are required to build a model using Machine Learning in

order to predict the actual value of the prospective properties and decide whether to invest in them or not. For this company wants to know:

• Which variables are important to predict the price of variable?

• How do these variables describe the price of the house?

• Review of Literature

Surprise housing company is a US based company is actually trying to buy the houses at a low price in Australian region and sell them at high price and making profits out of it.

They are trying to use data analytics to know in what areas they should be looking as houses are necessary need for every human being but they want to know which are the important attributes and how do they correlate with the prices of the house.

We will be finding these important features and how do they affect positively and negatively with sale price of the houses and use them to predict the housing prices using machine learning models.

• Motivation for the Problem Undertaken

My objective is to find the important attributes and how they affect the prices of the houses using data science.

To use analytics in areas like real estate and see how we can use data science to perform various analysis to help the company to make better decisions and generate profits is a interesting task as I will be able to learn a lot in this field and gain experience using data science.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

• Data contains 1460 entries each having 81 variables.

• Data contains Null values. You need to treat them using the domain knowledge and your own understanding.

• Extensive EDA has been performed to gain relationships of important variable and price.

• Data contains numerical as well as categorical variable. Which we will handle and convert the categorical variable into numerical for model evaluation.

• We will build Machine Learning models and determine the optimal values of Hyper Parameters.

• We will find important features which affect the price positively or negatively.

• Two datasets are being provided to you (test.csv, train.csv). we will train on train.csv dataset and predict on test.csv file.

- Data Sources and their formats

The data has been provided by the Surprise housing company. The data contains 1460 entries each having 81 variables.

This data contains numerical as well as object type data.

Following are the 81 variables with description :-

MSSubClass: Identifies the type of dwelling involved in the sale.

20	1-STORY 1946 & NEWER ALL STYLES
30	1-STORY 1945 & OLDER
40	1-STORY W/FINISHED ATTIC ALL AGES
45	1-1/2 STORY - UNFINISHED ALL AGES
50	1-1/2 STORY FINISHED ALL AGES
60	2-STORY 1946 & NEWER
70	2-STORY 1945 & OLDER
75	2-1/2 STORY ALL AGES
80	SPLIT OR MULTI-LEVEL
85	SPLIT FOYER
90	DUPLEX - ALL STYLES AND AGES
120	1-STORY PUD (Planned Unit Development) - 1946 & NEWER
150	1-1/2 STORY PUD - ALL AGES
160	2-STORY PUD - 1946 & NEWER
180	PUD - MULTILEVEL - INCL SPLIT LEV/FOYER
190	2 FAMILY CONVERSION - ALL STYLES AND AGES

MSZoning: Identifies the general zoning classification of the sale.

A	Agriculture
C	Commercial
FV	Floating Village Residential
I	Industrial
RH	Residential High Density
RL	Residential Low Density

RP	Residential Low Density Park
RM	Residential Medium Density

LotFrontage: Linear feet of street connected to property

LotArea: Lot size in square feet

Street: Type of road access to property

Grvl	Gravel
Pave	Paved

Alley: Type of alley access to property

Grvl	Gravel
Pave	Paved
NA	No alley access

LotShape: General shape of property

Reg	Regular
IR1	Slightly irregular
IR2	Moderately Irregular
IR3	Irregular

LandContour: Flatness of the property

Lvl	Near Flat/Level
Bnk	Banked - Quick and significant rise from street grade to building
HLS	Hillside - Significant slope from side to side
Low	Depression

Utilities: Type of utilities available

AllPub	All public Utilities (E,G,W,& S)
NoSewr	Electricity, Gas, and Water (Septic Tank)
NoSeWa	Electricity and Gas Only
ELO	Electricity only

LotConfig: Lot configuration

Inside	Inside lot
Corner	Corner lot
CulDSac	Cul-de-sac
FR2	Frontage on 2 sides of property
FR3	Frontage on 3 sides of property

LandSlope: Slope of property

Gtl	Gentle slope

Mod	Moderate Slope
Sev	Severe Slope

Neighborhood: Physical locations within Ames city limits

Blmngtn	Bloomington Heights
Blueste	Bluestem
BrDale	Briardale
BrkSide	Brookside
ClearCr	Clear Creek
CollgCr	College Creek
Crawfor	Crawford
Edwards	Edwards
Gilbert	Gilbert
IDOTRR	Iowa DOT and Rail Road
MeadowV	Meadow Village
Mitchel	Mitchell
Names	North Ames
NoRidge	Northridge
NPkVill	Northpark Villa
NridgHt	Northridge Heights
NWAmes	Northwest Ames
OldTown	Old Town
SWISU	South & West of Iowa State University
Sawyer	Sawyer

SawyerW	Sawyer West
Somerst	Somerset
StoneBr	Stone Brook
Timber	Timberland
Veenker	Veenker

Condition1: Proximity to various conditions

Artery	Adjacent to arterial street
Feedr	Adjacent to feeder street
Norm	Normal
RRNn	Within 200' of North-South Railroad
RRAn	Adjacent to North-South Railroad
PosN	Near positive off-site feature--park, greenbelt, etc.
PosA	Adjacent to postive off-site feature
RRNe	Within 200' of East-West Railroad
RRAe	Adjacent to East-West Railroad

Condition2: Proximity to various conditions (if more than one is present)

Artery	Adjacent to arterial street
Feedr	Adjacent to feeder street
Norm	Normal
RRNn	Within 200' of North-South Railroad
RRAn	Adjacent to North-South Railroad

PosN Near positive off-site feature--park, greenbelt, etc.

PosA Adjacent to postive off-site feature

RRNe Within 200' of East-West Railroad

RRAe Adjacent to East-West Railroad

BldgType: Type of dwelling

1Fam Single-family Detached

2FmCon Two-family Conversion; originally built as one-family dwelling

Duplx Duplex

TwnhsE Townhouse End Unit

TwnhsI Townhouse Inside Unit

HouseStyle: Style of dwelling

1Story One story

1.5Fin One and one-half story: 2nd level finished

1.5Unf One and one-half story: 2nd level unfinished

2Story Two story

2.5Fin Two and one-half story: 2nd level finished

2.5Unf Two and one-half story: 2nd level unfinished

SFoyer Split Foyer

SLvl Split Level

OverallQual: Rates the overall material and finish of the house

10 Very Excellent

9 Excellent

8 Very Good

7 Good

6 Above Average

5 Average

4 Below Average

3 Fair

2 Poor

1 Very Poor

OverallCond: Rates the overall condition of the house

10 Very Excellent

9 Excellent

8 Very Good

7 Good

6 Above Average

5 Average

4 Below Average

3 Fair

2 Poor

1 Very Poor

YearBuilt: Original construction date

YearRemodAdd: Remodel date (same as construction date if no remodeling or additions)

RoofStyle: Type of roof

Flat	Flat
Gable	Gable
Gambrel	Gabrel (Barn)
Hip	Hip
Mansard	Mansard
Shed	Shed

RoofMatl: Roof material

ClyTile	Clay or Tile
CompShg	Standard (Composite) Shingle
Membran	Membrane
Metal	Metal
Roll	Roll
Tar&Grv	Gravel & Tar
WdShake	Wood Shakes
WdShngl	Wood Shingles

Exterior1st: Exterior covering on house

AsbShng Asbestos Shingles

AsphShn Asphalt Shingles

BrkComm Brick Common

BrkFaceBrick Face

CBlock Cinder Block

CemntBd Cement Board

HdBoard Hard Board

ImStuccImitation Stucco

MetalSdMetal Siding

Other Other

Plywood Plywood

PreCast PreCast

Stone Stone

Stucco Stucco

VinylSdVinyl Siding

Wd Sdng Wood Siding

WdShing Wood Shingles

Exterior2nd: Exterior covering on house (if more than one material)

AsbShng Asbestos Shingles

AsphShn Asphalt Shingles

BrkComm Brick Common

BrkFaceBrick Face

CBlock Cinder Block

CemntBd Cement Board

HdBoard Hard Board

ImStucc Imitation Stucco

MetalSd Metal Siding

Other Other

Plywood Plywood

PreCast PreCast

Stone Stone

Stucco Stucco

VinylSd Vinyl Siding

Wd Sdng Wood Siding

WdShing Wood Shingles

MasVnrType: Masonry veneer type

BrkCmn Brick Common

BrkFace Brick Face

CBlock Cinder Block

None None

Stone Stone

MasVnrArea: Masonry veneer area in square feet

ExterQual: Evaluates the quality of the material on the exterior

Ex	Excellent
Gd	Good
TA	Average/Typical
Fa	Fair
Po	Poor

ExterCond: Evaluates the present condition of the material on the exterior

Ex	Excellent
Gd	Good
TA	Average/Typical
Fa	Fair
Po	Poor

Foundation: Type of foundation

BrkTil	Brick & Tile
CBlock	Cinder Block
PConc	Poured Contrete
Slab	Slab
Stone	Stone
Wood	Wood

BsmtQual: Evaluates the height of the basement

Ex Excellent (100+ inches)

Gd Good (90-99 inches)

TA Typical (80-89 inches)

Fa Fair (70-79 inches)

Po Poor (<70 inches

NA No Basement

BsmtCond: Evaluates the general condition of the basement

Ex Excellent

Gd Good

TA Typical - slight dampness allowed

Fa Fair - dampness or some cracking or settling

Po Poor - Severe cracking, settling, or wetness

NA No Basement

BsmtExposure: Refers to walkout or garden level walls

Gd Good Exposure

Av Average Exposure (split levels or foyers typically score average or above)

Mn Mimimum Exposure

No No Exposure

NA No Basement

BsmtFinType1: Rating of basement finished area

GLQ	Good Living Quarters
ALQ	Average Living Quarters
BLQ	Below Average Living Quarters
Rec	Average Rec Room
LwQ	Low Quality
Unf	Unfinshed
NA	No Basement

BsmtFinSF1: Type 1 finished square feet

BsmtFinType2: Rating of basement finished area (if multiple types)

GLQ	Good Living Quarters
ALQ	Average Living Quarters
BLQ	Below Average Living Quarters
Rec	Average Rec Room
LwQ	Low Quality
Unf	Unfinshed
NA	No Basement

BsmtFinSF2: Type 2 finished square feet

BsmtUnfSF: Unfinished square feet of basement area

TotalBsmtSF: Total square feet of basement area

Heating: Type of heating

Floor	Floor Furnace
GasA	Gas forced warm air furnace
GasW	Gas hot water or steam heat
Grav	Gravity furnace
OthW	Hot water or steam heat other than gas
Wall	Wall furnace

HeatingQC: Heating quality and condition

Ex	Excellent
Gd	Good
TA	Average/Typical
Fa	Fair
Po	Poor

CentralAir: Central air conditioning

N	No
Y	Yes

Electrical: Electrical system

SBrkr	Standard Circuit Breakers & Romex
FuseA	Fuse Box over 60 AMP and all Romex wiring (Average)
FuseF	60 AMP Fuse Box and mostly Romex wiring (Fair)
FuseP	60 AMP Fuse Box and mostly knob & tube wiring (poor)
Mix	Mixed

1stFlrSF: First Floor square feet

2ndFlrSF: Second floor square feet

LowQualFinSF: Low quality finished square feet (all floors)

GrLivArea: Above grade (ground) living area square feet

BsmtFullBath: Basement full bathrooms

BsmtHalfBath: Basement half bathrooms

FullBath: Full bathrooms above grade

HalfBath: Half baths above grade

Bedroom: Bedrooms above grade (does NOT include basement bedrooms)

Kitchen: Kitchens above grade

KitchenQual: Kitchen quality

Ex	Excellent
Gd	Good
TA	Typical/Average
Fa	Fair
Po	Poor

TotRmsAbvGrd: Total rooms above grade (does not include bathrooms)

Functional: Home functionality (Assume typical unless deductions are warranted)

Typ	Typical Functionality
Min1	Minor Deductions 1
Min2	Minor Deductions 2
Mod	Moderate Deductions
Maj1	Major Deductions 1
Maj2	Major Deductions 2
Sev	Severely Damaged
Sal	Salvage only

Fireplaces: Number of fireplaces

FireplaceQu: Fireplace quality

Ex Excellent - Exceptional Masonry Fireplace

Gd Good - Masonry Fireplace in main level

TA Average - Prefabricated Fireplace in main living area or Masonry Fireplace in basement

Fa Fair - Prefabricated Fireplace in basement

Po Poor - Ben Franklin Stove

NA No Fireplace

GarageType: Garage location

2Types More than one type of garage

Attchd Attached to home

Basment Basement Garage

BuiltIn Built-In (Garage part of house - typically has room above garage)

CarPort Car Port

Detchd Detached from home

NA No Garage

GarageYrBlt: Year garage was built

GarageFinish: Interior finish of the garage

Fin Finished

RFn	Rough Finished
Unf	Unfinished
NA	No Garage

GarageCars: Size of garage in car capacity

GarageArea: Size of garage in square feet

GarageQual: Garage quality

Ex	Excellent
Gd	Good
TA	Typical/Average
Fa	Fair
Po	Poor
NA	No Garage

GarageCond: Garage condition

Ex	Excellent
Gd	Good
TA	Typical/Average
Fa	Fair
Po	Poor
NA	No Garage

PavedDrive: Paved driveway

Y	Paved
P	Partial Pavement
N	Dirt/Gravel

WoodDeckSF: Wood deck area in square feet

OpenPorchSF: Open porch area in square feet

EnclosedPorch: Enclosed porch area in square feet

3SsnPorch: Three season porch area in square feet

ScreenPorch: Screen porch area in square feet

PoolArea: Pool area in square feet

PoolQC: Pool quality

Ex	Excellent
Gd	Good
TA	Average/Typical
Fa	Fair

NA No Pool

Fence: Fence quality

GdPrv Good Privacy
MnPrv Minimum Privacy
GdWo Good Wood
MnWw Minimum Wood/Wire
NA No Fence

MiscFeature: Miscellaneous feature not covered in other categories

Elev Elevator
Gar2 2nd Garage (if not described in garage section)
Othr Other
Shed Shed (over 100 SF)
TenC Tennis Court
NA None

MiscVal: $Value of miscellaneous feature

MoSold: Month Sold (MM)

YrSold: Year Sold (YYYY)

SaleType: Type of sale

WD	Warranty Deed - Conventional
CWD	Warranty Deed - Cash
VWD	Warranty Deed - VA Loan
New	Home just constructed and sold
COD	Court Officer Deed/Estate
Con	Contract 15% Down payment regular terms
ConLw	Contract Low Down payment and low interest
ConLI	Contract Low Interest
ConLD	Contract Low Down
Oth	Other

SaleCondition: Condition of sale

Normal	Normal Sale
Abnorml	Abnormal Sale - trade, foreclosure, short sale
AdjLand	Adjoining Land Purchase
Alloca	Allocation - two linked properties with separate deeds, typically condo with a garage unit
Family	Sale between family members
Partial	Home was not completed when last assessed (associated with New Homes)

Here is the snapshot of the data :-

```
In [139]: df.head(7)
```

Out[139]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	M
0	127	120	RL	NaN	4928	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	
1	889	20	RL	95.0	15865	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	
2	793	60	RL	92.0	9920	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	
3	110	20	RL	105.0	11751	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	
4	422	20	RL	NaN	16635	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	
5	1197	60	RL	58.0	14054	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	
6	561	20	RL	NaN	11341	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	

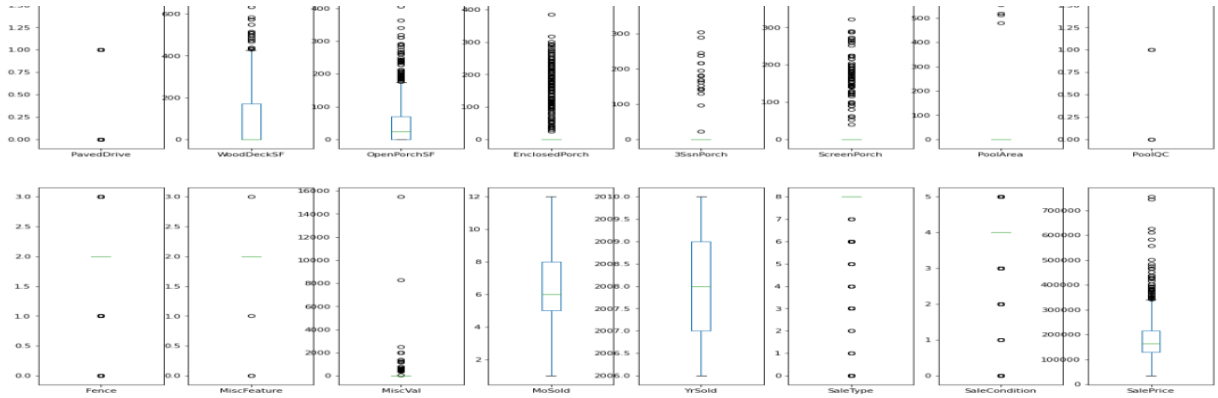
- Data Pre-processing Done

We saw that the data had some nan values present in the data so we replaced the nan vales with mean for non-categorical variables and we replaced the categorical variables with mode(most frequently occurring element in that column).

```
In [140]: #filling the nan values for non-categorical
df['LotFrontage']=df['LotFrontage'].fillna(df['LotFrontage'].mean())
```

```
In [141]: #filling the nan values for categorical data
lis=['Alley','MasVnrType','MasVnrArea','BsmtQual','BsmtCond','BsmtExposure','BsmtFinType1','BsmtFinType2','FireplaceQu',
     'GarageType','GarageYrBlt','GarageFinish','GarageQual','GarageCond','PoolQC','Fence','MiscFeature']
for i in lis:
    df[i]=df[i].fillna(df[i].mode().iloc[0])
```

We also saw some outliers present in the data which we removed through quantile method and cleaned the data.



we can definitely see some outliers present in the data in almost all the columns we will try to remove the outliers with quantile method as after using zscore sometimes we still see some outliers

We used label encoder for encoding the categorical columns for converting its object type data to numerical so that it will help us in prediction models.

We also saw some skewness present in the data so we used power transform method to remove skewness from non-categorical data.

we can see the skewness present we will try to remove the skewness using power transform from the non categorical data

[182]:
```
from sklearn.preprocessing import power_transform
x[['LotArea','MasVnrArea','BsmtFinSF1','BsmtFinSF2','BsmtUnfSF','TotalBsmtSF',
        '1stFlrSF','2ndFlrSF','LowQualFinSF','GrLivArea','LotFrontage']]=power_transform(x[['LotArea','MasVnrArea','BsmtFinSF1'
        '1stFlrSF','2ndFlrSF','LowQualFinSF','GrLivArea','LotFrontage']],method='yeo-johnson')
x
```

[182]:

	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	LotConfig	...	ScreenPorch	PoolArea	PoolQC	Fence
0	120	3	0.049487	-1.306083	1	0	0	3	0	4	...	0	0	2	2
1	20	3	1.312560	1.356458	1	0	0	3	0	4	...	216	0	2	2
2	60	3	1.156220	0.113089	1	0	0	3	0	1	...	0	0	2	2
3	20	3	1.831046	0.530989	1	0	0	3	0	4	...	0	0	2	2

- ## Hardware and Software Requirements and Tools Used

Hardware specifications are :-

Ryzen 5

16gb ram

RTX 2070 super graphics

Software used :-

Operating system : windows 10

Jupyter notebook and anaconda navigator- for coding and using data analytics tools and libraries.

Libraries used are matplot and seaborn for visualization purpose and scipy and sklearn for building models and pre-processing of data.

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

We saw that the data have some categorical data for which we used label encoder to convert the object type data to numerical.

We found null or nan values present in the data so we replaced them with mean for non-categorical data and by mode for categorical variables.

We found outliers and removed them with quantile method as we cannot proceed to model building without removing the outliers.

We removed the skewness from non-categorical variables using power-transform method.

We used standard scaler to scale the data.

- Testing of Identified Approaches (Algorithms)

We split the data into into X and Y where x hold all the input variables after perfroming all the EDA and cleansing of data and Y holds the output variable which is a Sale price.

The data is in float type which clearly indicates that it's a regression problem and will need regression models to train test and predict the output.

We used train test split to split the data and perform machine learning tasks.

- Run and Evaluate selected models

We used train_test_split to split the data into xtrain,xtest and ytrain,ytest with test size set to 0.20 which is 20% of the data set. It means that ytest and xtest will hold 20% of the random data which the model will use for testing purpose and predict the data for the xtest and chech how accurate it is with ytest.

Below is the snaopshot of the code :-

Builing prediction models

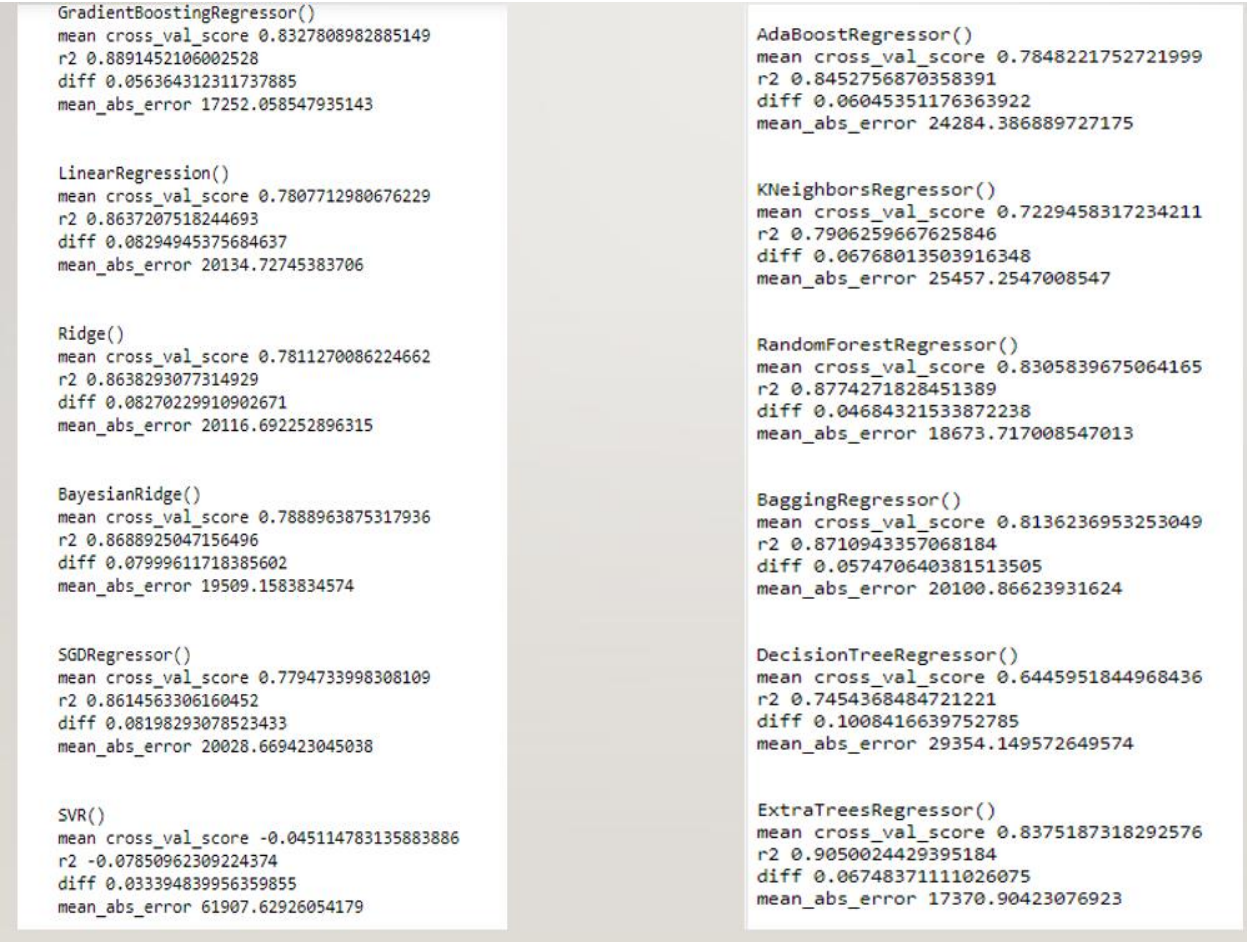
```
191]: #making a list of the regression models which are to be tested with the data set
models=[GradientBoostingRegressor(),LinearRegression(),Ridge(),BayesianRidge(),SGDRegressor(),SVR(),
        AdaBoostRegressor(),KNeighborsRegressor(),RandomForestRegressor(),BaggingRegressor(),
       DecisionTreeRegressor(),ExtraTreesRegressor()]
```

we will check the metrics such as r2_score and mean cross val score as well as mean absolute erroe before choosing the best model

```
217]: #making a for loop to check the models and their mean cross_val score with scoring set to r2
for i in models:
    xtrain,xtest,ytrain,ytest=train_test_split(x,y,random_state=85,test_size=0.20)
    score=cross_val_score(i,xtrain,ytrain,cv=5,scoring='r2').mean()
    i.fit(xtrain,ytrain)
    ypred=i.predict(xtest)
    if r2_score(ytest,ypred)>score:
        diff=r2_score(ytest,ypred)-score
    else:
        diff=score-r2_score(ytest,ypred)
    print(i)
    print('mean_cross_val_score',score)
    print('r2',r2_score(ytest,ypred))
    print('diff',diff)
    print('mean_abs_error',mean_absolute_error(ytest, ypred))
    print('\n')
```

We included cross val score and cv set to 5 and checking the mean score. Random state is set to 85, we will be checking the r2 score and difference between the r2 score and cross val mean score. Where diff represents the difference in cross_val mean score and r2_score and check the difference. We can’t choose the model if the difference is high as it leads to overfitting and underfitting problems.

Below is the image of the results that we got from the above code:-



What we observed from the results:-

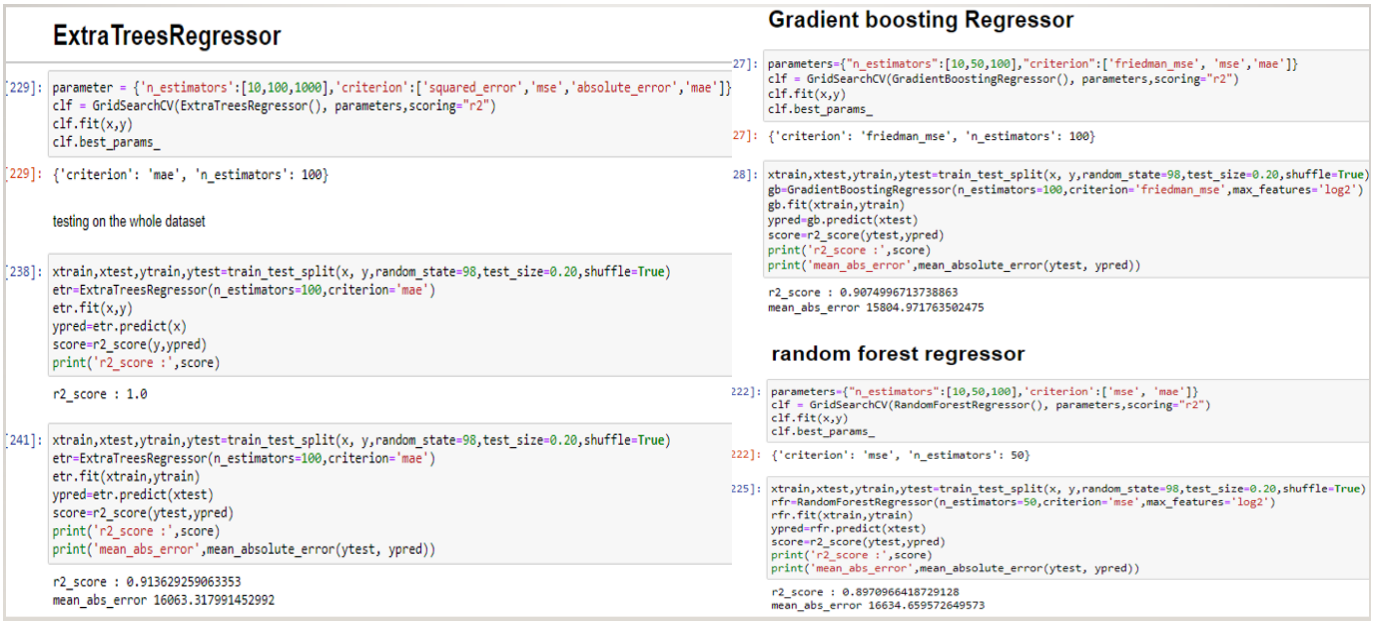
1) Our best performing model is ExtratreeRegressor with 90% accuracy.
2) After extra tree our 2nd and 3rd best models are random forest regressor with close to 88% r2 score and gradient boosting with close to 89% r2 score.
3) We need to hyper tune these models for checking the best accuracy and select the best model.

- Key Metrics for success in solving problem under consideration

We used r2_score for checking accuracy of the model.

We used cross val score and checking the mean score with sv set to 5 and checking the difference in the mean cross val score and r2 score to avoid other problems and choose the models with best accuracy and least difference.

We used hyper tuning the models :-



ExtraTreesRegressor

```
[229]: parameter = {'n_estimators':[10,100,1000],'criterion':['squared_error','mse','absolute_error','mae']}
       clf = GridSearchCV(ExtraTreesRegressor(), parameters,scoring="r2")
       clf.fit(x,y)
       clf.best_params_

[229]: {'criterion': 'mae', 'n_estimators': 100}
```

testing on the whole dataset

```
[238]: xtrain,xtest,ytrain,ytest=train_test_split(x, y,random_state=98,test_size=0.20,shuffle=True)
       etr=ExtraTreesRegressor(n_estimators=100,criterion='mae')
       etr.fit(x,y)
       ypred=etr.predict(x)
       score=r2_score(y,ypred)
       print('r2_score :',score)

       r2_score : 1.0

[241]: xtrain,xtest,ytrain,ytest=train_test_split(x, y,random_state=98,test_size=0.20,shuffle=True)
       etr=ExtraTreesRegressor(n_estimators=100,criterion='mae')
       etr.fit(xtrain,ytrain)
       ypred=etr.predict(xtest)
       score=r2_score(ytest,ypred)
       print('r2_score :',score)
       print('mean_abs_error',mean_absolute_error(ytest, ypred))

       r2_score : 0.913629259063353
       mean_abs_error 16063.317991452992
```

Gradient boosting Regressor

```
27]: parameters={"n_estimators":[10,50,100],"criterion":['friedman_mse', 'mse','mae']}
     clf = GridSearchCV(GradientBoostingRegressor(), parameters,scoring="r2")
     clf.fit(x,y)
     clf.best_params_

27]: {'criterion': 'friedman_mse', 'n_estimators': 100}

28]: xtrain,xtest,ytrain,ytest=train_test_split(x, y,random_state=98,test_size=0.20,shuffle=True)
     gb=GradientBoostingRegressor(n_estimators=100,criterion='friedman_mse',max_features='log2')
     gb.fit(xtrain,ytrain)
     ypred=gb.predict(xtest)
     score=r2_score(ytest,ypred)
     print('r2_score :',score)
     print('mean_abs_error',mean_absolute_error(ytest, ypred))

     r2_score : 0.9074996713738863
     mean_abs_error 15804.971763502475
```

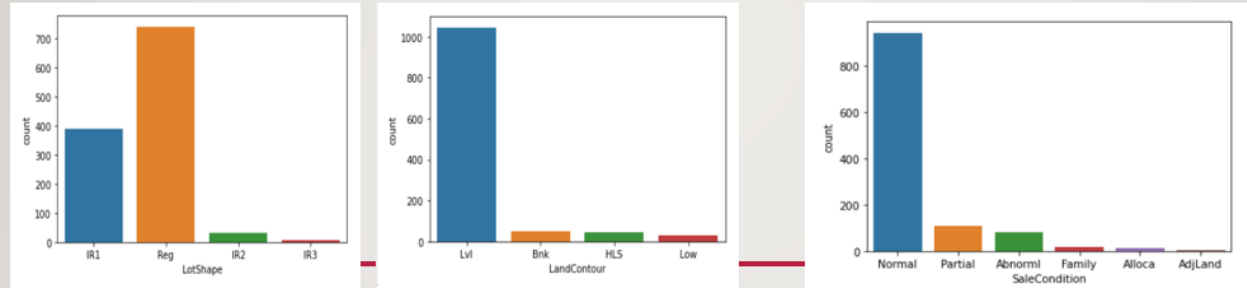
random forest regressor

```
222]: parameters={"n_estimators":[10,50,100],'criterion':['mse', 'mae']}
      clf = GridSearchCV(RandomForestRegressor(), parameters,scoring="r2")
      clf.fit(x,y)
      clf.best_params_

222]: {'criterion': 'mse', 'n_estimators': 50}

225]: xtrain,xtest,ytrain,ytest=train_test_split(x, y,random_state=98,test_size=0.20,shuffle=True)
      rfr=RandomForestRegressor(n_estimators=50,criterion='mse',max_features='log2')
      rfr.fit(xtrain,ytrain)
      ypred=rfr.predict(xtest)
      score=r2_score(ytest,ypred)
      print('r2_score :',score)
      print('mean_abs_error',mean_absolute_error(ytest, ypred))

      r2_score : 0.8970966418729128
      mean_abs_error 16634.659572649573
```

What we observed was:-

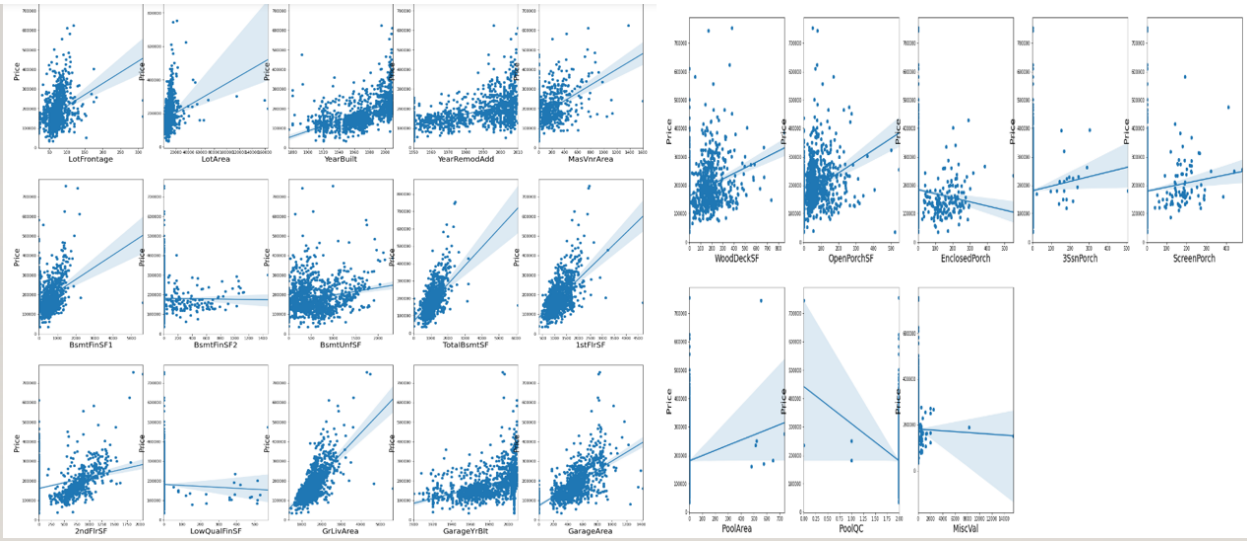
- After hyper tuning we got close to 90% accuracy and with less error with random forest regressor.
- after hyper tuning on Gradient boosting we got the accuracy of 90%.
- we found our best performing model with 91% accuracy which is ExtraTreeRegressor.

Visualizations



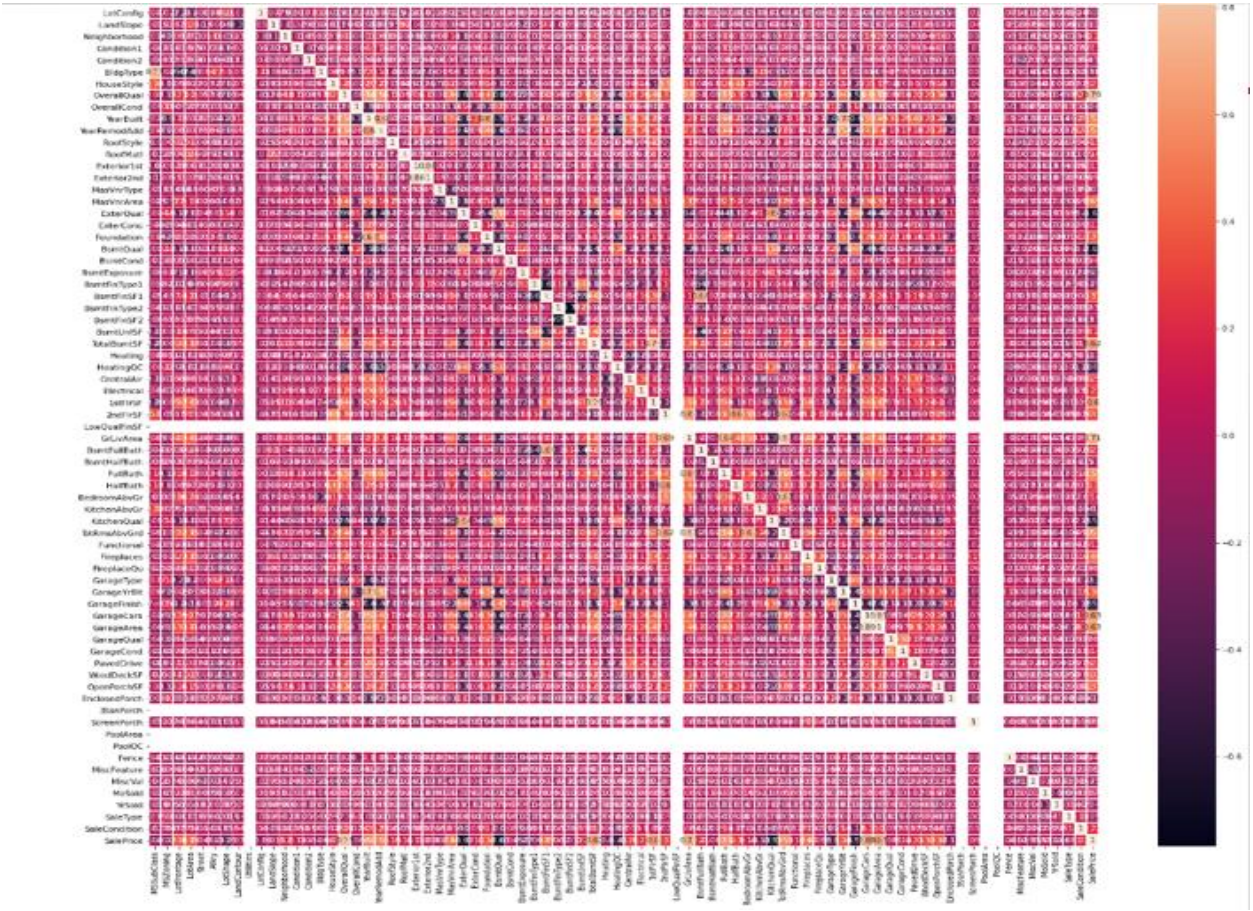
- From the above count plots of object type data what we can observe is :-
- In lotshape the highest number of counts are Reg followed by IR1
- In land contour lvl has the highest majority and rest all other similar to each other.
- In sale condition normal sale have the highest number of counts followed by partial. Sales condition is strong with normal people.

We used scatter and reg plot to check the positive and negative correlation in the variables with price.



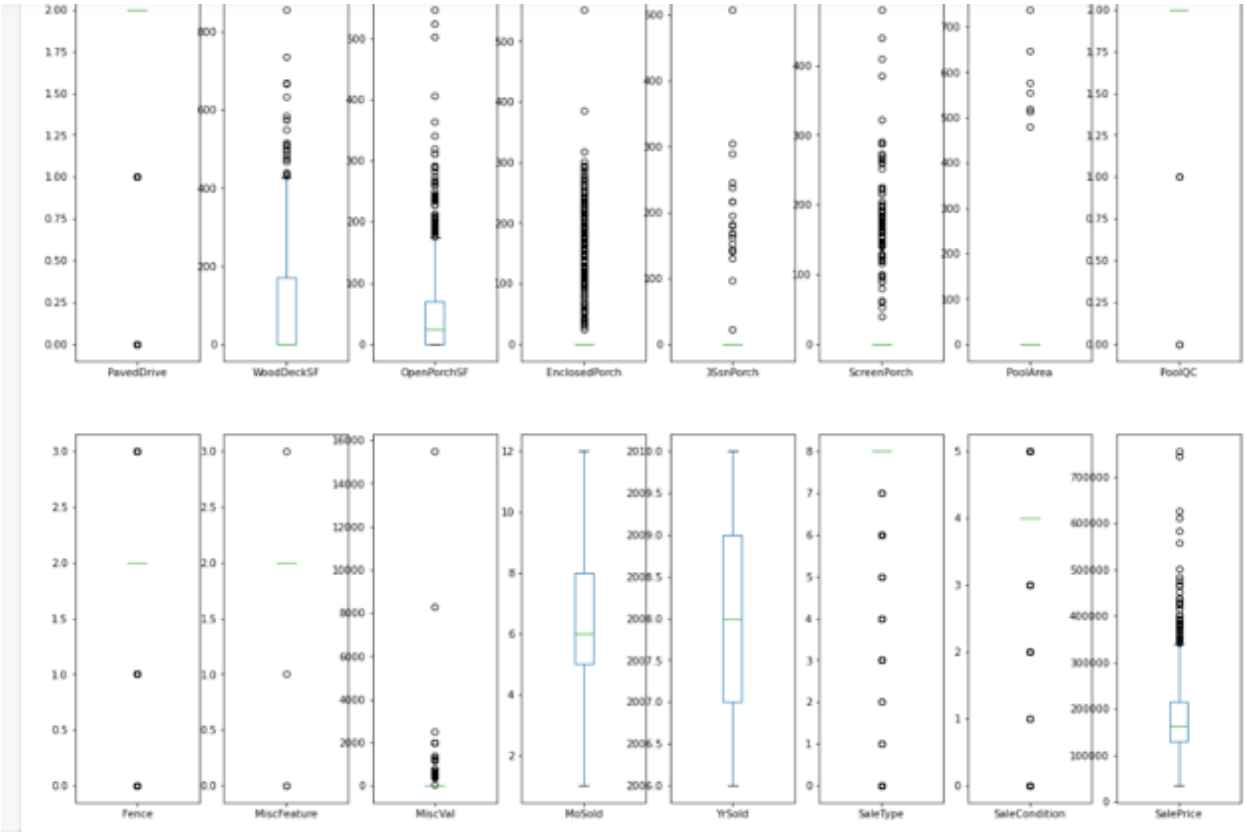
- **Observation of scatter and reg plots :-**

- Variables MiscVal, Enclosedporch have some negative correlation with sale price.
- Variables OpenporchSF, WoodDeckSF, GarageArea, GarageYrBuilt, GrlivArea, 1stflrsf, 2ndflrsf, totalbsmtsf, YearremodAdd, Yearbuilt , lotfrontage,lotarea, masvnrarea, bsmntfinsf1 have positive correlation with column salesprice as we can see in the graphs above
- This gives us the insights of our problem of finding most positive and negative correlated variables with the price.



- After checking the correlation there is not much insight as very less columns are strongly correlated with the output variable which is Salesprice of the houses in the Australian region.

We used box plots to detect outliers and here is a snapshot of boxplots:-

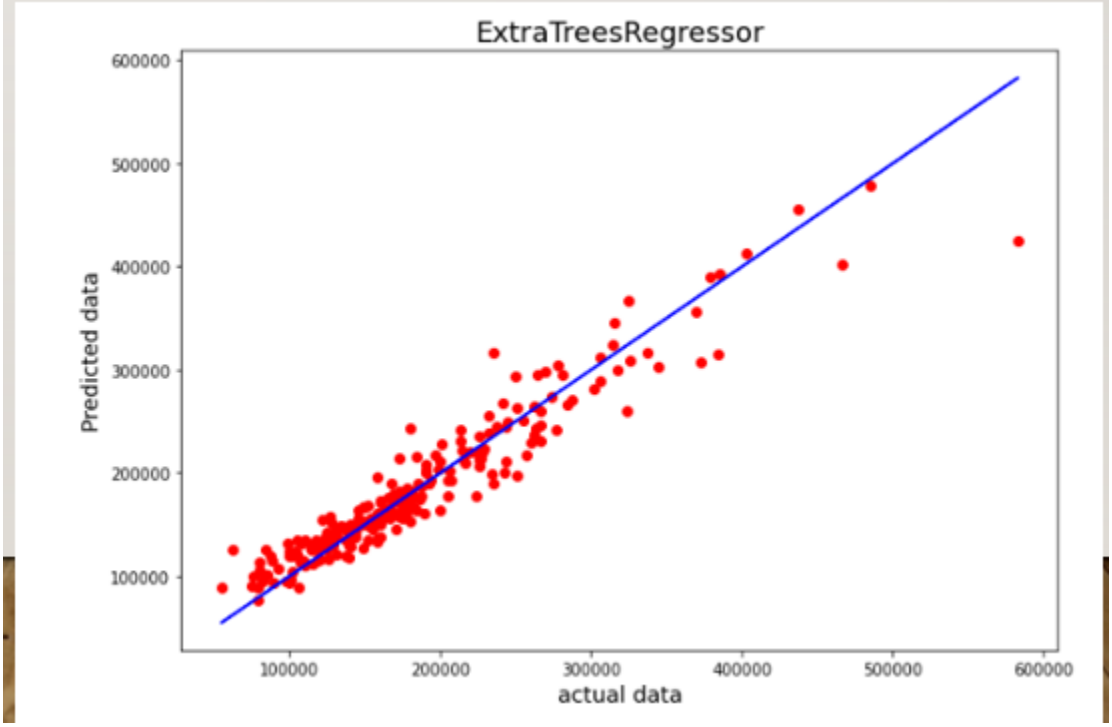


We used distribution plots to check the normal distribution in the variables.

- <u>Interpretation of the Results</u>

1) We saw that the data was not in its finest form we need to clear some nan values and outliers and clean the data.
2) We also saw biasing in the data which we removed using power transform method.
3) From visualization we found the positively correlated variables such as Garagearea and negatively correlated variables such as Enclosedporch with the output variable which is sales price. This will help the surprise housing company to focus on such areas and come up with better strategies and generate profit.

4) We saw that normal sale have the highest counts when it comes to sale condition
5) We found the best peroforming model after checking the metrics such as r2_score, cross val mean score, and mean_squared_error which is Extra Trees Regressor with the accuracy of 91% after hyper tuning.
6) After selecting the models we compared the model's with the predicted values and actual value



We can see that the actual data and predicted data are very close to each other. This indicates that our model is giving good results with accuracy of 91%.

Conclusion

- Key Findings and Conclusions of the Study

We had to perform analysis on the train data and clean the data and build a model which will predict the house prices for the test data.

We did some visualization and pre-processing and after scaling the data we were able to build a model with 91% accuracy.

We used the model to predict house prices for the test data. Here is the image of the results:-

```
: #predicting the test data using ExtraTrees regressor
  df2['saleprice']=etr.predict(df2)
  df2['saleprice']
```

```
: 0      338042.72
  1      262362.55
  2      297387.63
  3      247925.09
  4      306689.73
           ...
  287    290547.50
  288    240060.32
  289    260649.98
  290    271039.73
  291    226478.10
  Name: saleprice, Length: 292, dtype: float64
```

We saved the predicted results in a csv file.

- Learning Outcomes of the Study in respect of Data Science

We found the important features and how do they affect positively and negatively with sale price of the houses and used them to predict the housing prices using machine learning models.

I learned that using data science it can help the business to make better strategies and focus on the areas which affects the output variables and make better decisions.

I also found out that there are so many variables which can affect the house prices which was something new to me. People tend to check all these variables before buying the house and how the most important features can influence buying decision of a human being.

- Limitations of this work and Scope for Future Work

We can see the data was not in its finest form with some more analysis we can still improve the accuracy of the model.

The goal was to achieve 100% accuracy in predicting the house prices but we ended up with the accuracy of 91%.

Due to lack of experience I was not able to reach my goal but with working on more such projects and gaining more experience will help me to grow and develop more valuable skills to reach my goal in future.