FLIP ROBO

Housing Price Prediction

Submitted by:
AKASH TALWAR

Problem Statement

- 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.

- 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 data is provided in the CSV file below.
- The company is looking at prospective properties to buy houses to enter the market. You 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?

Business Goal

- You 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.

Technical Requirements:

- · 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 to be performed to gain relationships of important variable and price.
- · Data contains numerical as well as categorical variable. You need to handle them accordingly.
- · You have to build Machine Learning models, apply regularization and determine the optimal values of Hyper Parameters.
- · You need to find important features which affect the price positively or negatively.

Features:

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
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

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

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

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)

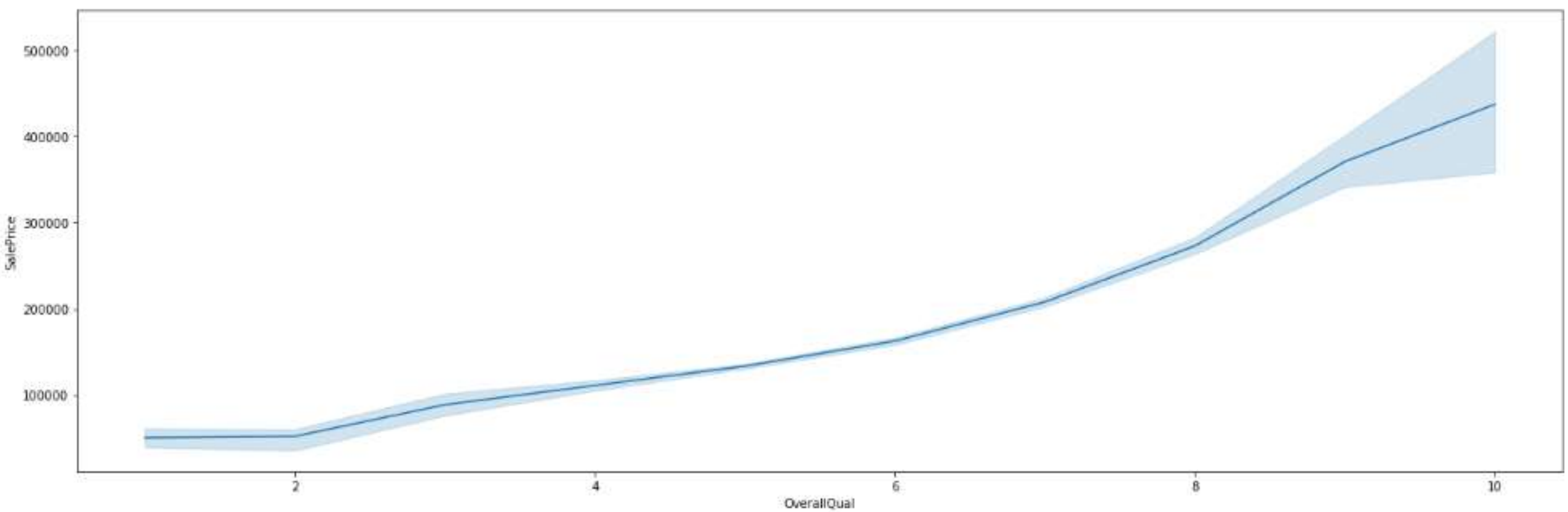
EDA: Exploratory Data Analysis

❖ **Features that positively contribute to Sale Price (target variable)**

1. OverallQual: Rates the overall material and finish of the house

```
#Bi variate analysis
#OverallQual vs SalePrice
plt.figure(figsize=(22,7))
sns.lineplot(x="OverallQual",y="SalePrice",data=dfn_train)
```

```
<AxesSubplot:xlabel='OverallQual', ylabel='SalePrice'>
```



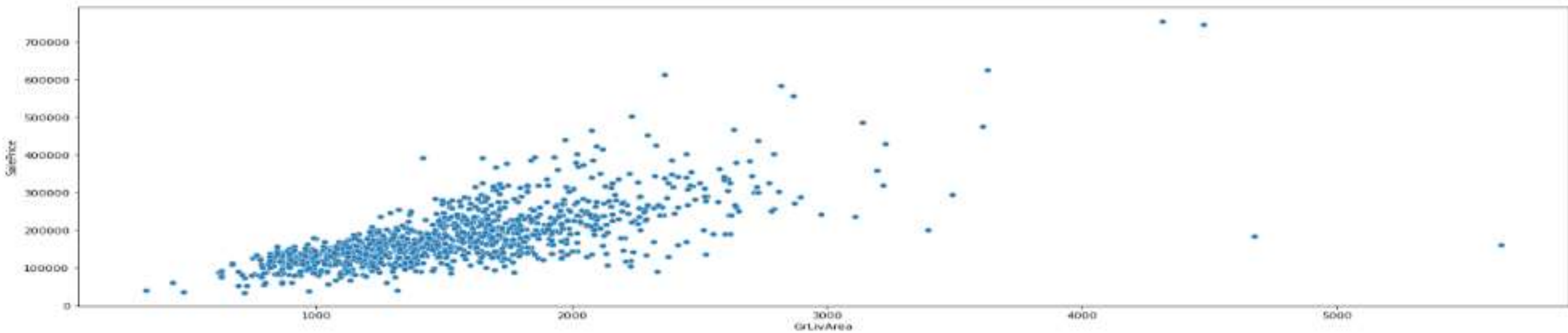

```
Observation: Rates of the overall material used and finish of the house has strong positive correlation with House sales price
```

❖ Features that positively contribute to Sale Price (target variable)

2. GrLivArea: Above grade (ground) living area square feet

```
#GrLivArea vs SalePrice
plt.figure(figsize=(22,7))
sns.scatterplot(x="GrLivArea",y="SalePrice",data=dfn_train)
```

```
<AxesSubplot:xlabel='GrLivArea', ylabel='SalePrice'>
```

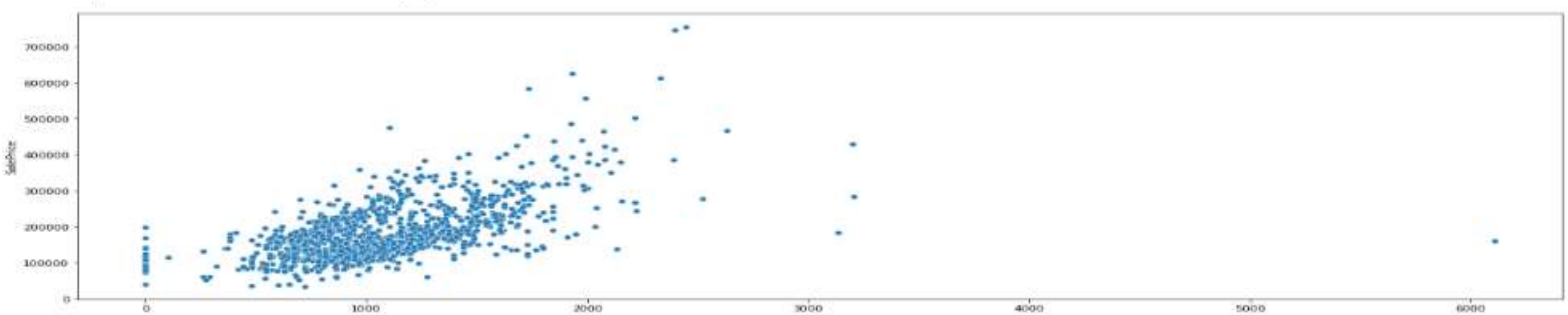



```
Observation:Feature GrLivArea [Above grade (ground) living area square feet] is positively contributing to house sales price.
```

3. TotalBsmtSF: Total square feet of basement area

```
#TotalBsmtSF vs SalePrice
plt.figure(figsize=(22,7))
sns.scatterplot(x="TotalBsmtSF",y="SalePrice",data=dfn_train)
```

```
<AxesSubplot:xlabel='TotalBsmtSF', ylabel='SalePrice'>
```



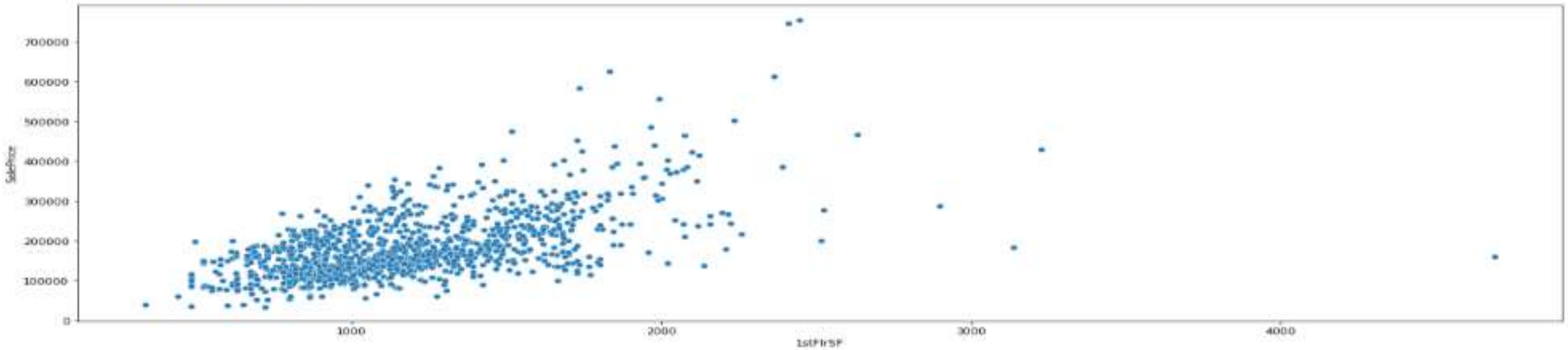

```
Observation:More the Feature TotalBsmtSF [Total square feet of basement area], more is the house sales price.
```

❖ Features that positively contribute to Sale Price (target variable)

4. 1stFlrSF: First Floor square feet

```
#1stFlrSF vs SalePrice
plt.figure(figsize=(22,7))
sns.scatterplot(x="1stFlrSF",y="SalePrice",data=dfn_train)
```

```
<AxesSubplot:xlabel='1stFlrSF', ylabel='SalePrice'>
```

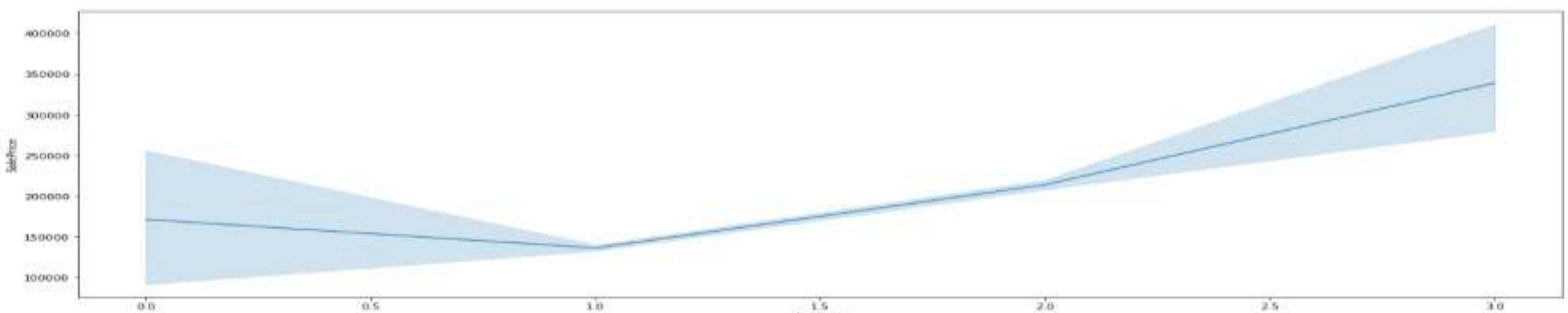



```
Observation:Feature 1stFlrSF [First Floor square feet] contributes positively to the house sales price.
```


5. BsmtFullBath: Basement full bathrooms

```
#FullBath vs SalePrice
plt.figure(figsize=(22,7))
sns.lineplot(x="FullBath",y="SalePrice",data=dfn_train)
```

```
<AxesSubplot:xlabel='FullBath', ylabel='SalePrice'>
```



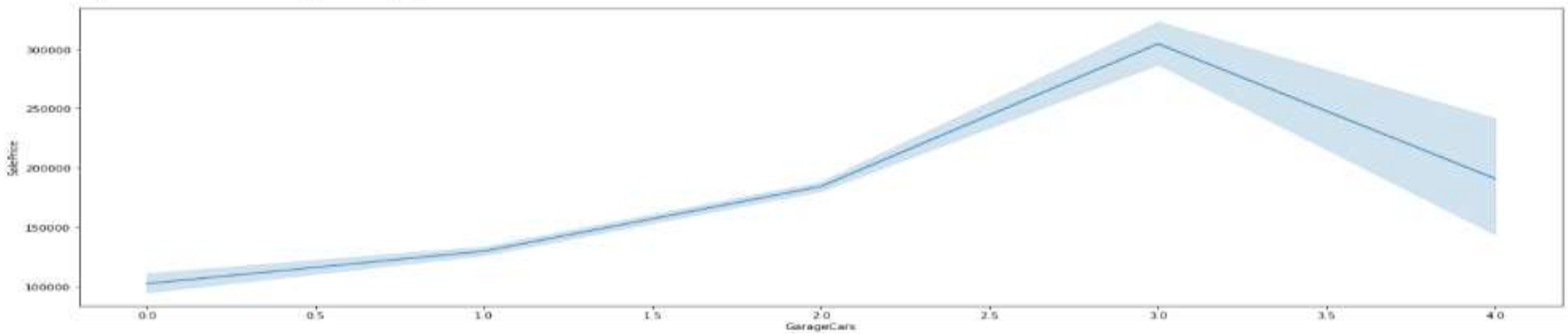

```
Observation:House price are higher if basements have higher number of bathrooms ie FullBath [Basement full bathrooms]
```

❖ Features that positively contribute to Sale Price (target variable)

6. GarageCars: Size of garage in car capacity

```
#GarageCars vs SalePrice
plt.figure(figsize=(22,7))
sns.lineplot(x="GarageCars",y="SalePrice",data=dfn_train)
```

```
<AxesSubplot:xlabel='GarageCars', ylabel='SalePrice'>
```

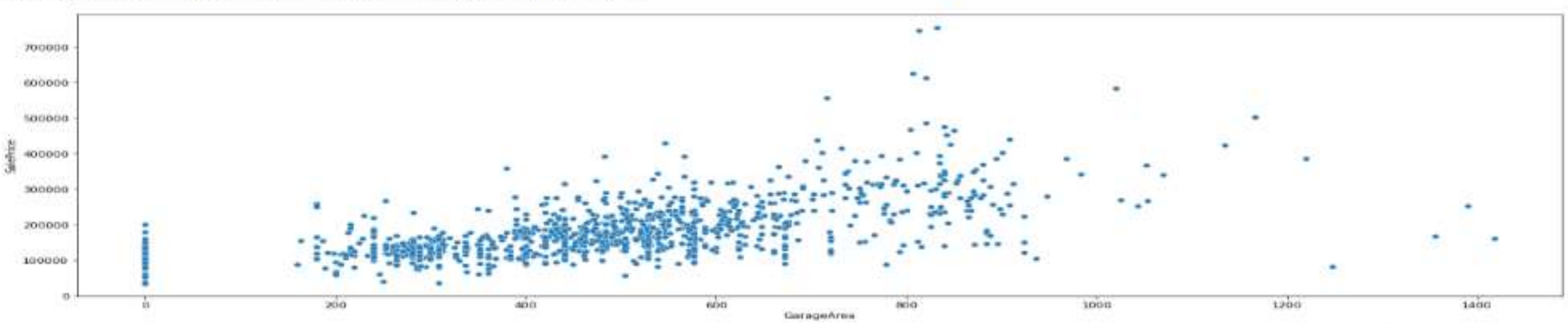



```
Observation:Houses with provision to park more number of cars are higher in price ie GarageArea[Size of garage in car capacity]
```

7. GarageArea: Size of garage in square feet

```
#GarageArea vs SalePrice
plt.figure(figsize=(22,7))
sns.scatterplot(x="GarageArea",y="SalePrice",data=dfn_train)
```

```
<AxesSubplot:xlabel='GarageArea', ylabel='SalePrice'>
```



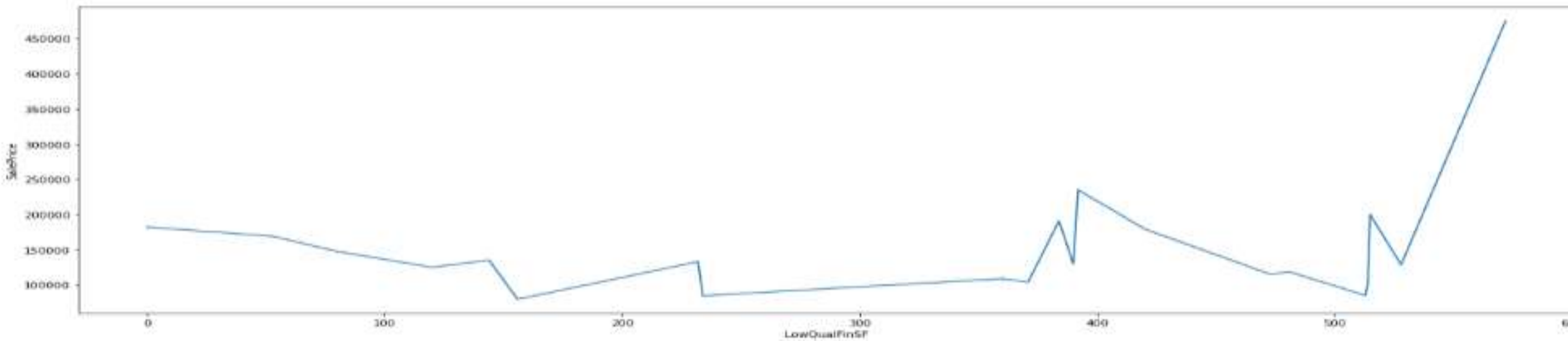

```
Observation: Feature GarageArea [Size of garage in square feet] contributes positively to house price
```

❖ Features that negatively affect Sale Price (target variable)

1. LowQualFinSF: Low quality finished square feet (all floors)

```
#Features that affect the Hose price negatively
#LowQualFinSF vs SalePrice
plt.figure(figsize=(22,7))
sns.lineplot(x="LowQualFinSF",y="SalePrice",data=dfn_train)
```

```
<AxesSubplot:xlabel='LowQualFinSF', ylabel='SalePrice'>
```

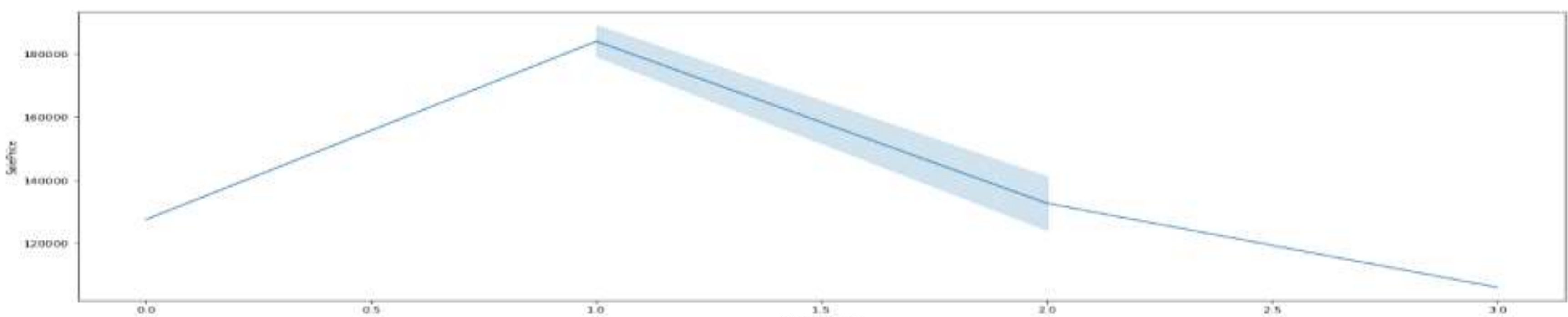



```
Observation: Feature like "Low quality finished square feet (all floors)" bring down the house price
```


2. Kitchen: Kitchens above grade

```
#KitchenAbvGr vs SalePrice
plt.figure(figsize=(22,7))
sns.lineplot(x="KitchenAbvGr",y="SalePrice",data=dfn_train)
```

```
<AxesSubplot:xlabel='KitchenAbvGr', ylabel='SalePrice'>
```



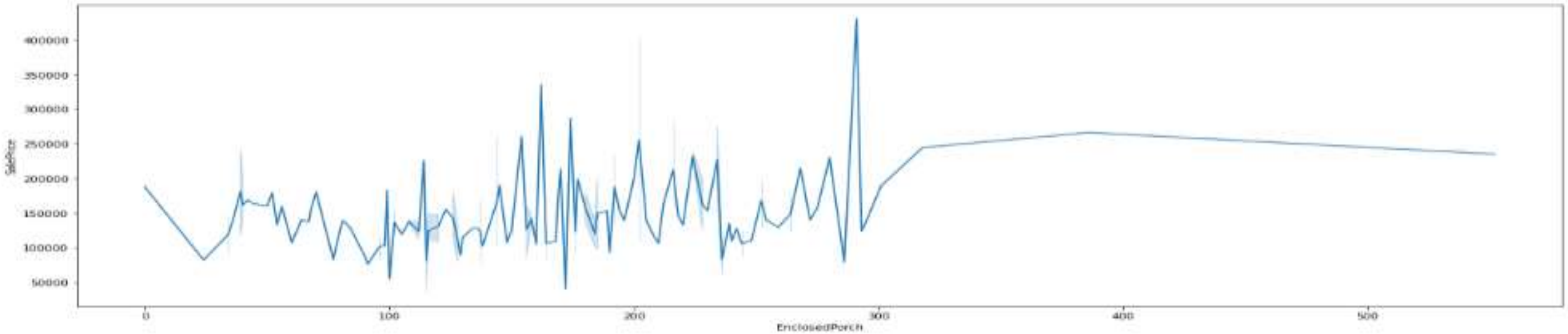

```
Observation:Feature KitchenAbvGr [Kitchens above grade] is negatively correlated to House price
```

❖ Features that negatively affect Sale Price (target variable)

3. EnclosedPorch: Enclosed porch area in square feet

```
#EnclosedPorch vs SalePrice
plt.figure(figsize=(22,7))
sns.lineplot(x="EnclosedPorch",y="SalePrice",data=dfn_train)
```

<AxesSubplot:xlabel='EnclosedPorch', ylabel='SalePrice'>

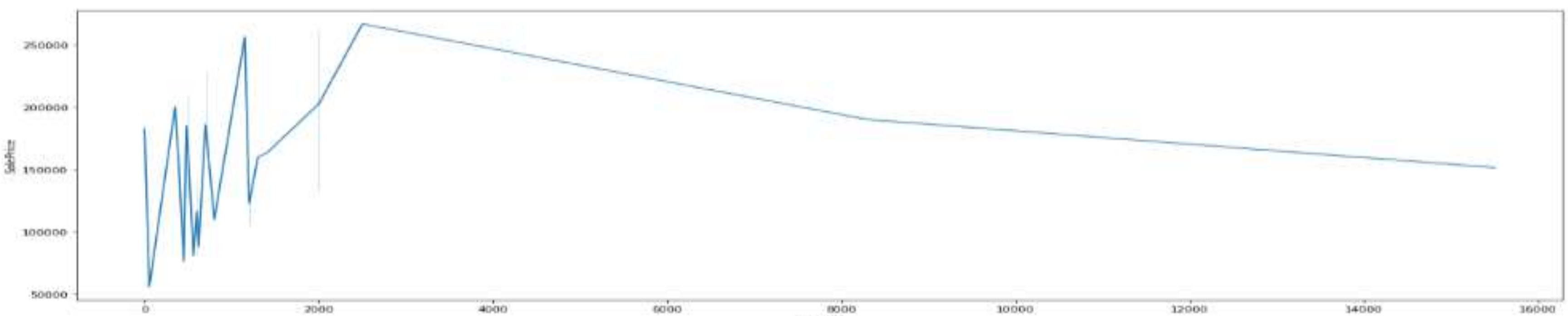



```
Observation: Feature like EnclosedPorch[Enclosed porch area in square feet] brings down the house price.
```


4. MiscVal: Value of miscellaneous feature

```
#MiscVal vs SalePrice
plt.figure(figsize=(22,7))
sns.lineplot(x="MiscVal",y="SalePrice",data=dfn_train)
```

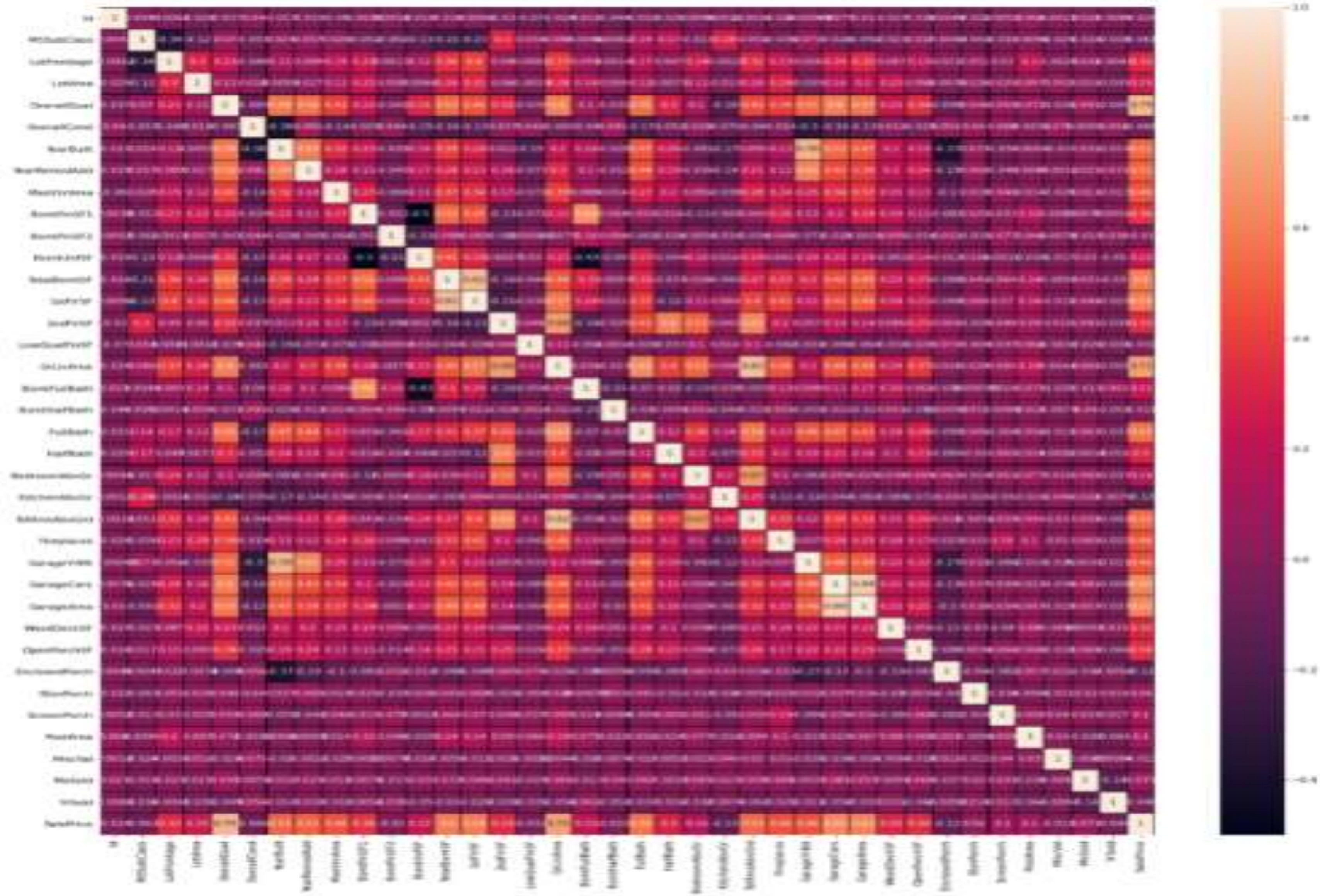
<AxesSubplot:xlabel='MiscVal', ylabel='SalePrice'>




```
Observation: Housing price go down with feature like  MiscVal [Value of miscellaneous feature]
```

❖ Numerical features that are Correlated with eachother

```
#Multi Variate Analysis
plt.figure(figsize=(22,22))
sns.heatmap(dfn_train.corr(),annot=True,linewidth=0.1,linecolor="Black")
plt.show()
```



❖ Assumption:

Data- Preprocessing ¶

```
#Assumption : Feature = "Street","Utilities","Condition2"
#99% of data is same in the feature, hence it can be dropped as there is no variance
df.drop(columns=["Street","Utilities"],inplace=True)
```

```
#Feature engineering
#YearBuilt
#Feature "YearBuilt" describes how old the house is but the values wont appropriately contribute to model building
#Assuming that the oldest house has high price, we can categorize the feature
print("Oldest house was built in year=",df.YearBuilt.min())
print("Latest house was built in year=",df.YearBuilt.max())
print("Range =",df.YearBuilt.max() - df.YearBuilt.min())

#We will divide the range into 10 intervals ie 14 years
#It means the house built in 14 years span belong to same rating

# 1872-1885: 10
# 1886-1899: 9
# 1900-1913: 8
# 1914-1927: 7
# 1928-1941: 6
# 1942-1955: 5
# 1956-1969: 4
# 1970-1993: 3
# 1984-2007: 2
# 1998-2012: 1
```



```
#We will divide the range into 10 intervals ie 14 years
#It means the house built in 14 years span belong to same rating

# 1872-1885: 10
# 1886-1899: 9
# 1900-1913: 8
# 1914-1927: 7
# 1928-1941: 6
# 1942-1955: 5
# 1956-1969: 4
# 1970-1993: 3
# 1984-2007: 2
# 1998-2012: 1
```



```
#New feature "AgeRating" extracted wrt "YearBuilt"
```

❖ Assumption:

```
#YearRemodAdd
#As per feature description - YearRemodAdd: Remodel date (same as construction date if no remodeling or additions)

#Assuming that house with remodel costs more.
#We compare feature "YearBuilt" & "YearRemodAdd" : new feature "RemodRating" can be derived
# if value are same then: RemodRating = 0     (ie No remodelling done)
# if value are not same then: RemodRating = 1 (ie Remodelling done)
```

```
# 696 out of 1460 houses are remodelled
sns.countplot(x="RemodRating",data=dfn)
```

```
<AxesSubplot:xlabel='RemodRating', ylabel='count'>
```

❖ Model Building & Cross Validation

1. Decision Tree Regressor

```
#Decision tree regressor
from sklearn.tree import DecisionTreeRegressor
dtr=DecisionTreeRegressor()
dt_r2=maxr2_score(dtr,pc_x,y)
```

```
max r2 score corresponding to  53  is  0.7258767589155597
```

```
#Cross validation (Decision tree)
cross_val(dtr,pc_x,y)
```

```
Mean r2 score for regressor:  0.5161373790747328
[0.53099171 0.55368055 0.47296049 0.52195243 0.50110172]
```

2. K Nearest Regressor

```
from sklearn.neighbors import KNeighborsRegressor
knr=KNeighborsRegressor()
r_state=maxr2_score(knr,pc_x,y)
```

```
max r2 score corresponding to  74  is  0.793195809152064
```

```
#Cross validation (KNN Classifier)
cross_val(knr,pc_x,y)
```

```
Mean r2 score for regressor:  0.6782362977260494
[0.66750537 0.68517542 0.54330341 0.7503157  0.74488159]
```

3. Random Forest Regressor

```
#Check maxr2_score function
from sklearn.ensemble import RandomForestRegressor
rfr=RandomForestRegressor()
rfr_r2=maxr2_score(rfr,pc_x,y)
```

```
max r2 score corresponding to  49  is  0.8791328630007117
```

```
#Cross validation (Random Forest)
cross_val(rfr,pc_x,y)
```

```
Mean r2 score for regressor:  0.7941464393082931
[0.78427416 0.81124034 0.70628763 0.82962562 0.83930445]
```

4. Ada-Boost Regressor

```
from sklearn.ensemble import AdaBoostRegressor
adr=AdaBoostRegressor()
r_state=maxr2_score(adr,pc_x,y)
```

```
max r2 score corresponding to  89  is  0.8154645937662629
```

```
#Cross validation (Adaboost Regressor)
cross_val(adr,pc_x,y)
```

```
Mean r2 score for regressor:  0.7228987360669794
[0.75766725 0.73012011 0.63809156 0.78773541 0.70087935]
```

❖ Finalized Model

- Random Forest is the best performing model.

Hyper Parameter Tuning & Model testing with Hyper Parameters

```
#Hyper Parameter tuning (random forest regressor)
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import GridSearchCV
rfr=RandomForestRegressor()
parameters = {"n_estimators":[10,100,500]}
rgr = GridSearchCV(rfr, parameters, cv=5)
rgr.fit(pc_x, y)
rgr.best_params_
```

```
{'n_estimators': 500}
```

```
#Random Forest with hyper parameters
# Random state= 49
# Parameter={'n_estimators': 500}

x_train,x_test,y_train,y_test=train_test_split(pc_x,y,test_size=0.2,random_state=49)
rfr=RandomForestRegressor(n_estimators=500)
rfr.fit(x_train,y_train)
y_pred=rfr.predict(x_test)
print("R2 score=",r2_score(y_test,y_pred))
print("RMSE=",np.sqrt(mean_squared_error(y_test,y_pred)))
print("Mean abs error=",mean_absolute_error(y_test,y_pred))
```

```
R2 score= 0.8745145435057429
RMSE= 27601.958695404774
Mean abs error= 19061.425854700858
```

❖ Conclusion

- Using Random Forest model the House Sales Price is predicted for test data.

Predicted values for 292 test datapoints.

1	SalePrice
2	388099.532
3	104743.7
4	375044.63
5	362707.74
6	371738.826
7	100542.616
8	102150.726
9	377217.384
10	382369.626
11	89306.412
12	89292.43
13	100201.56
14	369552.166
15	373312.864
16	383766.496
17	97435.92
18	84246.53
19	97312.11
20	88098.396
21	97318.78
22	85582.998
23	94209.156
24	82889.08
25	98764.394
26	84985.406
27	84637.544
28	83364.694
29	82106.156
30	373629.65
31	85672.104
32	100889.748
33	87289.646
34	97234.502
35	380593.562
36	87421.654
37	97795.318
38	87221.86
39	84649.378
40	363736.004
41	83732.144
42	94096.938
43	86439.138
44	370862.116
45	83567.96
46	87427.384
47	83528.246
48	83509.356
49	84902.44
50	91743.764
51	377654.808
52	372821.382
53	98151.826
54	83671.45
55	85742.554
56	85003.166
57	101694.508
58	84163.43
59	79667.458
60	100118.158
61	383040.194
62	85827.788
63	105074.788
64	97048.742
65	90723.006
66	88502.088
67	97978.726
68	361084.268
69	382753.524
70	87375.402
71	93666.246
72	370589.994
73	364312.15
74	374157.868
75	379106.064
76	89521.358
77	85661.35
78	372441.85
79	83843.016
80	394921.572
81	369713.72
82	78046.61
83	97111.962
84	372564.764
85	89114.182
86	98343.032
87	382197.998
88	363938.444
89	365337.962
90	367259.878
91	84240.612
92	98035.984
93	85660.058
94	80724.612
95	84369.544
96	376276.612
97	86766.164
98	82532.342
99	98869.696
100	89116.604
101	83784.278
102	86285.672
103	89057.318
104	84111.462
105	98976.296
106	84534.308
107	89826.836
108	386132.594
109	377406.812
110	84919.378
111	373560.93
112	383704.286
113	88351.562
114	386992.69
115	381606.808
116	85020.46
117	98646.874
118	86037.192
119	379787.088
120	365183.77
121	97119.582
122	90274.28
123	86531.926
124	371956.988
125	82876.652
126	357117.218
127	84287.552
128	96861.634
129	377193.262
130	96749.902
131	86332.558
132	96009.104
133	353627.874
134	90211.694
135	356610.394
136	381236.732
137	382207.692
138	88098.492
139	99168.996
140	378005.958
141	89364.444
142	103509.982
143	378920.976
144	102462.766
145	375141.304
146	378491.87
147	85625.776
148	83383.018
149	87684.808
150	363580.264
151	367229.716
152	86734.222
153	82004.768
154	83418.5
155	85742.436
156	82073.574
157	359894.188
158	383852.208
159	378454.828
160	84314.666
161	89531.112
162	86553.692
163	88858.286
164	85292.434
165	84245.03
166	94255.482
167	381191.89
168	99478.908
169	80182.842
170	84438.944
171	83605.874
172	85414.084
173	96847.786
174	97683.28
175	368032.512
176	90156.354
177	371760.164
178	87946.112
179	84904.036
180	96036.138
181	351397.942
182	370094.802
183	362367.478
184	375933.97
185	86503.754
186	385944.736
187	381350.712
188	377842.198
189	85166.79
190	80341.878
191	351156.35
192	83259.398
193	89157.502
194	88065.1
195	87923.984
196	83122.646
197	85942.022
198	92625.218
199	88297.556
200	83959.432
201	364079.112
202	88875.474
203	84487.152
204	375008.374
205	99977.99
206	370179.982
207	87285.794
208	87788.258
209	372430.372
210	384966.638
211	84659.746
212	91012.666
213	84317.348
214	95352.584
215	85499.744
216	82926.63
217	90461.518
218	89319.626
219	378900.79
220	86542.896
221	382643.254
222	86609.758
223	84245.456
224	96173.988
225	83299.632
226	86568.714
227	384052.62
228	368493.898
229	86738.93
230	383571.104
231	83388.096
232	81905.652
233	358616.304
234	85623.436
235	87243.002
236	86035.774
237	395314.452
238	80203.836
239	366162.454
240	378561.302
241	85851.442
242	368964.634
243	87699.45
244	374574.108
245	362922.27
246	85849.472
247	88458.046
248	88610.388
249	375955.904
250	362887.95
251	378874.434
252	380118.946
253	382747.402
254	82500.598
255	97496.58
256	82839.802
257	368681.028
258	86786.7
259	89398.728
260	381102.998
261	95430.5
262	359316.302
263	355261.582
264	96857.264
265	87350.756
266	379583.728
267	88710.426
268	360572.872
269	375546.058
270	90558.348
271	88520.158
272	374607.556
273	96317.252
274	384376.624
275	362419.62
276	388333.702
277	91097.928
278	84362.284
279	100728.57
280	99783.916
281	378777.676
282	95544.344
283	89456.608
284	359148.882
285	386810.992
286	92663.166
287	87092.032
288	84712.534
289	86758.87
290	86180.452
291	369959.036
292	88798.218
293	87402.224

Thank You