

HOUSING: PRICE PREDICTION

Submitted by:

ASHOK KUMAR SHARMA

**ACKNOWLEDGMENT**

I would like to express my deep sense of gratitude to my SME (Subject Matter Expert) **Mr. Shubham Yadav** as well as **Flip Robo Technologies** who gave me the golden opportunity to do this data analysis project on **Housing: Price Prediction**, which also helped me in doing lots of research and I came to know about so many new things.

I am very much thankful to **Dr. Deepika, Trainer (DataTrained)**, for their valuable guidance, keen interest and encouragement at various stages of my training period which eventually helped me a lot in doing this project.

I also acknowledge with thanks for suggestion and timely guidance, which I have received from my SME Mr. Shubham Yadav during this project, which immensely helped me in the evaluation of my ideas on the project.

ASHOK KUMAR SHARMA

**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.

* Conceptual Background of the Domain Problem

Housing, or more generally living spaces, refers to the construction and assigned usage of houses or buildings collectively, for the purpose of sheltering people. Housing is recognized as a social determinant of health. Lack of housing or poor-quality housing can negatively affect an individual's physical and mental health. Housing attributes that negatively affect physical health include dampness, mold, inadequate heating, and overcrowding. Mental health is also affected by inadequate heating, overcrowding, dampness, and mold, as well as lack of personal space.

* Review of Literature

1. What is House Price Prediction?

**House Price Predictions** are beneficial for property investors to know the trend of **housing prices** in a certain location.

* Motivation for the Problem Undertaken

This model will 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.

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem

For checking datatypes and null values, padas.DataFrame.info() method has been used. To change the datatypes pandas.Series.astype() method has been used and to change the null values pandas.Series.replace() method has been used. To get the statistical summary overview, pandas.DataFrame.describe() method has been used to infer the following:

1. Count: to count the number of records.
2. Mean: to calculate the mean of the feature.
3. Std: to calculate the Standard Deviation of the feature.
4. Min: to find the minimum value of the feature.
5. 25% (1st Quartile): to find the first quartile of the feature.
6. 50% (2nd Quartile): to find the median or second quartile.
7. 75% (3rd Quartile): to find the third quartile of the feature.
8. Max: to find the maximum value of the feature.

* Data Sources and their formats

The dataset is being provided by Flib Robo Technologies in .csv (Comma Separated Values) format and contains 1168 records with 81 features as explained below:

1. Id: Identification number for particular record.
2. MSSubClass: Identifies the type of dwelling involved in the sale as follows-
   * **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
3. MSZoning: Identifies the general zoning classification of the sale as follows-
   * **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
4. LotFrontage: Linear feet of street connected to property.
5. LotArea: Lot size in square feet.
6. Street: Type of road access to property as follows-
   * **Grvl:** Gravel
   * **Pave:** Paved
7. Alley: Type of alley access to property as follows-
   * **Grvl:** Gravel
   * **Pave:** Paved
   * **NA:** No alley access
8. LotShape: General shape of property as follows-
   * **Reg:** Regular
   * **IR1:** Slightly irregular
   * **IR2:** Moderately Irregular
   * **IR3:** Irregular
9. LandContour: Flatness of the property as follows-
   * **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
10. Utilities: Type of utilities available as follows-
    * **AllPub:** All public Utilities (E,G,W,& S)
    * **NoSewr:** Electricity, Gas, and Water (Septic Tank)
    * **NoSeWa:** Electricity and Gas Only
    * **ELO:** Electricity only
11. LotConfig: Lot configuration as follows-
    * **Inside:** Inside lot
    * **Corner:** Corner lot
    * **CulDSac:** Cul-de-sac
    * **FR2:** Frontage on 2 sides of property
    * **FR3:** Frontage on 3 sides of property
12. LandSlope: Slope of property as follows-
    * **Gtl:** Gentle slope
    * **Mod:** Moderate Slope
    * **Sev:** Severe Slope
13. 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
14. Condition1: Proximity to various conditions as follows-
    * **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
15. Condition2: Proximity to various conditions (if more than one is present) as follows-
    * **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
16. BldgType: Type of dwelling as follows-
    * **1Fam:** Single-family Detached
    * **2FmCon:** Two-family Conversion; originally built as one-family dwelling
    * **Duplx:** Duplex
    * **TwnhsE:** Townhouse End Unit
    * **TwnhsI:** Townhouse Inside Unit
17. HouseStyle: Style of dwelling as follows-
    * **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
18. OverallQual: Rates the overall material and finish of the house as follows-
    * **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
19. OverallCond: Rates the overall condition of the house as follows-
    * **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
20. YearBuilt: Original construction date
21. YearRemodAdd: Remodel date (same as construction date if no remodeling or additions)
22. RoofStyle: Type of roof as follows-
    * **Flat:** Flat
    * **Gable:** Gable
    * **Gambrel:** Gabrel (Barn)
    * **Hip:** Hip
    * **Mansard:** Mansard
    * **Shed:** Shed
23. RoofMatl: Roof material as follows-
    * **ClyTile:** Clay or Tile
    * **CompShg:** Standard (Composite) Shingle
    * **Membran:** Membrane
    * **Metal:** Metal
    * **Roll:** Roll
    * **Tar&Grv:** Gravel & Tar
    * **WdShake:** Wood Shakes
    * **WdShngl:** Wood Shingles
24. Exterior1st: Exterior covering on house as follows-
    * **AsbShng:** Asbestos Shingles
    * **AsphShn:** Asphalt Shingles
    * **BrkComm:** Brick Common
    * **BrkFace:** Brick 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
25. Exterior2nd: Exterior covering on house (if more than one material) as follows-
    * **AsbShng:** Asbestos Shingles
    * **AsphShn:** Asphalt Shingles
    * **BrkComm:** Brick Common
    * **BrkFace:** Brick 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
26. MasVnrType: Masonry veneer type as follows-
    * **BrkCmn:** Brick Common
    * **BrkFace:** Brick Face
    * **CBlock:** Cinder Block
    * **None:** None
    * **Stone:** Stone
27. MasVnrArea: Masonry veneer area in square feet
28. ExterQual: Evaluates the quality of the material on the exterior as follows-
    * **Ex:** Excellent
    * **Gd:** Good
    * **TA:** Average/Typical
    * **Fa:** Fair
    * **Po:** Poor
29. ExterCond: Evaluates the present condition of the material on the exterior as follows-
    * **Ex:** Excellent
    * **Gd:** Good
    * **TA:** Average/Typical
    * **Fa:** Fair
    * **Po:** Poor
30. Foundation: Type of foundation as follows-
    * **BrkTil:** Brick & Tile
    * **CBlock:** Cinder Block
    * **PConc:** Poured Contrete
    * **Slab:** Slab
    * **Stone:** Stone
    * **Wood:** Wood
31. BsmtQual: Evaluates the height of the basement as follows-
    * **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
32. BsmtCond: Evaluates the general condition of the basement as foloows-
    * **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
33. BsmtExposure: Refers to walkout or garden level walls as follows-
    * **Gd:** Good Exposure
    * **Av:** Average Exposure (split levels or foyers typically score average or above)
    * **Mn:** Mimimum Exposure
    * **No:** No Exposure
    * **NA:** No Basement
34. BsmtFinType1: Rating of basement finished area as follows-
    * **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
35. BsmtFinSF1: Type 1 finished square feet
36. BsmtFinType2: Rating of basement finished area (if multiple types) as follows-
    * **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
37. BsmtFinSF2: Type 2 finished square feet
38. BsmtUnfSF: Unfinished square feet of basement area
39. TotalBsmtSF: Total square feet of basement area
40. Heating: Type of heating as follows-
    * **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
41. HeatingQC: Heating quality and condition as follows-
    * **Ex:** Excellent
    * **Gd:** Good
    * **TA:** Average/Typical
    * **Fa:** Fair
    * **Po:** Poor
42. CentralAir: Central air conditioning as follows-
    * **N:** No
    * **Y:** Yes
43. Electrical: Electrical system as follows-
    * **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
44. 1stFlrSF: First Floor square feet
45. 2ndFlrSF: Second floor square feet
46. LowQualFinSF: Low quality finished square feet (all floors)
47. GrLivArea: Above grade (ground) living area square feet
48. BsmtFullBath: Basement full bathrooms
49. BsmtHalfBath: Basement half bathrooms
50. FullBath: Full bathrooms above grade
51. HalfBath: Half baths above grade
52. BedroomAbvGr: Bedrooms above grade (does NOT include basement bedrooms)
53. KitchenAbvGr: Kitchens above grade
54. KitchenQual: Kitchen quality as follows-
    * **Ex:** Excellent
    * **Gd:** Good
    * **TA:** Typical/Average
    * **Fa:** Fair
    * **Po:** Poor
55. TotRmsAbvGrd: Total rooms above grade (does not include bathrooms)
56. Functional: Home functionality (Assume typical unless deductions are warranted) as follows-
    * **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
57. Fireplaces: Number of fireplaces
58. FireplaceQu: Fireplace quality as follows-
    * **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
59. GarageType: Garage location as follows-
    * **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
60. GarageYrBlt: Year garage was built
61. GarageFinish: Interior finish of the garage as follows-
    * **Fin:** Finished
    * **RFn:** Rough Finished
    * **Unf:** Unfinished
    * **NA:** No Garage
62. GarageCars: Size of garage in car capacity
63. GarageArea: Size of garage in square feet
64. GarageQual: Garage quality as follows-
    * **Ex:** Excellent
    * **Gd:** Good
    * **TA:** Typical/Average
    * **Fa:** Fair
    * **Po:** Poor
    * **NA:** No Garage
65. GarageCond: Garage condition as follows-
    * **Ex:** Excellent
    * **Gd:** Good
    * **TA:** Typical/Average
    * **Fa:** Fair
    * **Po:** Poor
    * **NA:** No Garage
66. PavedDrive: Paved driveway as follows-
    * **Y:** Paved
    * **P:** Partial Pavement
    * **N:** Dirt/Gravel
67. WoodDeckSF: Wood deck area in square feet
68. OpenPorchSF: Open porch area in square feet
69. EnclosedPorch: Enclosed porch area in square feet
70. 3SsnPorch: Three season porch area in square feet
71. ScreenPorch: Screen porch area in square feet
72. PoolArea: Pool area in square feet
73. PoolQC: Pool quality as follows-
    * **Ex:** Excellent
    * **Gd:** Good
    * **TA:** Average/Typical
    * **Fa:** Fair
    * **NA:** No Pool
74. Fence: Fence quality as follows-
    * **GdPrv:** Good Privacy
    * **MnPrv:** Minimum Privacy
    * **GdWo:** Good Wood
    * **MnWw:** Minimum Wood/Wire
    * **NA:** No Fence
75. MiscFeature: Miscellaneous feature not covered in other categories as follows-
    * **Elev:** Elevator
    * **Gar2:** 2nd Garage (if not described in garage section)
    * **Othr:** Other
    * **Shed:** Shed (over 100 SF)
    * **TenC:** Tennis Court
    * **NA:** None
76. MiscVal: Value of miscellaneous feature
77. MoSold: Month Sold (MM)
78. YrSold: Year Sold (YYYY)
79. SaleType: Type of sale as follows-
    * **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
80. SaleCondition: Condition of sale as follows-
    * **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)
81. SalePrice: Sale price of the house.

* Data Preprocessing Done

The following pre-processing pipeline is required to perform model prediction:

* **Load Dataset**
* **Treat Data Types** of Features: 'LotArea','BsmtFinSF1','BsmtFinSF2','BsmtUnfSF','TotalBsmtSF','1stFlrSF','2ndFlrSF','LowQualFinSF', 'GrLivArea','GarageArea','WoodDeckSF','OpenPorchSF','EnclosedPorch','3SsnPorch','ScreenPorch', 'PoolArea' (from int to float)

'MSSubClass','OverallQual','OverallCond' (from int to object)

* **Drop features Id, Alley, FireplaceQu, PoolQC, Fence, MiscFeature, Utilities**
* **Treat null values** in **continous features with mean value** except feature YearBuilt and YearRemodAdd which needs to be treated with median value. Also, for **categorical features, treat with mode value**.
* **Encode categorical features** using OrdinalEncoder.
* **Remove Outliers** using zscore from scipy.stats.
* **Treat skewness of continuous data** (with threshold value -1 to +1) using power\_transform function from sklearn.preprocessing.
* **Load serialized model and make prediction** for test data.
* Data Inputs- Logic- Output Relationships

|  |  |  |
| --- | --- | --- |
| Input | Logic (algorithm) | Output |
| 0 MSSubClass object  1 MSZoning object  2 LotFrontage float64  3 LotArea float64  4 Street object  5 LotShape object  6 LandContour object  7 LotConfig object  8 LandSlope object  9 Neighborhood object  10 Condition1 object  11 Condition2 object  12 BldgType object  13 HouseStyle object  14 OverallQual object  15 OverallCond object  16 YearBuilt int64  17 YearRemodAdd int64  18 RoofStyle object  19 RoofMatl object  20 Exterior1st object  21 Exterior2nd object  22 MasVnrType object  23 MasVnrArea float64  24 ExterQual object  25 ExterCond object  26 Foundation object  27 BsmtQual object  28 BsmtCond object  29 BsmtExposure object  30 BsmtFinType1 object  31 BsmtFinSF1 float64  32 BsmtFinType2 object  33 BsmtFinSF2 float64  34 BsmtUnfSF float64  35 TotalBsmtSF float64  36 Heating object  37 HeatingQC object  38 CentralAir object  39 Electrical object  40 1stFlrSF float64  41 2ndFlrSF float64  42 LowQualFinSF float64  43 GrLivArea float64  44 BsmtFullBath int64  45 BsmtHalfBath int64  46 FullBath int64  47 HalfBath int64  48 BedroomAbvGr int64  49 KitchenAbvGr int64  50 KitchenQual object  51 TotRmsAbvGrd int64  52 Functional object  53 Fireplaces int64  54 GarageType object  55 GarageYrBlt float64  56 GarageFinish object  57 GarageCars int64  58 GarageArea float64  59 GarageQual object  60 GarageCond object  61 PavedDrive object  62 WoodDeckSF float64  63 OpenPorchSF float64  64 EnclosedPorch float64  65 3SsnPorch float64  66 ScreenPorch float64  67 PoolArea float64  68 MiscVal int64  69 MoSold int64  70 YrSold int64  71 SaleType object  72 SaleCondition object | LinearRegression  Lasso  Ridge  AdaBoostRegressor | SalePrice |

There are 73 input variables needs to be provided to the logic to get the output i.e. SalePrice. Logic highlighted in yellow i.e. AdaBoostRegressor is the best performing algorithm among all other logics on this dataset.

* Hardware and Software Requirements and Tools Used

During this project, following set of hardware is being used:

RAM: 8 GB

CPU: AMD A8 Quad Core 2.2 Ghz

GPU: AMD Redon R5 Graphics

and the following software and tools is being used:

* 1. Python
  2. Jupyter Notebook
  3. Anaconda

With following libraries and packages:

* Pandas
* Numpy
* Matplotlib
* Seaborn
* Scipy.stats
* sklearn

**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

To solve this problem following steps are used:

1. Load Dataset using pandas
2. Check of datatypes and correct them as per requirements.
3. Check for any null or irrelevant values and treat them accordingly.
4. Check the statistical summary for mean, minimum, maximum, median, standard deviation, skewness etc.
5. Remove outliers considering max allowed data loss 5-7%.
6. Scale the data for model training
7. Train & Test the model.

* Testing of Identified Approaches (Algorithms)

Following are the list of algorithms used for training and testing:

1. LinearRegression
2. Lasso
3. Ridge
4. AdaBoosRegressor

* Run and Evaluate selected models

A total of 4 algorithm has been used on this dataset for training testing purpose, these are LinearRegression, Lasso, Ridge and AdaBoostRegressor. To perform training and testing operation(s) following functions has been defined for which codes are as follows:

#importing required libraries

from sklearn.model\_selection import train\_test\_split, GridSearchCV, cross\_val\_score

from sklearn.metrics import r2\_score, mean\_squared\_error, mean\_absolute\_error

#Supressing simple warnings

import warnings

warnings.simplefilter('ignore')

import timeit

#Defining function for getting best random\_state

def get\_random\_state(model,x,y,r=range(0,200,50),test\_size=0.25):

r\_state = -999

a\_score = -999

for i in r:

x\_train,x\_test,y\_train,y\_test = train\_test\_split(x,y,test\_size=test\_size,random\_state=i)

model.fit(x\_train,y\_train)

predict\_y = model.predict(x\_test)

temp\_aScore = r2\_score(y\_test,predict\_y)

print(f"\t\t{i}:{temp\_aScore}")

if temp\_aScore>a\_score:

a\_score = temp\_aScore

r\_state = i

return r\_state, a\_score

#Defining function for getting best CV

def get\_cv(model,parameters,x\_train,y\_train,r=range(2,20,4)):

best\_cv = -999

best\_cvScore = -999

for i in r:

gscv = GridSearchCV(model,parameters)

gscv.fit(x\_train,y\_train)

temp\_cvScore = cross\_val\_score(gscv.best\_estimator\_,x\_train,y\_train,cv=i).mean()

if temp\_cvScore > best\_cvScore:

best\_cvScore = temp\_cvScore

best\_cv = i

return best\_cv, best\_cvScore

#Defining function for model training & testing

def train\_models(models,x,y,r\_range=range(0,200,50),t\_size=0.25,cv\_range=range(2,20,4)):

for i in models:

print(f"Processing: {i}...")

start\_time = timeit.default\_timer()

#Finding best random\_state for train\_test\_split

print(f"\tFinding best random\_state...")

best\_random\_state, best\_r2\_score = get\_random\_state(models[i]['name'],x,y,r\_range,t\_size)

print(f"\trandom\_state: {best\_random\_state}")

#Splitting train test data with best random\_state

print(f"\tSpliting train test data...")

x\_train,x\_test,y\_train,y\_test = train\_test\_split(x,y,test\_size=t\_size,random\_state=best\_random\_state)

#Hypertuning Parameters: Finding best CV

print(f"\tFinding best CV by hypertuning...")

best\_cv, best\_cvScore = get\_cv(models[i]['name'],models[i]['parameters'],x\_train,y\_train,cv\_range)

#Training Final model with hypertuned parameters

print(f"\tTraining with hypertuned parameters...")

gscv = GridSearchCV(models[i]['name'],models[i]['parameters'],cv=best\_cv)

gscv.fit(x\_train,y\_train)

#Checking final performance of the trained model

predict\_y = gscv.best\_estimator\_.predict(x\_test)

aScore = r2\_score(y\_test,predict\_y)

mse = mean\_squared\_error(y\_test,predict\_y)

mae = mean\_absolute\_error(y\_test,predict\_y)

end\_time = timeit.default\_timer()

#Storing model specs

models[i]['initial\_r2\_score'] = best\_r2\_score

models[i]['r2\_score'] = aScore

models[i]['mse'] = mse

models[i]['rmse'] = np.sqrt(mse)

models[i]['mae'] = mae

models[i]['random\_state'] = best\_random\_state

models[i]['x\_train'] = x\_train

models[i]['x\_test'] = x\_test

models[i]['y\_train'] = y\_train

models[i]['y\_test'] = y\_test

models[i]['cv'] = best\_cv

models[i]['cross\_val\_score'] = best\_cvScore

models[i]['gscv'] = gscv

models[i]['predict\_y'] = predict\_y

models[i]['build\_time'] = end\_time-start\_time

print(f"\tCompleted in {end\_time-start\_time}sec...\n\n")

return models

#Defining function for displaying model performance

def display\_performance(models):

#Displaying model performance and comparing it to select best model.

model\_names = []

model\_init\_aScores = []

model\_aScores = []

model\_cvScores = []

model\_bTimes = []

for i in models:

model = models[i]

print(f"START: {i}===================\n")

print(f"\tBest random\_state: {model['random\_state']} with best r2\_score: {model['initial\_r2\_score']}\n")

print(f"\tBest CV: {model['cv']} with best cross\_val\_score: {model['cross\_val\_score']}\n")

print(f"\tBest Parameters: {model['gscv'].best\_params\_}\n\n")

print(f"----Final Performance----")

print(f"R2 Score: {round(model['r2\_score']\*100,2)}%\n")

print(f"MSE:\n{model['mse']}\n")

print(f"RMSE:\n{model['rmse']}\n")

print(f"MAE:\n{model['mae']}\n")

print(f"Total Build Time: {model['build\_time']}sec")

print(f"END: {i}======================\n\n\n")

model\_names.append(i)

model\_init\_aScores.append(model['initial\_r2\_score'])

model\_aScores.append(model['r2\_score'])

model\_cvScores.append(model['cross\_val\_score'])

model\_bTimes.append(model['build\_time'])

df\_cmp = pd.DataFrame({"Name":model\_names,"initial\_r2\_score":model\_init\_aScores,"final\_r2\_score":model\_aScores,"cross\_val\_score":model\_cvScores})

df\_cmp['Difference(final\_r2\_score-cross\_val\_score)'] = df\_cmp['final\_r2\_score']-df\_cmp['cross\_val\_score']

df\_cmp['build\_time(in seconds)'] = model\_bTimes

return df\_cmp

#Preparing List of Models and Testing them to get best model.

from sklearn.linear\_model import LinearRegression, Lasso, Ridge

from sklearn.ensemble import AdaBoostRegressor

#List of Models with parameters

models = {

"LinearRegression":{

"name": LinearRegression(),

"parameters":{

"fit\_intercept": [True,False],

"normalize": [True,False]

}

},

"Lasso":{

"name": Lasso(),

"parameters":{

"alpha": [0.0001,0.001],

"fit\_intercept": [True,False],

"selection": ['cyclic','random']

}

},

"Ridge":{

"name": Ridge(),

"parameters":{

"alpha": [0.0001,0.001],

"fit\_intercept": [True,False],

"solver": ['lsqr','saga']

}

},

"AdaBoostRegressor":{

"name": AdaBoostRegressor(),

"parameters":{

"n\_estimators": [50,100,150]

}

},

}

#Training models

build\_models = train\_models(models,scaled\_x,Y,r\_range=[50,52,54,56],cv\_range=[5,10])

Processing: LinearRegression...

Finding best random\_state...

50:0.8352041304434419

52:0.8151560036926654

54:0.7273902256286164

56:0.7840643296549948

random\_state: 50

Spliting train test data...

Finding best CV by hypertuning...

Training with hypertuned parameters...

Completed in 3.657814099999996sec...

Processing: Lasso...

Finding best random\_state...

50:0.8353215617629535

52:0.8152542738972486

54:0.7275233736366427

56:0.784148232883415

random\_state: 50

Spliting train test data...

Finding best CV by hypertuning...

Training with hypertuned parameters...

Completed in 26.3009969sec...

Processing: Ridge...

Finding best random\_state...

50:0.8376541467623737

52:0.8172095761709339

54:0.7303521809287534

56:0.7859521503199864

random\_state: 50

Spliting train test data...

Finding best CV by hypertuning...

Training with hypertuned parameters...

Completed in 99.20021009999996sec...

Processing: AdaBoostRegressor...

Finding best random\_state...

50:0.8051387225758828

52:0.812375154713809

54:0.7081514757584519

56:0.7887810749774529

random\_state: 52

Spliting train test data...

Finding best CV by hypertuning...

Training with hypertuned parameters...

Completed in 77.1145406sec...

#Displaying model performances

display\_performance(build\_models)

START: LinearRegression===================

Best random\_state: 50 with best r2\_score: 0.8352041304434419

Best CV: 5 with best cross\_val\_score: 0.7205653405098646

Best Parameters: {'fit\_intercept': False, 'normalize': True}

----Final Performance----

R2 Score: 82.02%

MSE:

1013880568.3484807

RMSE:

31841.49130220632

MAE:

22443.97646492671

Total Build Time: 3.657814099999996sec

END: LinearRegression======================

START: Lasso===================

Best random\_state: 50 with best r2\_score: 0.8353215617629535

Best CV: 5 with best cross\_val\_score: 0.7217447014865561

Best Parameters: {'alpha': 0.001, 'fit\_intercept': False, 'selection': 'random'}

----Final Performance----

R2 Score: 82.11%

MSE:

1008603279.7333405

RMSE:

31758.515074438546

MAE:

22429.112437176867

Total Build Time: 26.3009969sec

END: Lasso======================

START: Ridge===================

Best random\_state: 50 with best r2\_score: 0.8376541467623737

Best CV: 5 with best cross\_val\_score: 0.7228589396281201

Best Parameters: {'alpha': 0.001, 'fit\_intercept': False, 'solver': 'saga'}

----Final Performance----

R2 Score: 82.69%

MSE:

976109302.8095839

RMSE:

31242.748003490087

MAE:

22393.89334531166

Total Build Time: 99.20021009999996sec

END: Ridge======================

START: AdaBoostRegressor===================

Best random\_state: 52 with best r2\_score: 0.812375154713809

Best CV: 10 with best cross\_val\_score: 0.7752609629255278

Best Parameters: {'n\_estimators': 150}

----Final Performance----

R2 Score: 80.84%

MSE:

1096925517.4501777

RMSE:

33119.865903263824

MAE:

24113.36686757997

Total Build Time: 77.1145406sec

END: AdaBoostRegressor======================

Out[87]:

|  | **Name** | **initial\_r2\_score** | **final\_r2\_score** | **cross\_val\_score** | **Difference(final\_r2\_score-cross\_val\_score)** | **build\_time(in seconds)** |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | LinearRegression | 0.835204 | 0.820169 | 0.720565 | 0.099603 | 3.657814 |
| **1** | Lasso | 0.835322 | 0.821105 | 0.721745 | 0.099360 | 26.300997 |
| **2** | Ridge | 0.837654 | 0.826868 | 0.722859 | 0.104009 | 99.200210 |
| **3** | AdaBoostRegressor | 0.812375 | 0.808377 | 0.775261 | 0.033116 | 77.114541 |

From the above model performance comparison it is clear that **AdaBoostRegressor** out-performs the other models with **r2\_score of 80.83%** and **lowest difference between r2\_score and cross\_val\_score**. Therefore, continuing with **AdaBoostRegressor** as final model.

* Key Metrics for success in solving problem under consideration

To find out best performing model following metrices are used:

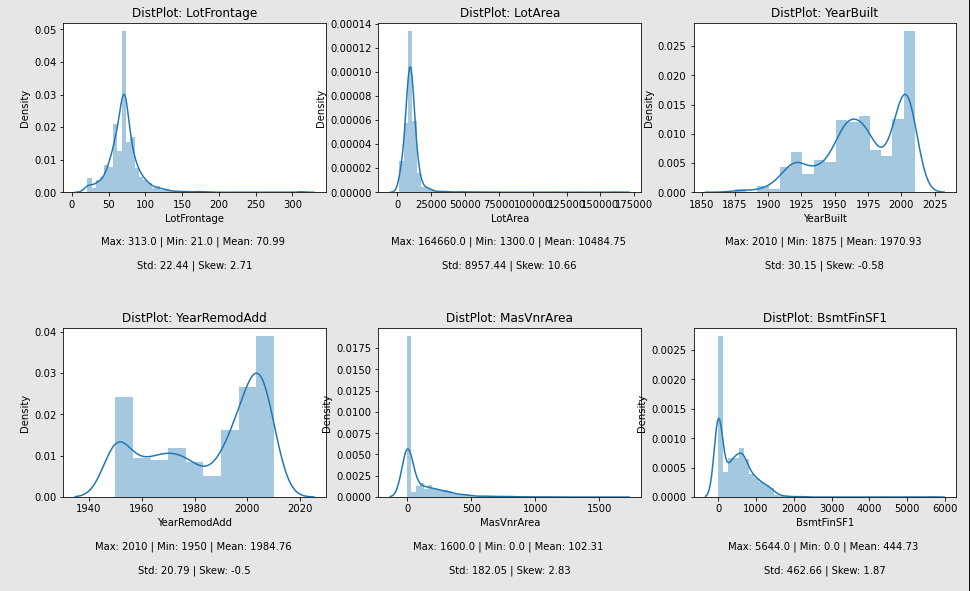
1. R2 Score: It is used to check the model performance score between 0.0 to 1.0
2. Mean Squared Error: The mean squared error of a model with respect to a test set is the mean of the squared prediction errors over all instances in the test set.
3. Root Mean Squared Error: It is the root of mean squared error.
4. Mean Absolute Error: It is the average of absolute errors for a group of predictions and observations as a measurement of the magnitude of errors for the entire group.

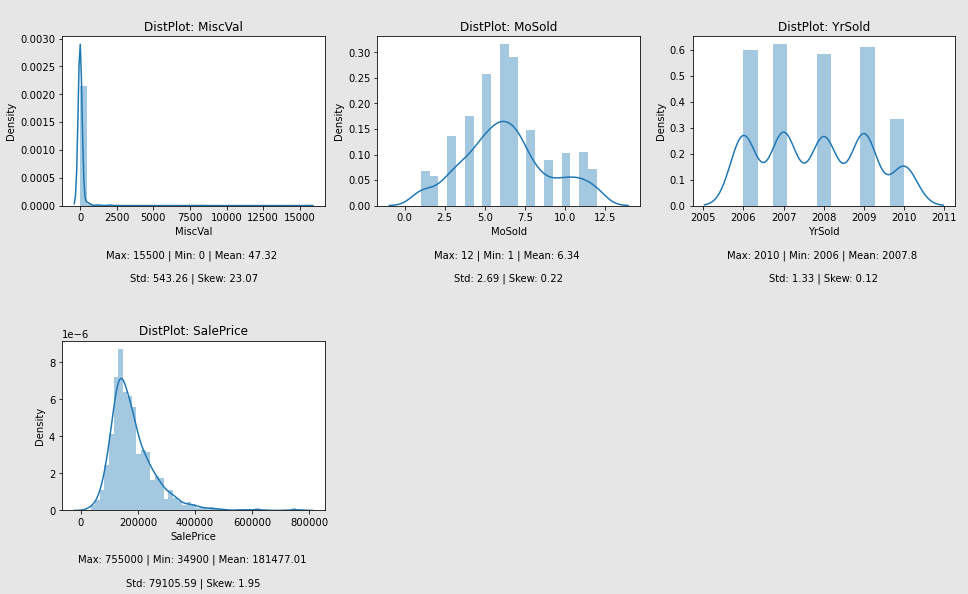
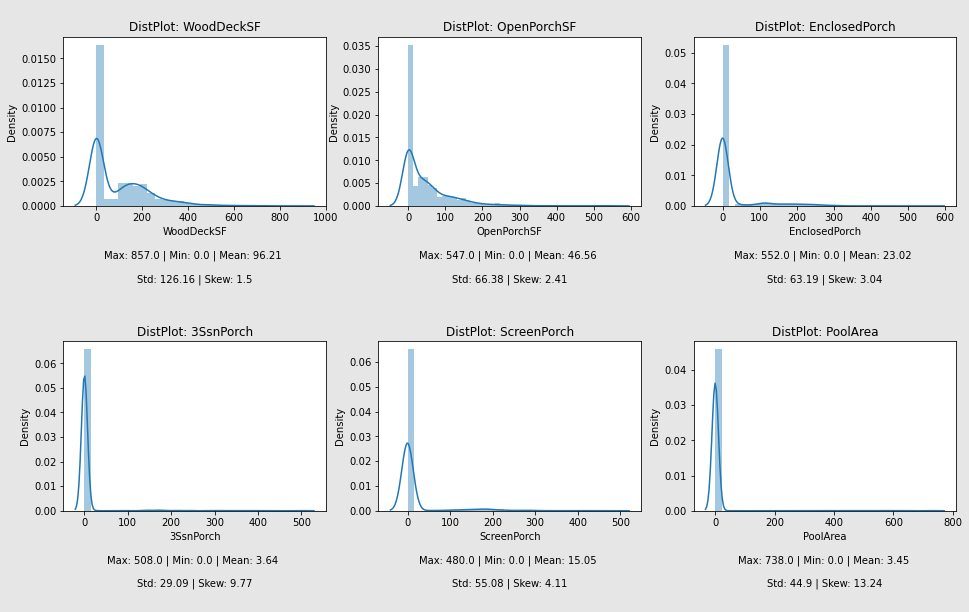
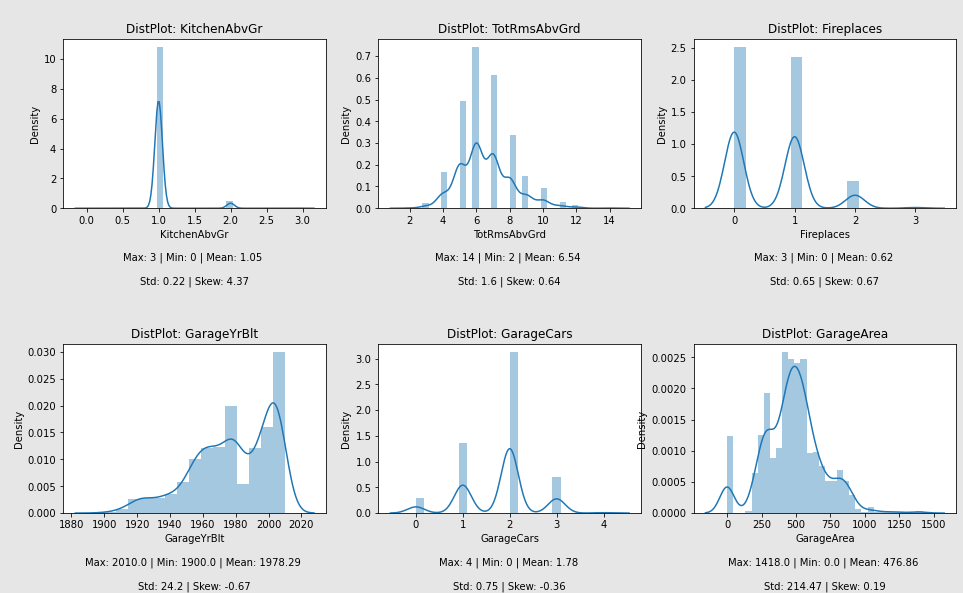
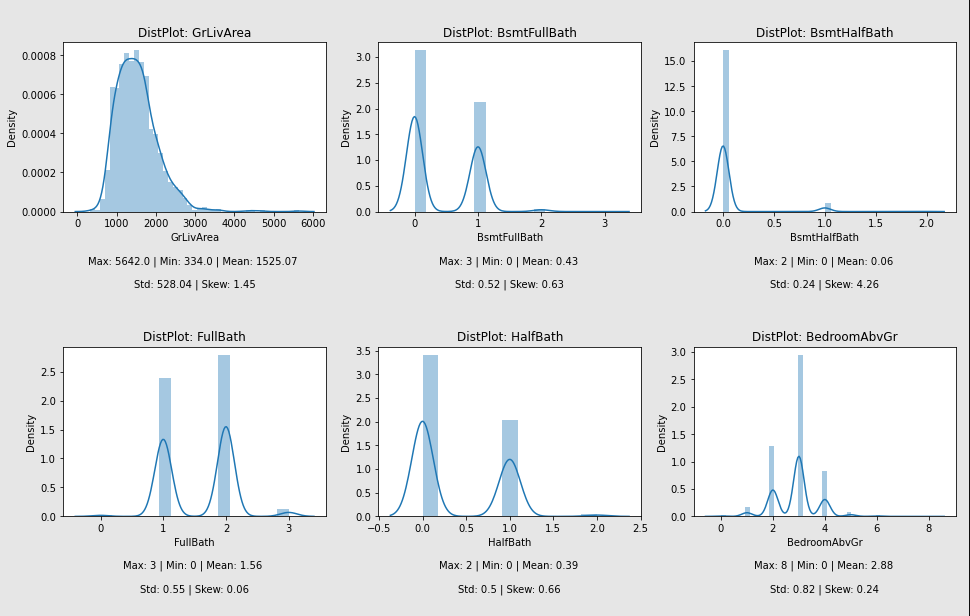
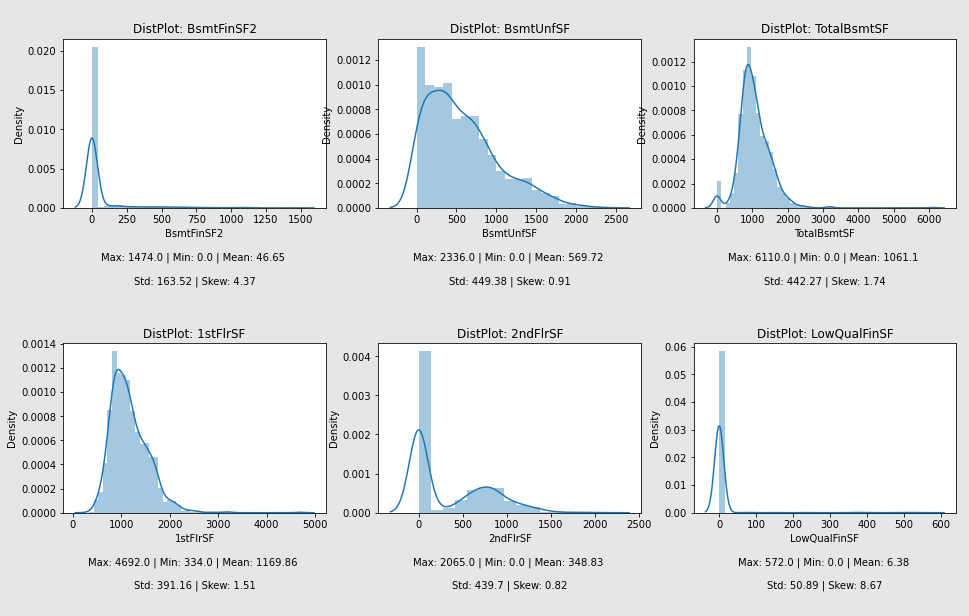
* Visualizations

To better understand the data, following types of visualizations have been used: 1. Univariate, 2. Bivariate and 3. Multivariate.

1. Univariate Analysis: Univariate analysis is the simplest form of data analysis where the data being analysed contains only one variable. In this project, distribution plot, count plot and box plot has been used.

Distribution Plot (distplot):





**Remarks:**

**for feature LotFrontage:**

* Data is not distributed normally or in bell curve.
* Data ranges from 21 to 313 with mean value 70.99
* Data is sort of positively skewed.

**for feature LotArea:**

* Data is not distributed normally or in bell curve.
* Data ranges from 1300 to 164660 with mean value 10484.75
* Data is **highly positively skewed and needs to be treated accordingly**.

**for feature YearBuilt:**

* Data is not distributed normally or in bell curve.
* Data ranges from 1875 to 2010 with mean value 1971
* Data is sort of negatively skewed.

**for feature YearRemodAdd:**

* Data is not distributed normally or in bell curve.
* Data ranges from 1950 to 2010 with mean value 1985
* Data is sort of negatively skewed.

**for feature MassVnrArea:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 1600 with mean value 102.31
* Data is sort of positively skewed.
* Data is spreaded.

**for feature BsmtFinSF1:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 5644 with mean value 444.73
* Data is sort of positively skewed.
* Data is spreaded.

**for feature BsmtFinSF2:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 1474 with mean value 46.65
* Data is sort of positively skewed.
* Data is spreaded.

**for feature BsmtUnfSF:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 2336 with mean value 569.72
* Data is sort of positively skewed.
* Data is spreaded.

**for feature TotalBsmtSF:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 6110 with mean value 1061.1
* Data is sort of positively skewed.
* Data is spreaded.

**for feature 1stFlrSF:**

* Data is not distributed normally or in bell curve.
* Data ranges from 334 to 4692 with mean value 1169.86
* Data is sort of positively skewed.
* Data is spreaded.

**for feature 2ndFlrSF:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 2065 with mean value 348.83
* Data is sort of positively skewed.
* Data is spreaded.

**for feature LowQualFinSF:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 572 with mean value 6.38
* Data is **highly positively skewed and needs to be treated accordingly**.

**for feature GrLivArea:**

* Data is not distributed normally or in bell curve.
* Data ranges from 334 to 5642 with mean value 1525.07
* Data is sort of positively skewed.
* Data is spreaded.

**for feature BsmtFullBath:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 3 with mean value 0.43

**for feature BsmtHalfBath:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 2 with mean value 0.06
* Data is positively skewed.

**for feature FullBath:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 3 with mean value 1.56

**for feature HalfBath:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 2 with mean value 0.39

**for feature BedroomAbvGr:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 8 with mean value 2.88

**for feature KitchenAbvGr:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 3 with mean value 1.05

**for feature TotRmsAbvGrd:**

* Data is not distributed normally or in bell curve.
* Data ranges from 2 to 14 with mean value 6.54

**for feature Fireplaces:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 3 with mean value 0.62

**for feature GarageYrBlt:**

* Data is not distributed normally or in bell curve.
* Data ranges from 1900 to 2010 with mean value 1978

**for feature GarageCars:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 4 with mean value 1.78

**for feature GarageArea:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 1418 with mean value 776.86

**for feature WoodDeckSF:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 857 with mean value 96.21
* Data is spreaded.

**for feature OpenPorchSF:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 547 with mean value 46.56
* Data is spreaded.

**for feature EnclosedPorch:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 552 with mean value 23.02

**for feature 3SsnPorch:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 508 with mean value 3.64

**for feature ScreenPorch:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 480 with mean value 15.05
* Data is positively high skewed.

**for feature PoolArea:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 738 with mean value 345

**for feature MiscVal:**

* Data is not distributed normally or in bell curve.
* Data ranges from 0 to 15500 with mean value 47.32
* Data is positively high skewed.

**for feature MoSold:**

* Data is not distributed normally or in bell curve.
* Data ranges from 1 to 12 with mean value 6.34

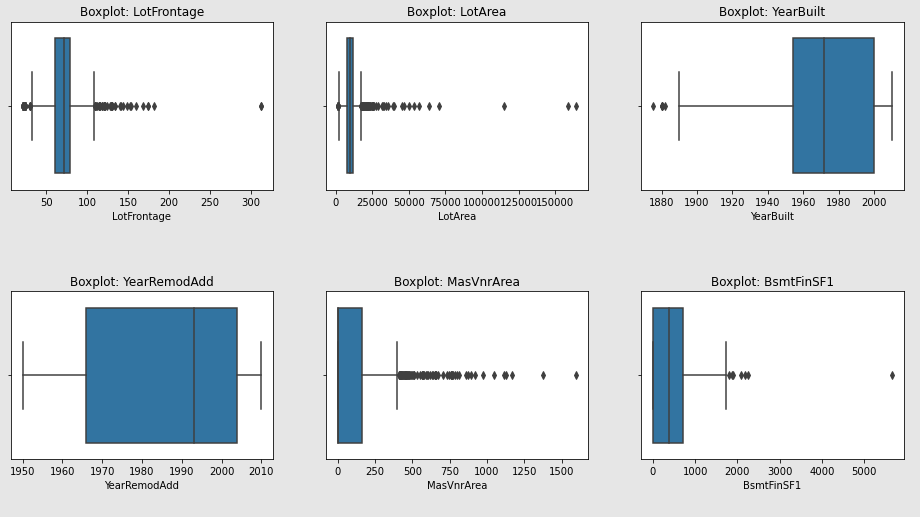
**for feature YrSold:**

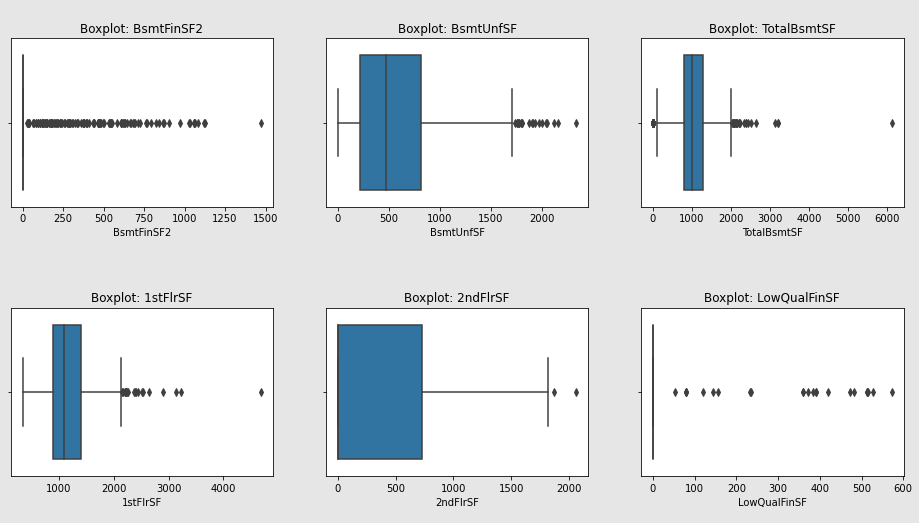
* Data is not distributed normally or in bell curve.
* Data ranges from 2006 to 2010 with mean value 2007

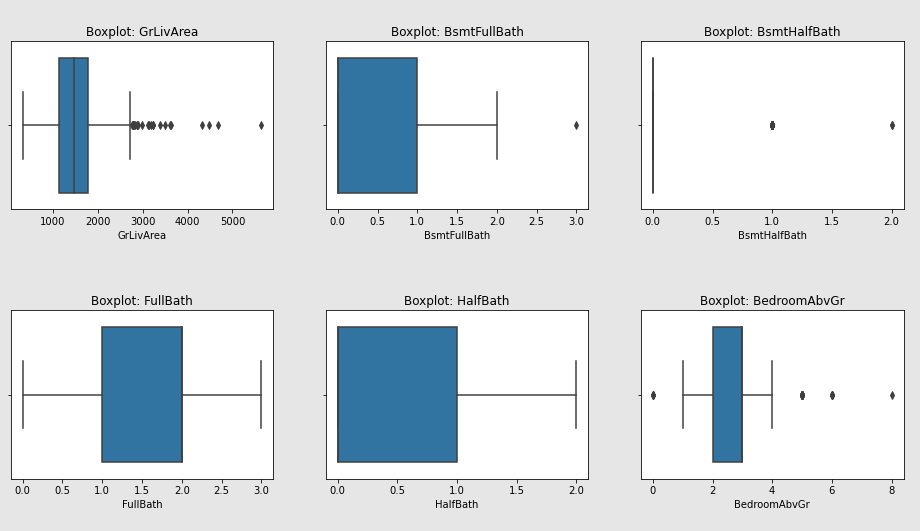
**for feature SalePrice:**

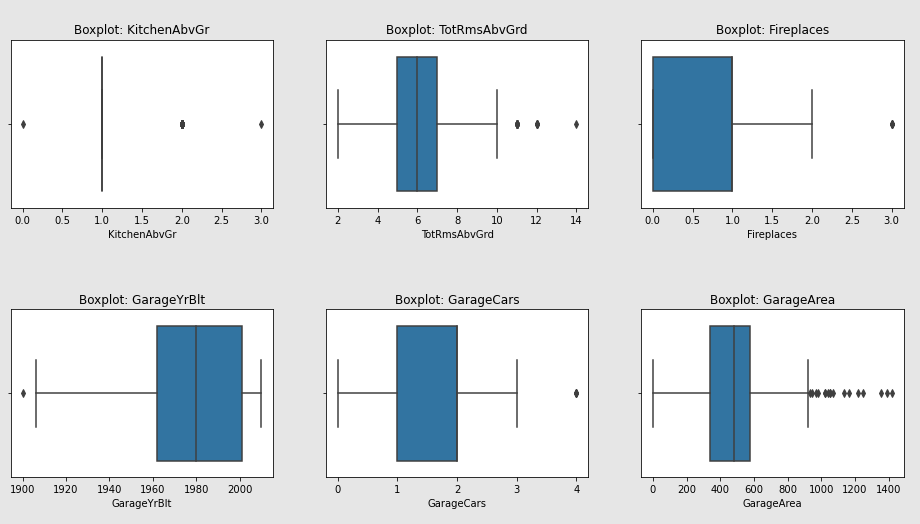
* Data is somewhat distributed normally but not in bell curve.
* Data ranges from 34900 to 755000 with mean value 181477.01

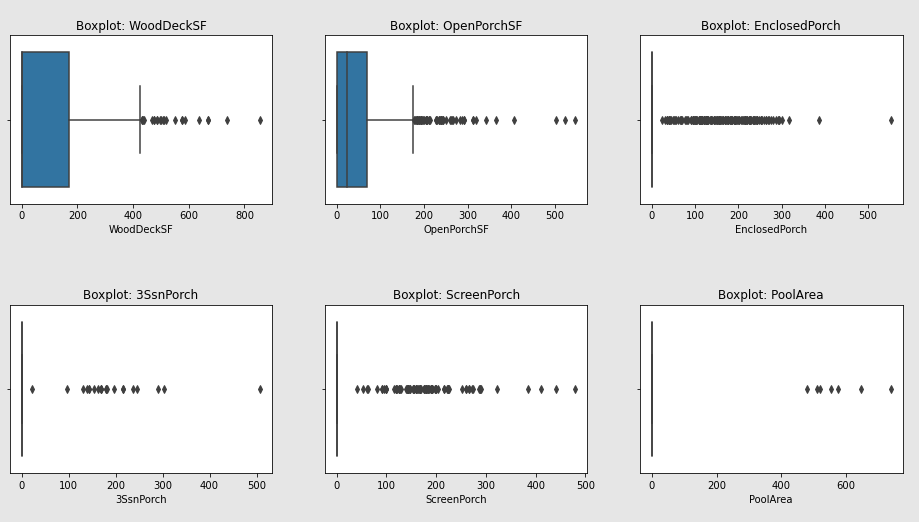
Next is boxplot which used to detect the presence of outliers in the data:

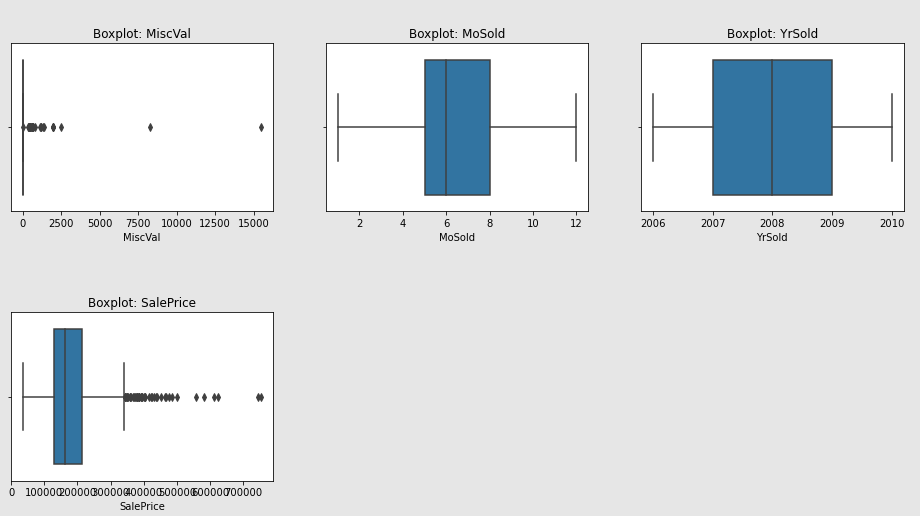








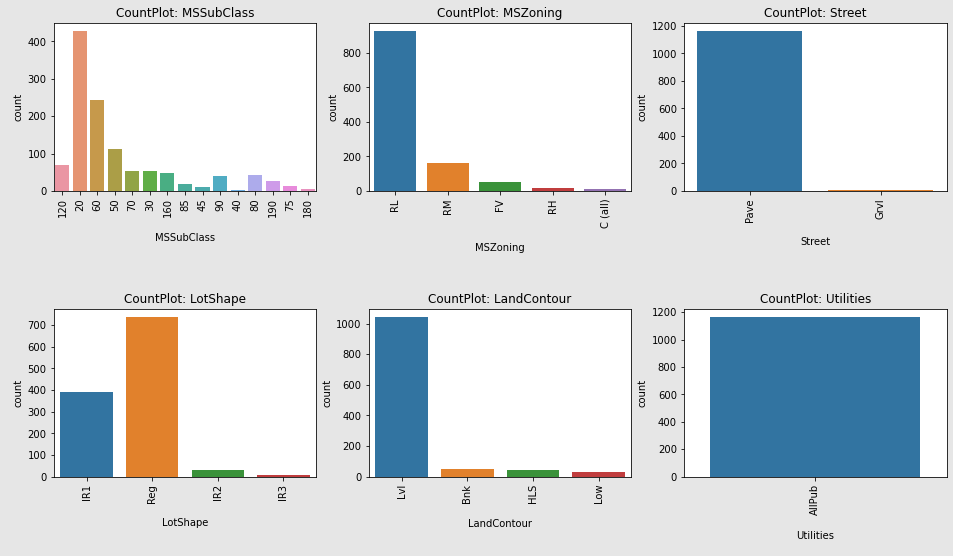


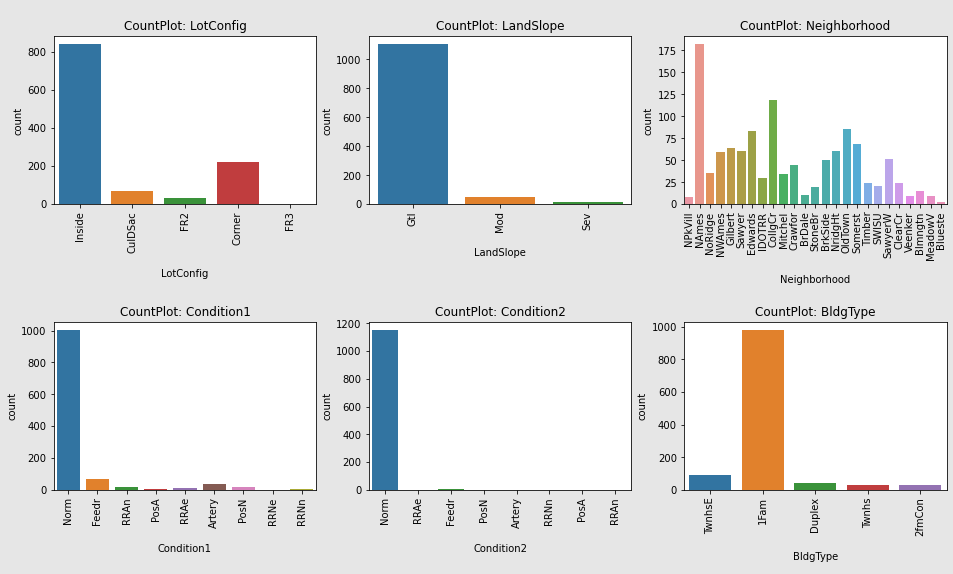


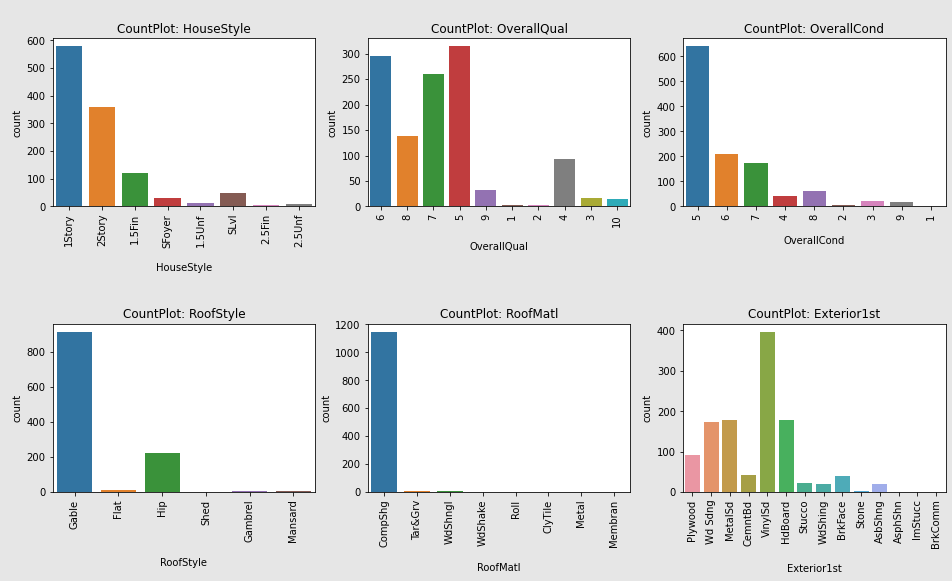
#### Remarks:

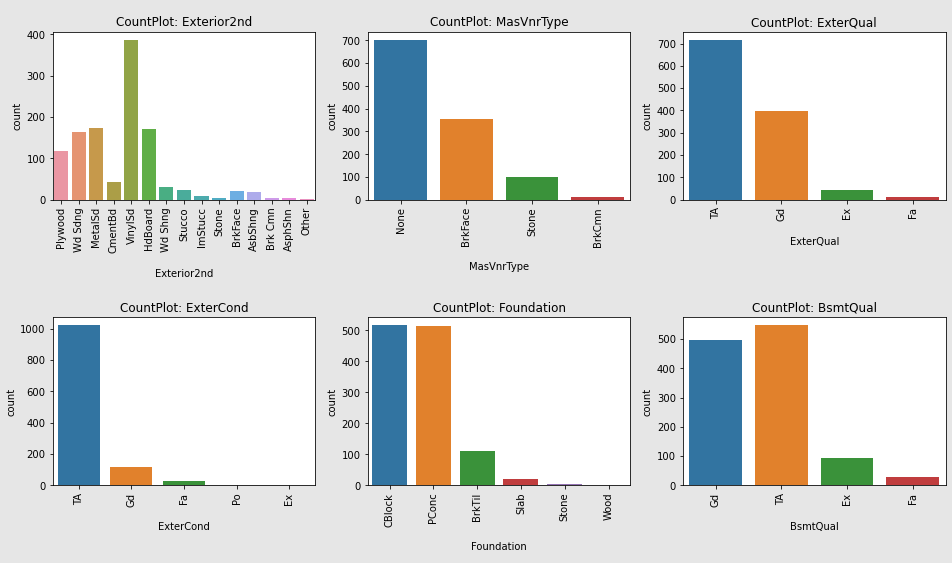
* Except features YearBuilt, YearRemodAdd, 2ndFlrSF, BsmtFullBath, BsmtHalfBath, FullBath, HalfBath, BedroomAbvGr, KitchenAbvGr, TotRmsAbvGrd, Fireplaces, GarageYrBlt, GarageCars, MoSold and YrSold, boxplot shows **some or more outliers might be present in other features**.

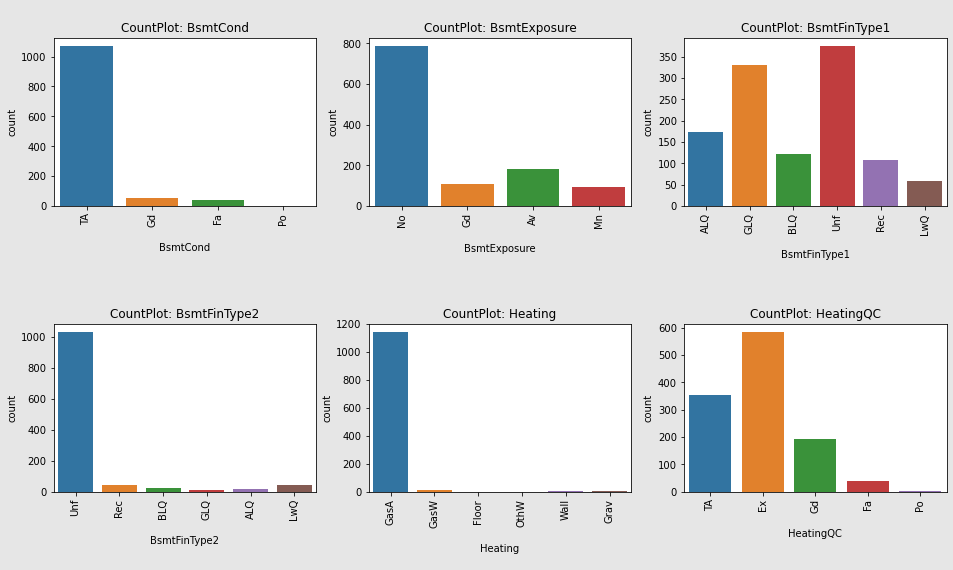
**Count Plot** for categorical variables gives some more insights of dataset:

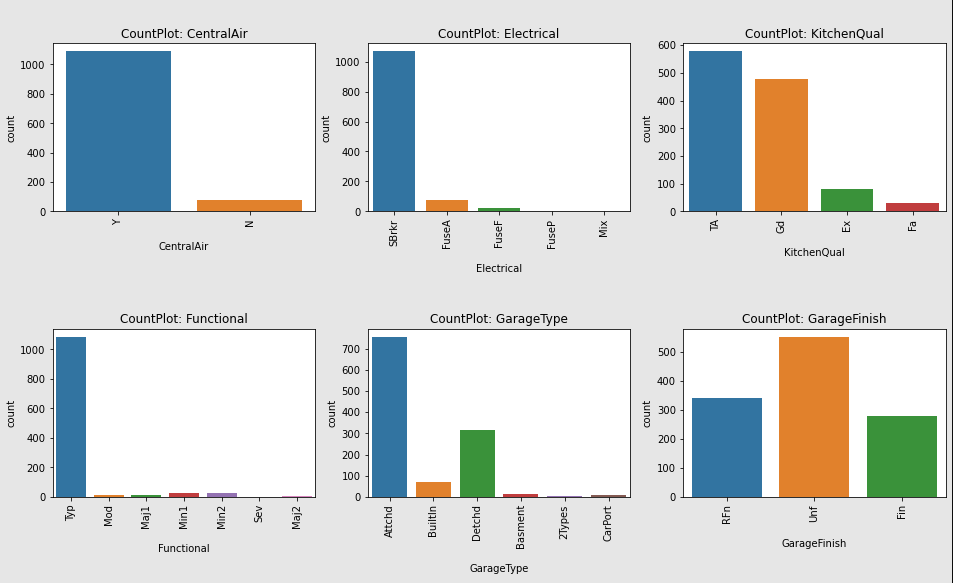


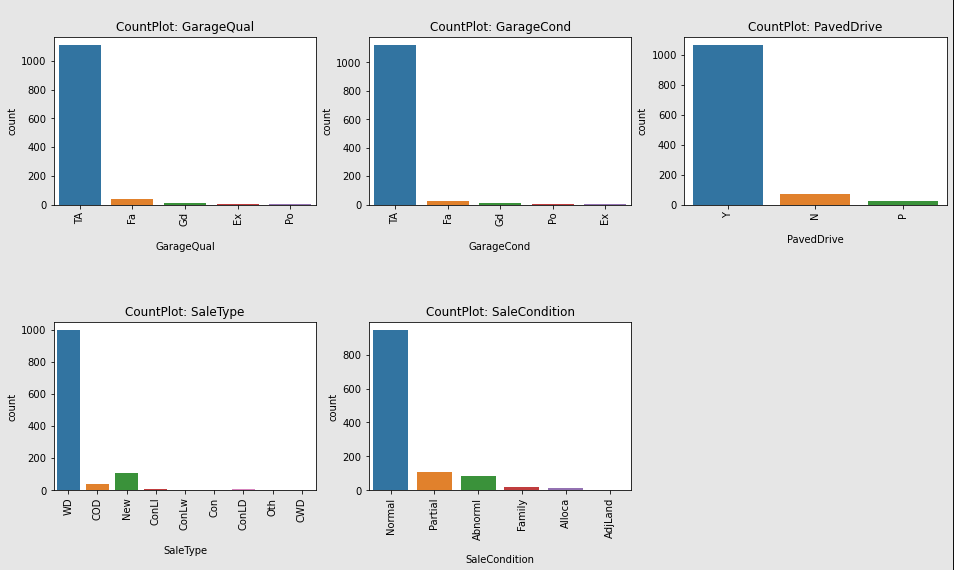












#### Remarks:

**for feature MSSubClass:**

* Maximum number of records are available for 20.
* Minimum number of records are available for 40.
* Majority of records are for 20, 60, 50 & 120.

**for feature MSZoning:**

* Maximum number of records are available for RL.
* Minimum number of records are available for C(all).

**for feature Street:**

* Maximum number of records are available for Pave.
* Minimum number of records are available for Grvl.

**for feature LotShape:**

* Maximum number of records are available for Reg.
* Minimum number of records are available for IR3.
* Majority of records are for Reg & IR1.

**for feature LandContour:**

* Maximum number of records are available for Lvl.
* Minimum number of records are available for Low.

**for feature Utilities:**

* All of the records are for AllPub and hence, this **feature can be dropped** from the dataset as it is of no use.

**for feature LotConfig:**

* Maximum number of records are available for Inside.
* Minimum number of records are available for FR3.

**for feature LandSlope:**

* Maximum number of records are available for Gtl.
* Minimum number of records are available for Sev.

**for feature Neighborhood:**

* Maximum number of records are available for NAmes.
* Minimum number of records are available for Blueste.
* Majority of records are for NAmes, CollgCr, OldTown, Edwards, Somerst, Gilbert, Sawyer, NWAmes and NridgHt.

**for feature Condition1:**

* Most of the records are for Norm.

**for feature Condition2:**

* Most of the records are for Norm.

**for feature BldgType:**

* Most of the records are for 1Fam.

**for feature HouseStyle:**

* Most of the records are for 1Story and 2Story.

**for feature OverallQual:**

* Most of the records are for 5, 6, 7, 8, & 4.

**for feature OverallCond:**

* Most of the records are for 5, 6, 7.

**for feature RoofStyle:**

* Most of the records are for Gable.

**for feature RoofMatl:**

* Most of the records are for CompShg.

**for feature Exterior1st:**

* Most of the records are for VinylSd, HdBoard, MetalSd, Wd Sdng & Plywood.

**for feature Exterior2nd:**

* Most of the records are for VinylSd, HdBoard, MetalSd, Wd Sdng & Plywood.

**for feature MasVnrType:**

* Most of the records are for None and BrkFace.

**for feature ExterQual:**

* Most of the records are for TA & Gd.

**for feature ExterCond:**

* Most of the records are for TA.

**for feature Foundation:**

* Most of the records are for CBlock and PConc

**for feature BsmtQual:**

* Most of the records are for TA & Gd.

**for feature BsmtCond:**

* Most of the records are for TA.

**for feature BsmtExposure:**

* Most of the records are for No.

**for feature BsmtFinType1:**

* Most of the records are for Unf, GLQ and ALQ.

**for feature BsmtFinType2:**

* Most of the records are for Unf.

**for feature Heating:**

* Most of the records are for GasA.

**for feature HeatingQC:**

* Most of the records are for Ex, TA and Gd.

**for feature CentralAir:**

* Most of the records are for Y.

**for feature Electrical:**

* Most of the records are for SBrkr.

**for feature KitchenQual:**

* Most of the records are for TA & Gd.

**for feature Functional:**

* Most of the records are for Typ.

**for feature GarageType:**

* Most of the records are for Attchd and Detchd.

**for feature GarageFinish:**

* Most of the records are for Unf.

**for feature GarageQual:**

* Most of the records are for TA.

**for feature GarageCond:**

* Most of the records are for TA.

**for feature PavedDrive:**

* Most of the records are for Y.

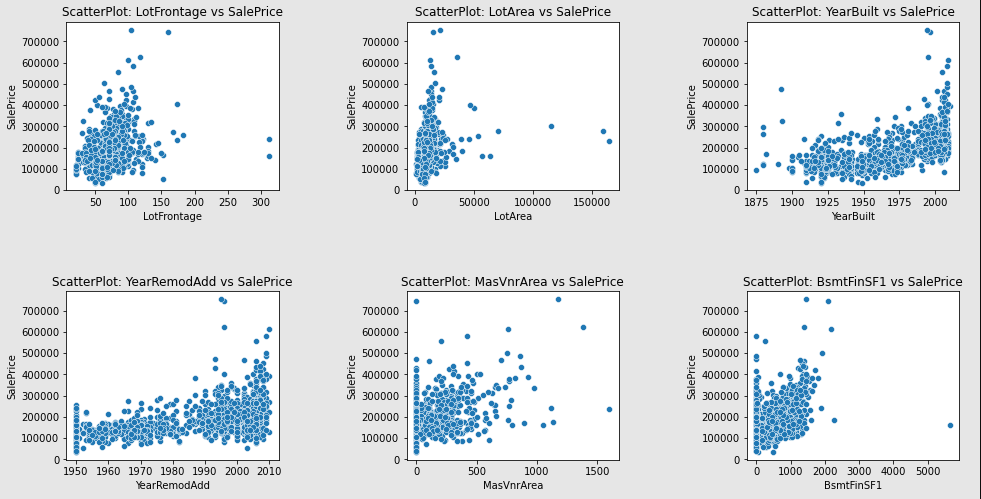
**for feature SaleType:**

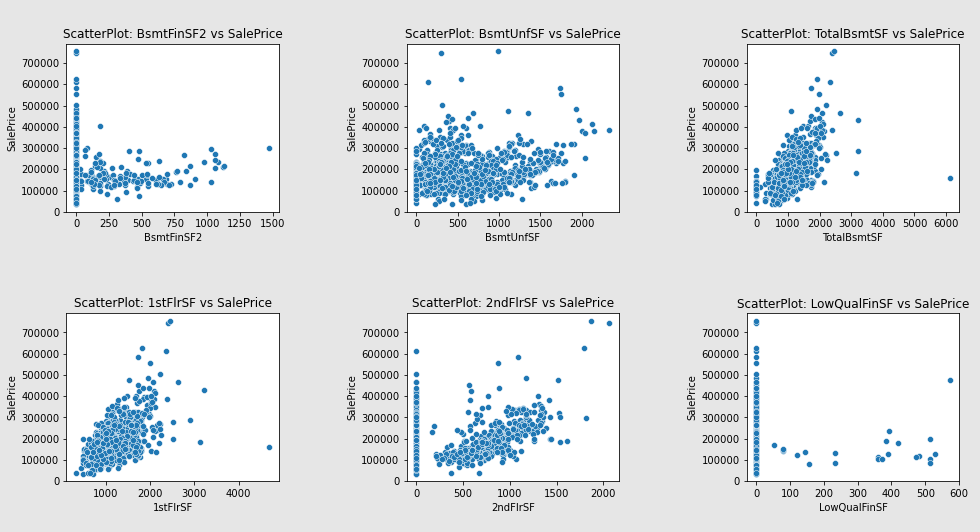
* Most of the records are for WD.

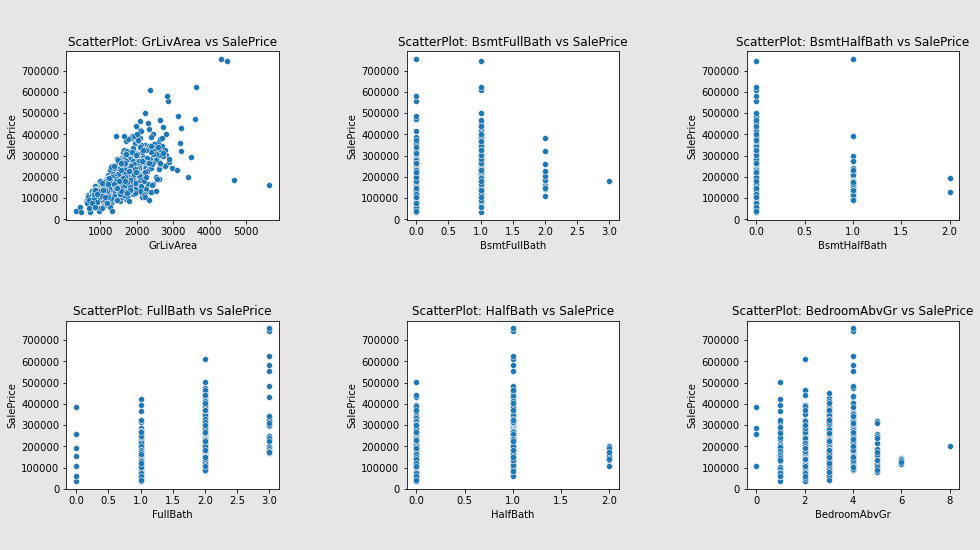
**for feature SaleCondition:**

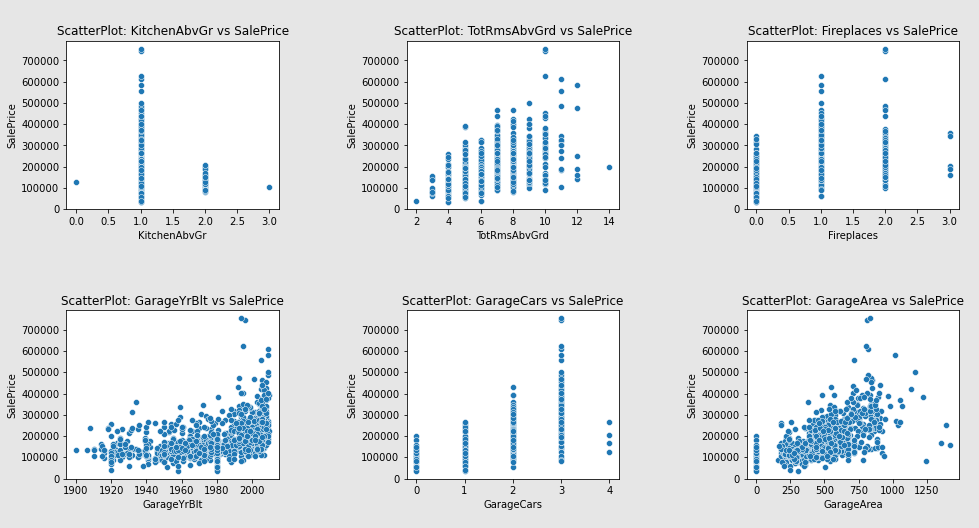
* Most of the records are for Normal.

1. Bivariate Analysis: Bivariate analysis is one of the simplest forms of quantitative analysis. It involves the analysis of two variables, for the purpose of determining the empirical relationship between them. We have analyzed the data and it’s relationship with target variable using scatterplot and barplot as shown below:

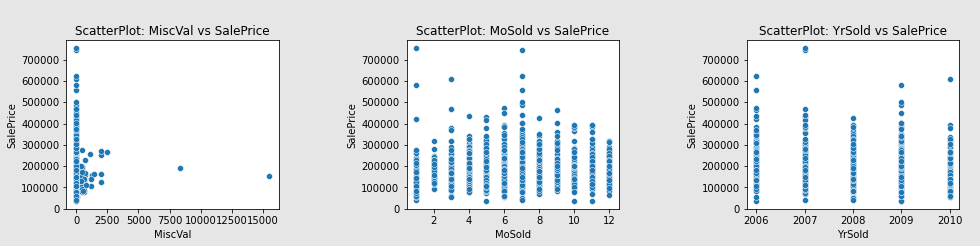












#### Remarks:

**for LotFrontage vs SalePrice:**

* As the LotFrontage increases SalePrice also increases.
* Most of the SalePrice ranges from 100000 to 350000 for LotFrontage between 20 to 110.

**for LoatArea vs SalePrice:**

* As the LotArea increases SalePrice also increases.
* Most of the SalePrice ranges from 100000 to 300000 for LotArea between 0 to 20000.

**for YearBuilt vs SalePrice:**

* As the YearBuilt increases SalePrice increases.
* Most of the SalePrice ranges from 90000 to 350000 for YearBuilt between 1904 to 2004.

**for YearRemodAdd vs SalePrice:**

* As the YearRemodAdd increases SalePrice increases.
* Most of the SalePrice ranges from 100000 to 400000 for YearRemodAdd between 1950 to 2010.

**for MasVnrArea vs SalePrice:**

* As the MasVnrArea increases SalePrice increases.
* Most of the SalePrice ranges from 100000 to 400000 for MasVnrArea between 0 to 500.

**for BsmtFinSF1 vs SalePrice:**

* As the BsmtFinSF1 increases SalePrice increases.
* Most of the SalePrice ranges from 100000 to 350000 for BsmtFinSF1 between 0 to 1400.

**for BsmtFinSF2 vs SalePrice:**

* SalePrice ranges from 100000 to 250000 for BsmtFinSF2 from 0 to 1000.

**for BsmtUnfSF vs SalePrice:**

* SalePrice ranges from 100000 to 300000 for BsmtUnfSF from 0 to 1800.

**for TotalBsmtSF vs SalePrice:**

* SalePrice increases as TotalBsmtSF increases.
* Most of SalePrice ranges from 10000 to 450000 for TotalBsmtSF from 0 to 2200.

**for 1stFlrSF vs SalePrice:**

* SalePrice increases as 1stFlrSF increases.
* Most of SalePrice ranges from 10000 to 480000 for 1stFlrSF from 400 to 2200.

**for 2ndFlrSF vs SalePrice:**

* SalePrice increases as 2ndFlrSF increases.
* Most of SalePrice ranges from 6000 to 350000 for 2ndFlrSF from 300 to 1400.

**for LowQualFinSF vs SalePrice:**

* Most of the SalePrice ranges from 0 to 610000 for LowQualFinSF 0.

**for GrLivArea vs SalePrice:**

* SalePrice increases as GrLivArea increases.
* Most of SalePrice ranges from 10000 to 400000 for GrLivArea from 600 to 3000.

**for BsmtFullBath vs SalePrice:**

* SalePrice ranges from 10000 to 410000 for BsmtFullBath 0.
* SalePrice ranges from 10000 to 500000 for BsmtFullBath 1.
* SalePrice ranges from 100000 to 300000 for BsmtFullBath 2.

**for BsmtHalfBath vs SalePrice:**

* SalePrice ranges from 10000 to 610000 for BsmtHalfBath 0.
* SalePrice ranges from 100000 to 300000 for BsmtHalfBath 1.
* SalePrice ranges from 100000 to 200000 for BsmtHalfBath 2.

**for FullBath vs SalePrice:**

* SalePrice ranges from 10000 to 250000 for FullBath 0.
* SalePrice ranges from 10000 to 410000 for FullBath 1.
* SalePrice ranges from 100000 to 500000 for FullBath 2.
* SalePrice ranges from 190000 to 600000 for FullBath 3.

**for HalfBath vs SalePrice:**

* SalePrice ranges from 10000 to 500000 for HalfBath 0.
* SalePrice ranges from 100000 to 600000 for HalfBath 1.
* SalePrice ranges from 100000 to 200000 for HalfBath 2.

**for BedroomAbvGr vs SalePrice:**

* SalePrice ranges from 100000 to 390000 for BedroomAbvGr 0.
* SalePrice ranges from 10000 to 410000 for BedroomAbvGr 1.
* SalePrice ranges from 10000 to 450000 for BedroomAbvGr 2.
* SalePrice ranges from 10000 to 430000 for BedroomAbvGr 3.
* SalePrice ranges from 80000 to 610000 for BedroomAbvGr 4.
* SalePrice ranges from 70000 to 250000 for BedroomAbvGr 5.
* SalePrice ranges from 100000 to 130000 for BedroomAbvGr 6.
* SalePrice for BedroomAbvGr 8 is 200000.

**for KitchenAbvGr vs SalePrice:**

* SalePrice for KitchenAbvGr 0 is aprox. 120000.
* SalePrice ranges from 10000 to 620000 for KitchenAbvGr 1.
* SalePrice ranges from 90000 to 210000 for KitchenAbvGr 2.
* SalePrice for KitchenAbvGr 3 is aprox. 90000.

**for TotRmsAbvGrd vs SalePrice:**

* SalePrice increases as TotRmsAbvGrd increases.
* Most of SalePrice ranges from 10000 to 400000 for TotRmsAbvGrd 4 to 12.

**for Fireplaces vs SalePrice:**

* SalePrice ranges from 10000 to 350000 for Fireplaces 0.
* SalePrice ranges from 20000 to 600000 for Fireplaces 1.
* SalePrice ranges from 80000 to 500000 for Fireplaces 2.
* SalePrice ranges from 180000 to 350000 for Fireplaces 3.

**for GarageYrBlt vs SalePrice:**

* SalePrice increases as GarageYrBlt increases.
* Most of SalePrice ranges from 80000 to 400000 for GarageYrBlt 1920 to 2010.

**for GarageCars vs SalePrice:**

* SalePrice ranges from 10000 to 200000 for GarageCars 0.
* SalePrice ranges from 10000 to 280000 for GarageCars 1.
* SalePrice ranges from 20000 to 420000 for GarageCars 2.
* SalePrice ranges from 50000 to 650000 for GarageCars 3.
* SalePrice ranges from 110000 to 300000 for GarageCars 4.

**for GarageArea vs SalePrice:**

* SalePrice increases as GarageArea increases.
* Most of SalePrice ranges from 50000 to 500000 for GarageArea 240 to 900.

**for WoodDeckSF vs SalePrice:**

* SalePrice increases as WoodDeckSF increases.
* Most of SalePrice ranges from 80000 to 400000 for WoodDeckSF 0 to 400.

**for OpenPorchSF vs SalePrice:**

* SalePrice increases as OpenPorchSF increases.
* Most of SalePrice ranges from 100000 to 400000 for OpenPorchSF 0 to 250.

**for EnclosedPorch vs SalePrice:**

* SalePrice increases as EnclosedPorch increases.
* Most of SalePrice ranges from 90000 to 200000 for EnclosedPorch 10 to 300.

**for 3SsnPorch vs SalePrice:**

* Most of the SalePrice ranges from 10000 to 620000 for 3SsnPorch 0.
* SalePrice ranges from 100000 to 400000 for 3SsnPorch 100 to 300.

**for ScreenPorch vs SalePrice:**

* Most of the SalePrice ranges from 10000 to 620000 for ScreenPorch 0.
* SalePrice ranges from 100000 to 400000 for ScreenPorch 50 to 300.

**for PoolArea vs SalePrice:**

* Most of the SalePrice ranges from 10000 to 620000 for PoolArea 0.
* SalePrice ranges from 150000 to 300000 for PoolArea 450 to 800.

**for MiscVal vs SalePrice:**

* Most of the SalePrice ranges from 10000 to 620000 for MiscVal 0.
* SalePrice ranges from 50000 to 280000 for MiscVal 500 to 2500.

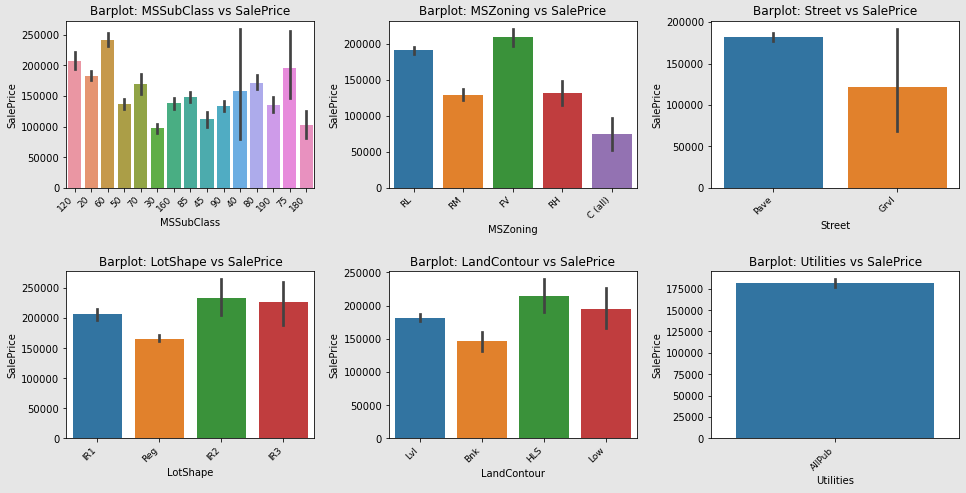
**for MoSold vs SalePrice:**

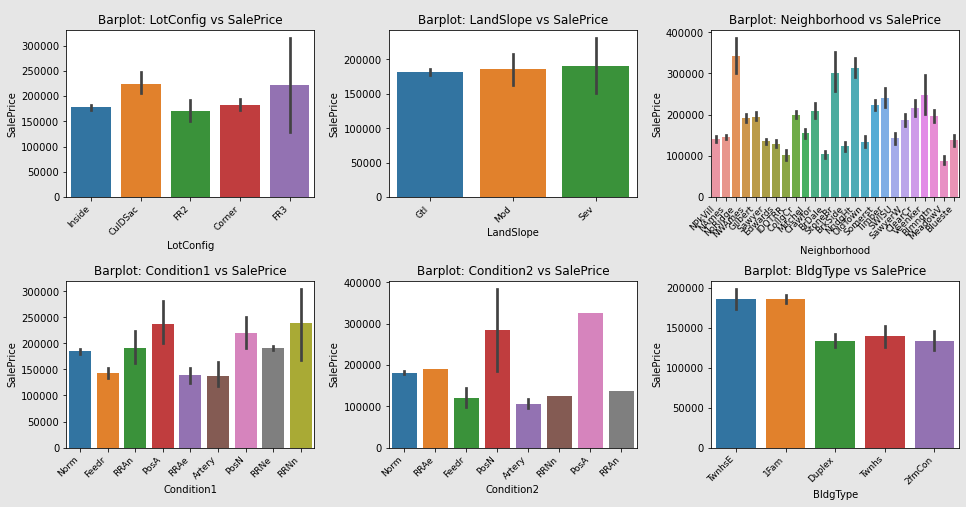
* Most of the SalePrice ranges from 10000 to 280000 for MoSold 1.
* Most of the SalePrice ranges from 80000 to 310000 for MoSold 2.
* Most of the SalePrice ranges from 30000 to 350000 for MoSold 3.
* Most of the SalePrice ranges from 60000 to 350000 for MoSold 4.
* Most of the SalePrice ranges from 10000 to 410000 for MoSold 5.
* Most of the SalePrice ranges from 30000 to 450000 for MoSold 6.
* Most of the SalePrice ranges from 20000 to 600000 for MoSold 7.
* Most of the SalePrice ranges from 50000 to 350000 for MoSold 8.
* Most of the SalePrice ranges from 60000 to 400000 for MoSold 9.
* Most of the SalePrice ranges from 30000 to 380000 for MoSold 10.
* Most of the SalePrice ranges from 10000 to 380000 for MoSold 11.
* Most of the SalePrice ranges from 40000 to 310000 for MoSold 12.

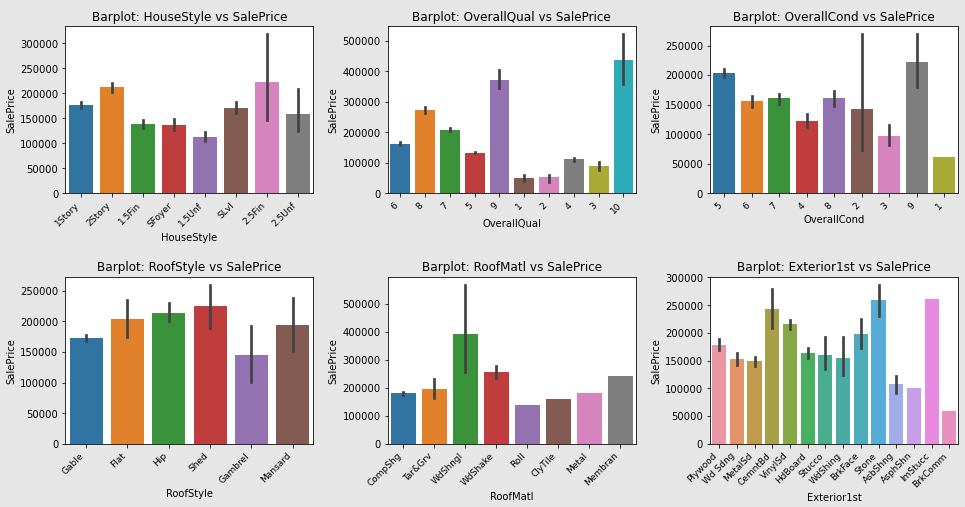
**for YrSold vs SalePrice:**

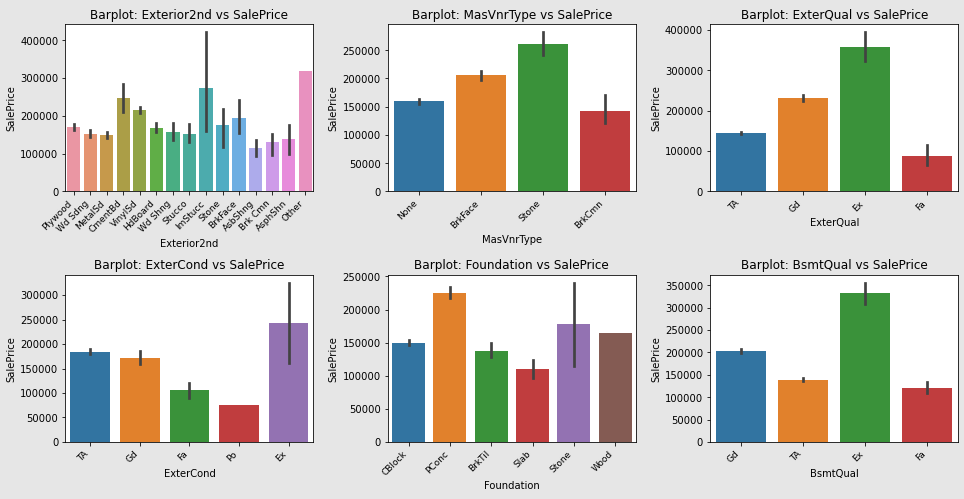
* Most of the SalePrice ranges from 10000 to 480000 for YrSold 2006.
* Most of the SalePrice ranges from 10000 to 480000 for YrSold 2007.
* Most of the SalePrice ranges from 10000 to 470000 for YrSold 2008.
* Most of the SalePrice ranges from 10000 to 600000 for YrSold 2009.
* Most of the SalePrice ranges from 20000 to 400000 for YrSold 2010.

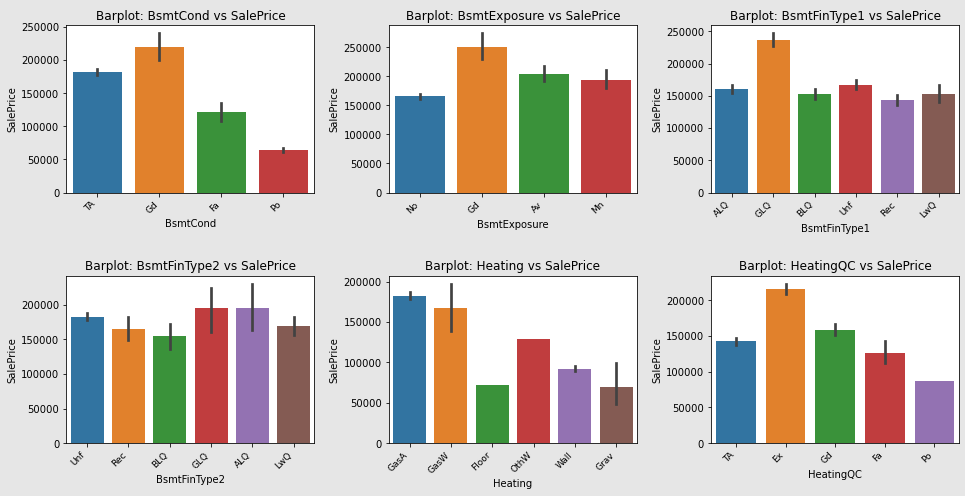
Barplot for relationship between categorical variable and target:

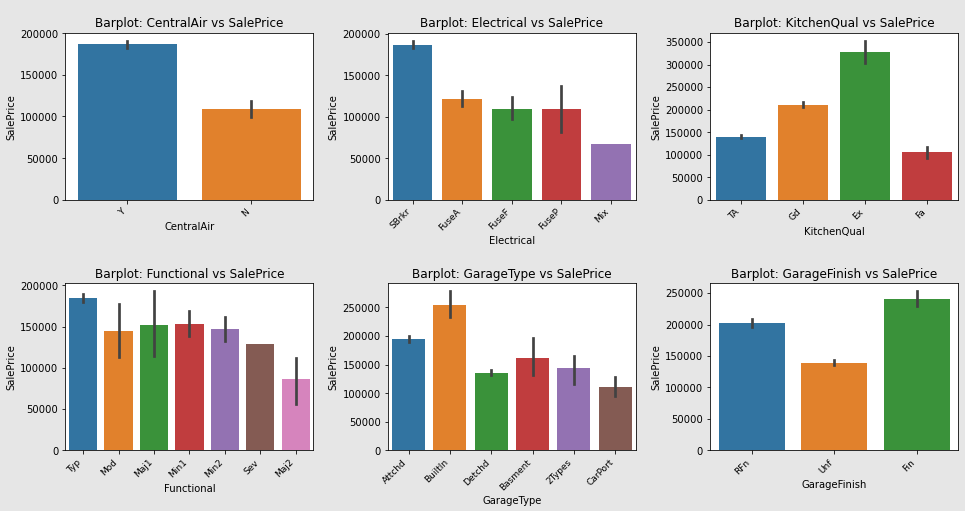


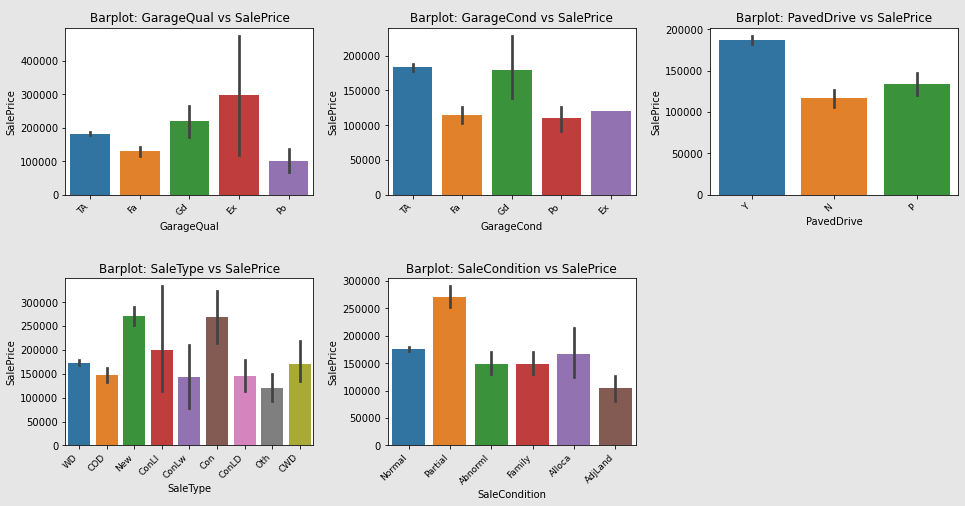












#### Remarks:

**for feature MSSubClass vs SalePrice:**

* MSSubClass with 60 has highest SalePrice while with 30 has lowest SalePrice.

**for feature MSZoning vs SalePrice:**

* MSZoning with FV has highest SalePrice while with C(all) has lowest SalePrice.

**for feature Street vs SalePrice:**

* Street with Pave has highest SalePrice while with Grvl has lowest SalePrice.

**for feature LotShape vs SalePrice:**

* LotShape with R2 and R3 has highest SalePrice while with Reg has lowest SalePrice.

**for feature LandContour vs SalePrice:**

* LandContour with HLS has highest SalePrice while with Bnk has lowest SalePrice.

**for feature Utilities vs SalePrice:**

* Utilities has a single AllPub, therefore, it can be **dropped from dataset** as it is of no use.

**for feature LotConfig vs SalePrice:**

* LotConfig with CulDSac and FR3 has highest SalePrice while with FR2 has lowest SalePrice.

**for feature LandSlope vs SalePrice:**

* LandSlope with Sev, Mod and Gtl has almost similar SalePrice.

**for feature Neighborhood vs SalePrice:**

* Neighborhood with NoRidge, NodgHt and StoneBr has highest SalePrice while with MeadowV has lowest SalePrice.

**for feature Condition1 vs SalePrice:**

* Condition1 with PosA and RRNn has highest SalePrice while with RRAe and Artery has lowest SalePrice.

**for feature Condition2 vs SalePrice:**

* Condition2 with PosA has highest SalePrice while with Artery, Feedr and RRNn has lowest SalePrice.

**for feature BldgType vs SalePrice:**

* BldgType with TwnhsE and 1Fam has highest SalePrice while with Duplex, Twnhs and 2fmCon has lowest SalePrice.

**for feature HouseStyle vs SalePrice:**

* HouseStyle with 2.5Fin and 2Story has highest SalePrice while with 1.5Unf has lowest SalePrice.

**for feature OverallQual vs SalePrice:**

* OverallQual with 10 and 9 has highest SalePrice while with 1 and 2 has lowest SalePrice.

**for feature OverallCond vs SalePrice:**

* LotConfig with 9 and 5 has highest SalePrice while with 1 has lowest SalePrice.

**for feature RoofStyle vs SalePrice:**

* RoofStyle with Shed and Hip has highest SalePrice while with Gambrel has lowest SalePrice.

**for feature RoofMatl vs SalePrice:**

* RoofMatl with WdShngl has highest SalePrice while with Roll has lowest SalePrice.

**for feature Exterior1st vs SalePrice:**

* Exterior1st with Stone, lmStucc and CemntBd has highest SalePrice while with BrkComm has lowest SalePrice.

**for feature Exterior2nd vs SalePrice:**

* Exterior2nd with Other and lmStucc has highest SalePrice while with AsbShng has lowest SalePrice.

**for feature MasVnrType vs SalePrice:**

* MasVnrType with Stone has highest SalePrice while with BrkCmn has lowest SalePrice.

**for feature ExterQual vs SalePrice:**

* ExterQual with Ex has highest SalePrice while with Fa has lowest SalePrice.

**for feature ExterCond vs SalePrice:**

* ExterCond with Ex has highest SalePrice while with Po has lowest SalePrice.

**for feature Foundation vs SalePrice:**

* Foundation with PConc has highest SalePrice while with Slab has lowest SalePrice.

**for feature BsmtQual vs SalePrice:**

* Bsmt with Ex has highest SalePrice while with Fa and TA has lowest SalePrice.

**for feature BsmtCond vs SalePrice:**

* BsmtCond with Gd has highest SalePrice while with Po has lowest SalePrice.

**for feature BsmtExposure vs SalePrice:**

* BsmtExposure with Gd has highest SalePrice while with No has lowest SalePrice.

**for feature BsmtFinType1 vs SalePrice:**

* BsmtFinType1 with GLQ has highest SalePrice while with Rec, BLQ, Unf, LwQ and ALQ has lowest SalePrice but almost similer SalePrice.

**for feature BsmtFinType2 vs SalePrice:**

* BsmtFinType2 with GLQ and ALQ has highest SalePrice while with BLQ has lowest SalePrice.

**for feature Heating vs SalePrice:**

* Heating with GasA has highest SalePrice while with Floor has lowest SalePrice.

**for feature HeatingQC vs SalePrice:**

* HeatingQC with Ex has highest SalePrice while with Po has lowest SalePrice.

**for feature CentralAir vs SalePrice:**

* CentralAir with Y has highest SalePrice while with N has lowest SalePrice.

**for feature Electrical vs SalePrice:**

* Electrical with SBrkr has highest SalePrice while with Mix has lowest SalePrice.

**for feature KitchenQual vs SalePrice:**

* KitchenQual with Ex has highest SalePrice while with Fa has lowest SalePrice.

**for feature LotConfig vs SalePrice:**

* LotConfig with CulDSac and FR3 has highest SalePrice while with FR2 has lowest SalePrice.

**for feature Functional vs SalePrice:**

* Functional with Typ has highest SalePrice while with Maj2 has lowest SalePrice.

**for feature GarageType vs SalePrice:**

* GarageType with BuiltIn has highest SalePrice while with CarPort has lowest SalePrice.

**for feature GarageFinish vs SalePrice:**

* GarageFinish with Fin has highest SalePrice while with Unf has lowest SalePrice.

**for feature GarageQual vs SalePrice:**

* GarageQual with Ex has highest SalePrice while with Po has lowest SalePrice.

**for feature GarageCond vs SalePrice:**

* GarageCond with Gd and TA has highest SalePrice while with Po and Fa has lowest SalePrice.

**for feature PavedDrive vs SalePrice:**

* PavedDrive with Y has highest SalePrice while with N has lowest SalePrice.

**for feature SaleType vs SalePrice:**

* SaleType with New and Con has highest SalePrice while with COD, ConLw, ConLD and Oth has lowest SalePrice.

**for feature SaleCondition vs SalePrice:**

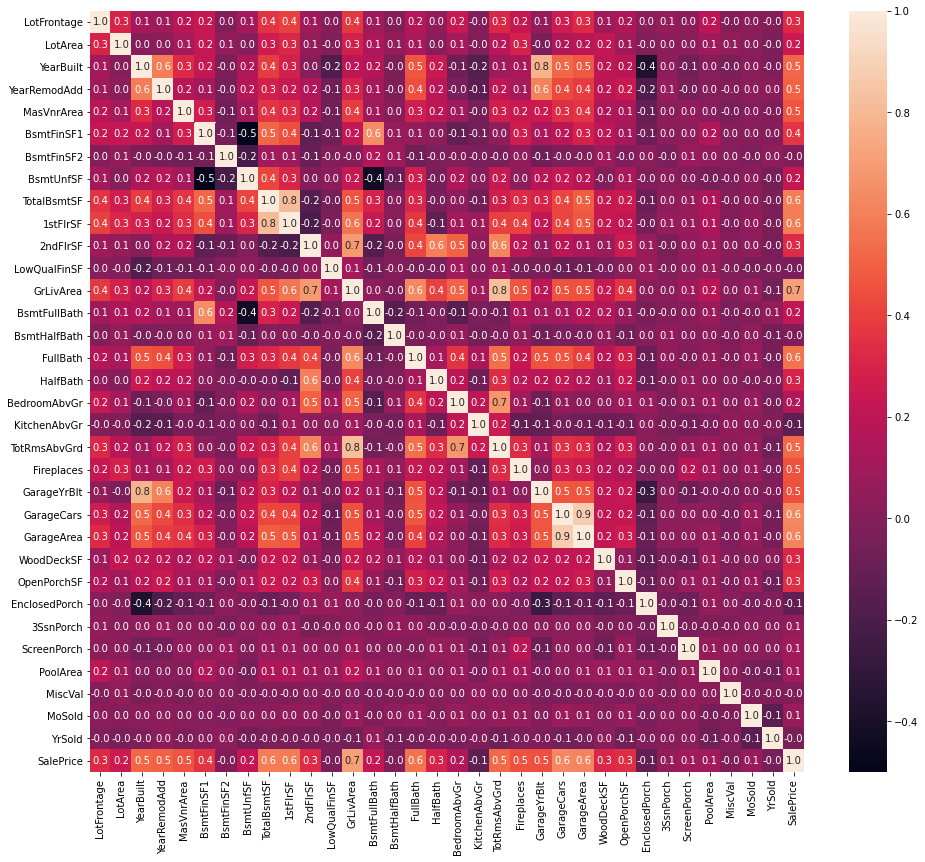
* SaleCondition with Partial has highest SalePrice while with AdjLand has lowest SalePrice.

**for feature LotConfig vs SalePrice:**

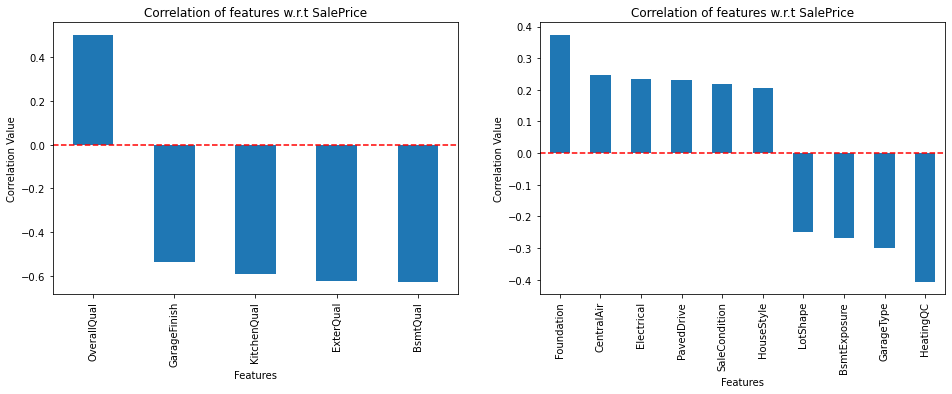
* LotConfig with CulDSac and FR3 has highest SalePrice while with FR2 has lowest SalePrice.

1. Multivariate Analysis: Multivariate analysis is based on the principles of multivariate statistics, which involves observation and analysis of more than one statistical outcome variable at a time.

Heatmap is being used to represent the correlation of features from a scale of -1.0 to 1.0. After going through heatmap it is found that Features GrLivArea, TotalBsmtSF, 1stFlrSF, 2ndFlrSF, FullBath, GarageCars, GarageArea, YearBuilt, YearRemodAdd, MasVnrArea, TotRmsAbvGrd, FirePlaces, GarageYrBlt are **positively high correlated to target SalePrice** while features like BsmtFinSF1, LotFrontage, 2ndFlrSF, HalfBath, WoodDeckSF, OpenPorchSF are **positively good correlated to SalePrice.**



To get the correlation of categorical features w.r.t target, a barplot is being used:



#### Remarks:

* Features OverallQual has **positively high correlation** with SalePrice while features BmstQual, ExterQual, KitchenQual, GarageFinish has **negatively high correlation** with SalePrice.
* Features Foundation, CentralAir, Electrical, PavedDrive, SaleCondition, HouseStyle has **positively good correlation** with SalePrice while features HeatingQC, GarageType, BsmtExposure, LotShape has **negatively good correlation** with SalePrice.
* All other features has low correlation with SalePrice.
* Interpretation of the Results

Starting with **univariate analysis**, with help of **distplot** we found that data in all continuous features are not distributted normally and needs to be scaled before sending for model training. Moving further, we also found that data in most of the features are skewed and spreaded, therefore, this needs to be treated before model training. Also, it is observed that except features YearBuilt, YearRemodAdd, 2ndFlrSF, BsmtFullBath, BsmtHalfBath, FullBath, HalfBath, BedroomAbvGr, KitchenAbvGr, TotRmsAbvGrd, Fireplaces, GarageYrBlt, GarageCars, MoSold and YrSold, **boxplot** shows some or more outliers might be present in other features. Further analysing the categorical features using **countplot**, it is observed that feature ***Utilities*** has a single value for all records, therefore, it ***needs to be dropped*** because of no use in model training. After the we came up with **bivariate analysis** which gives the close look of relationship between features and target variable SalePrice. Using **scatterplot** for continuous features, it is observed that SalePrice increases with the increase in feature LotFrontage, LotArea, YearBuilt, YearRemodAdd, MasVnrArea, BsmtFinSF1, TotalBsmtSF, 1stFlrSF, LowQualFinSF, GrLivArea, TotRmsAbvGrd, GarageYrBlt, GarageArea, WoodDeckSF, OpenPorchSF and MiscVal. Also, with the help of **barplot**, it is observed that each feature has value which contributes to the hike in SalePrice as well as reduced the SalePrice. Moving further with **multi-variate analysis** of continuous features, it is observed that features GrLivArea, TotalBsmtSF, 1stFlrSF, 2ndFlrSF, FullBath, GarageCars, GarageArea, YearBuilt, YearRemodAdd, MasVnrArea, TotRmsAbvGrd, FirePlaces, GarageYrBlt are positively high correlated to target SalePrice while features like BsmtFinSF1, LotFrontage, 2ndFlrSF, HalfBath, WoodDeckSF, OpenPorchSF are **positively good correlated** to SalePrice. Also, for categorical features, it is observed that OverallQual has **positively high correlation** with SalePrice while features BmstQual, ExterQual, KitchenQual, GarageFinish has **negatively high correlation** with SalePrice and features Foundation, CentralAir, Electrical, PavedDrive, SaleCondition, HouseStyle has **positively good correlation** with SalePrice while features HeatingQC, GarageType, BsmtExposure, LotShape has **negatively good correlation** with SalePrice. All other features has low or no correlation with SalePrice. The final shape of dataset is **1168 rows and 74 columns.**

**CONCLUSION**

* Key Findings and Conclusions of the Study

From the model performance comparison it is clear that **AdaBoostRegressor** out-performs the other models with **r2\_score of 80.83%** and **lowest difference between r2\_score and cross\_val\_score**. Therefore, continuing with **AdaBoostRegressor** as final model.

* Learning Outcomes of the Study in respect of Data Science

During the data analysis, some feature contains null values which I have replaced them with median. But these values can also be replaced with the mean of the feature which might impact the model performance either in positive or negative way. As of now, I am finishing this project with my current approach which gives the **final accuracy score of 80.83% and cross\_val\_score: 77.53%** and this can be further improved by training with more specific data.

* Limitations of this work and Scope for Future Work

Current model is limited to housing prediction data but this can further be improved for other sectors of property price prediction by training the model accordingly. The overall score can also be improved further by training the model with more specific data.