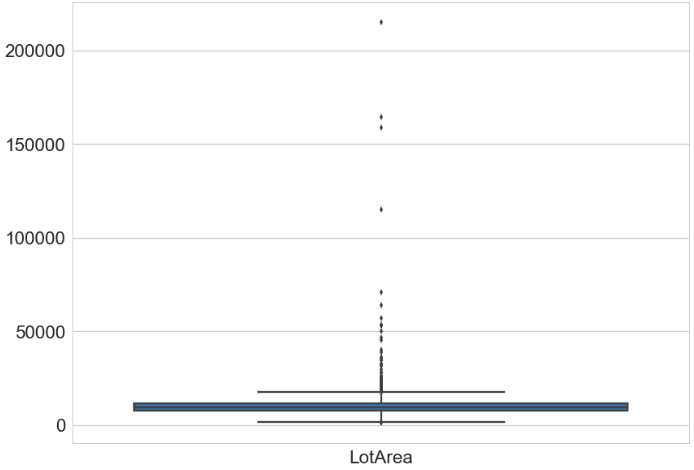
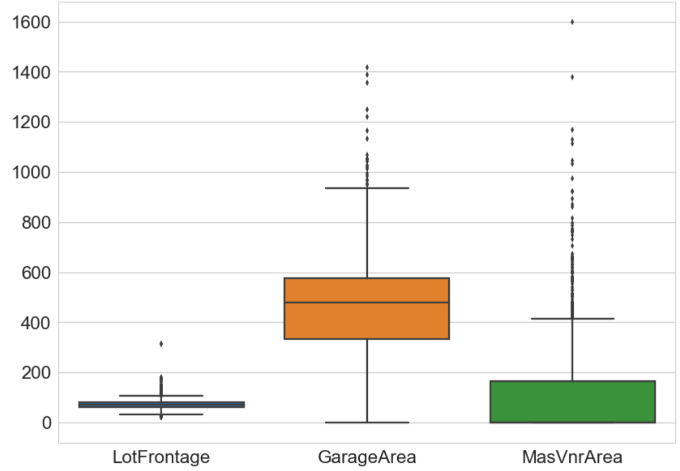
RMBI3110 Assignment 1

Li Chung Yat
20422979
7th October,2018

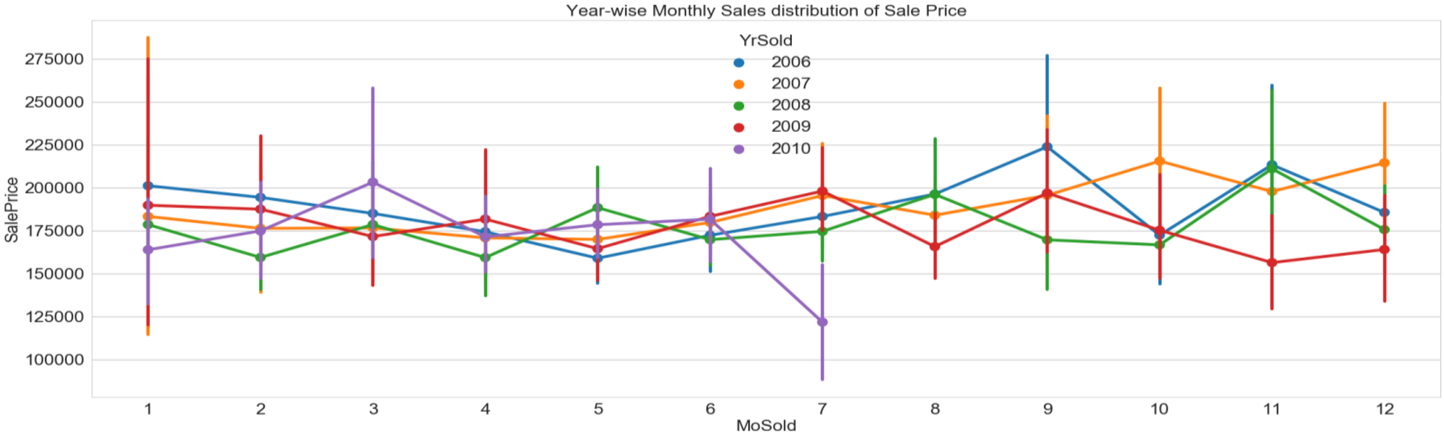
Exploratory data analysis. Please use summary statistic, bar chart, histogram, side by side boxplot, violin plot, scatter plot matrix, correlation matrix, and bubble plot to explore the data. You can choose the variables that you are interested in to illustrate the graph. Please explain your findings.
Finding outliers with Box-plot,
With original data,



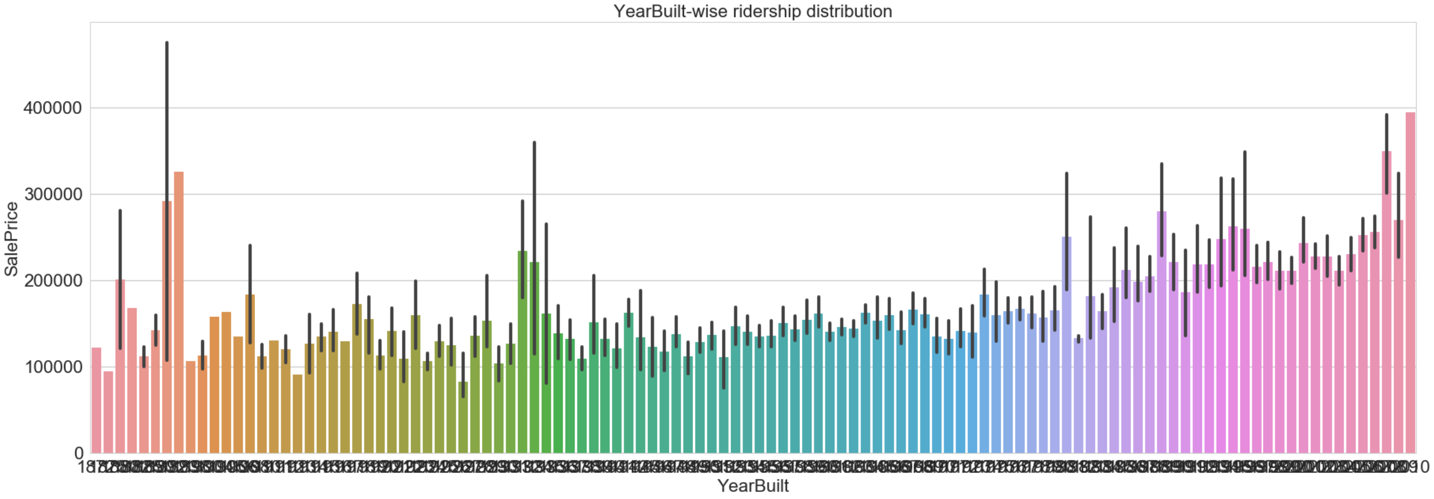


There are a number of outliers in LotArea predictor and their value are extremely large. I decided to drop outliers > 40000. I also drop outliers that have LotFrontage > 200 and MasVnrArea > 1000 so that I can strike a balance between error reduction and number of data dropped.

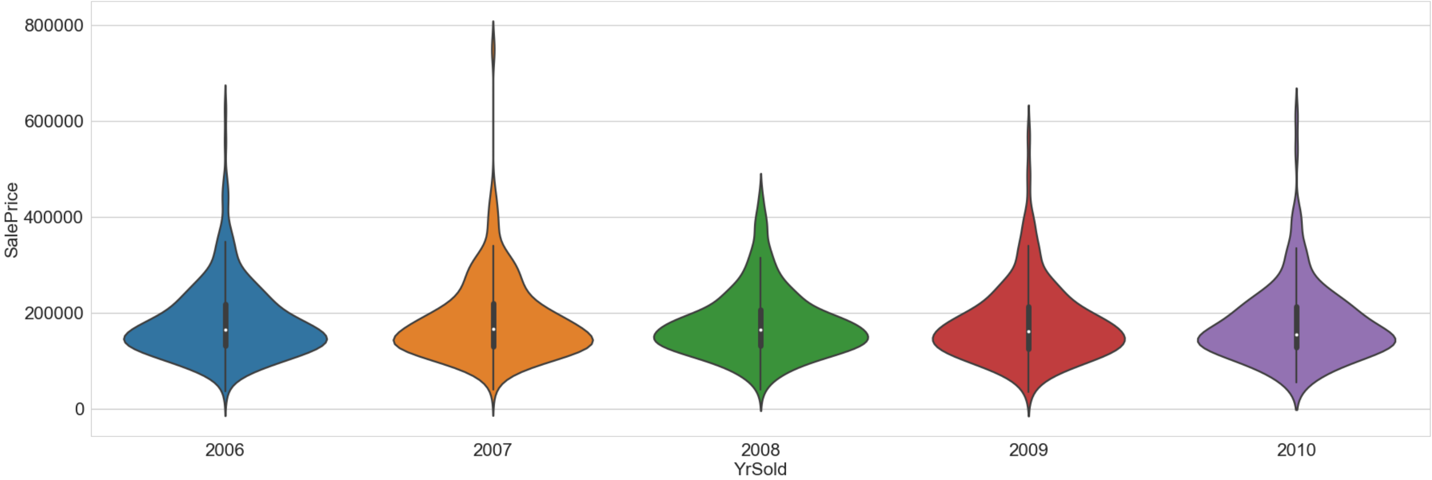
I decide to plot MoSold and YrSold attributes to see if there are monthly or yearly trend for sale price,



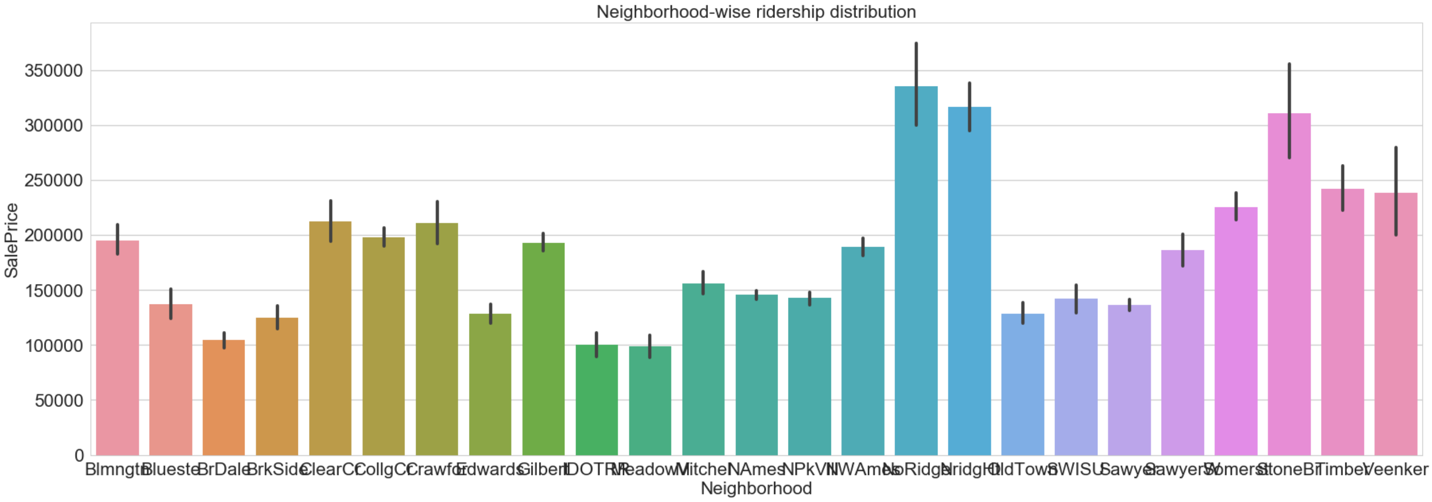
However, there are no significant change of price across years and specific trends can be observed.



By plotting YearBuilt against SalePrice, more newly built houses tend to have higher price. But there are also houses which was built long time ago price very high. By studying these houses' YearRemodAdd and LotArea attribute, they are all remodelled in 2000s and have large LotArea.



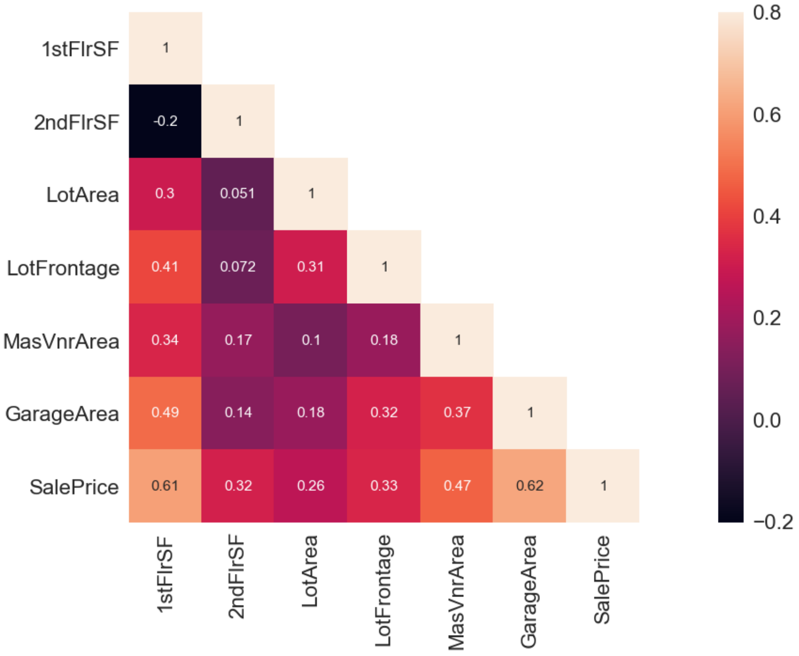
The Violin plot gives a clearer picture on the variation of sale price across 2006 to 2010, there are not much deviation. House price has always been right-skewed and mean sale price is around USD179886.



By plotting the 25 neighbourhoods against houses sale price. NoRidge,NridgHt and StoneBr are the three neighbourhoods' house priced highest. Edwards, MeadowV and IDORTT are the three price lowest.

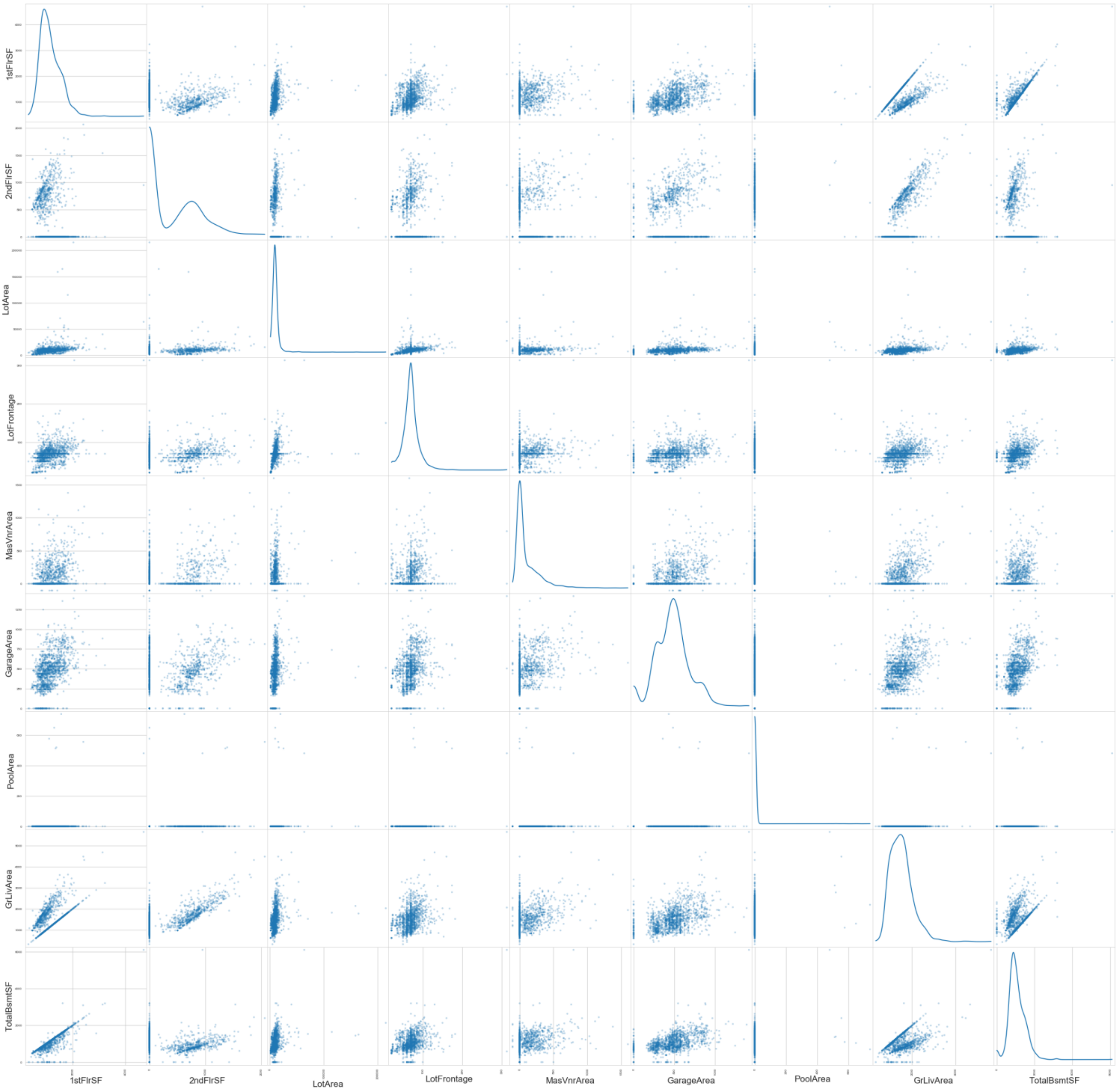
This is the correlation matrix of area related attributes.

Sale Price all have positive correlation to most of area related attributes. And in between the area attributes, some of them also has positive correlation as well. I believe some of them will be screened in VIF testing due to high multicollinearity should be high among these attributes.

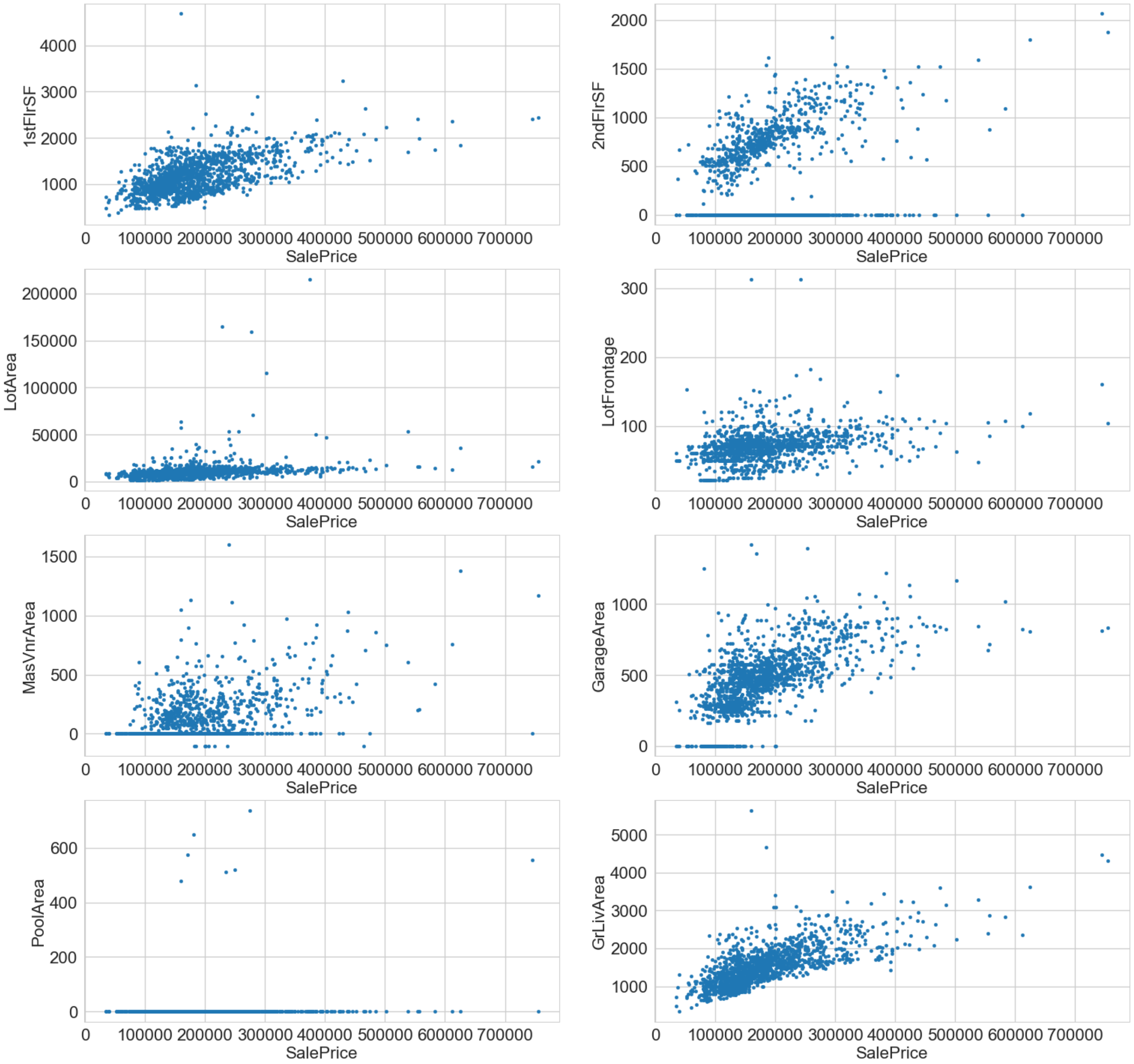


To further study correlations between different area of structures, a scatter plot matrix is plot to observe the linearity or underlying relationships between area related attributes. Lot Area is very independent with all these attributes but first floor area and second floor seems to have strong linearity relationship with total basement area of the house and living area of the house. The observations make a lot of sense because living area has to be dependent with the area the house has above the ground. There is no surprise that pool area cannot provide any informative observation as the data in PoolQu is too little. Almost all of the attributes are right skewed and implies there are lots of outliers included. However, this is a normal phenomenon as most of the apartments are in between 500 square feet to 3000 square feet, but this data set include big houses or villas that have multiple floors or even a swimming pool. However, I decided not to drop all of these outliers as these observations can fit quite well as it make sense that large house will cost more and vice versa.

Following is the scatter plot matrix for first floor area, second floor area, lot area, lot front age, masonry veneer area, garage area, pool area, living area and basement area predictors,



There must be relations between rating, area and facilities with sales price of a house. By using scatter plot, I observed that 1stFlrSF, 2ndFlrSF, GirLivArea and GarageArea seems to have positive relation with houses' price. Most of the data in PoolArea is 0, which implies that most of the house sold in dataset does not have a swimming pool. With large number in LotArea and LotFrontArea, in fact do not imply that it worth more. This has raised my attention to further study these data points, I decided to bring these data back in regression for further study.



Overall F-test on training data set after significance testing 1c) :

$$F(model) = \frac{\frac{SSR}{k}}{\frac{SSE}{n-k-1}}$$

F statistics = 68.5

Null hypothesis is rejected implies that at least one predictor is useful to perform predictions on house price.

Data cleaning

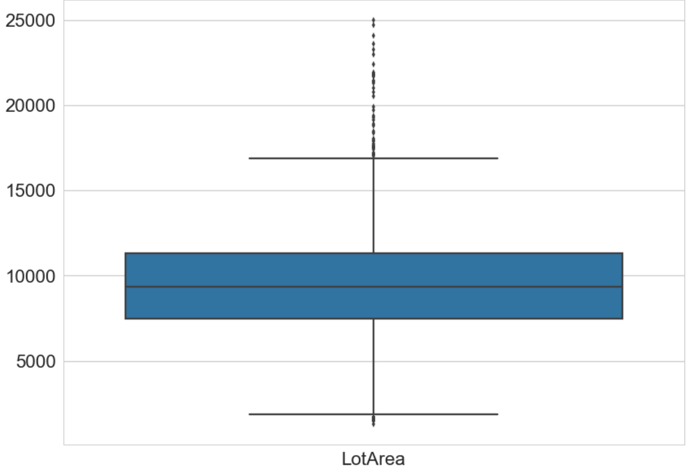
a) Check whether we have missing values. Indicate the method you deal with it.

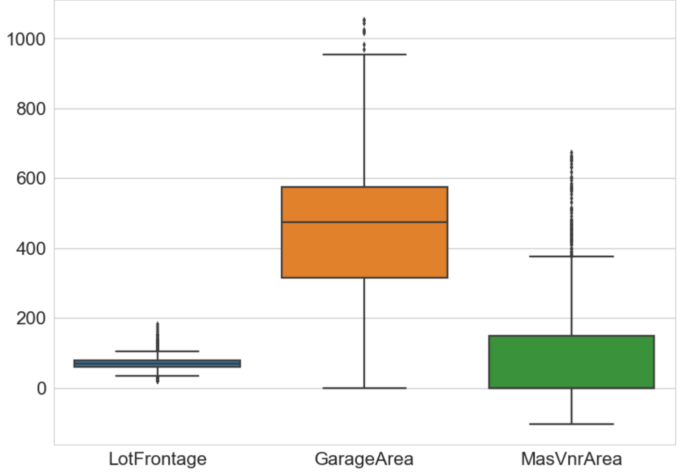
There are scattered nan value in LotFrontage, by replacing mean value of column LotFrontage, I can keep more records in database to give precise prediction. For attributes PoolQC, Fence and MiscFeature, the nan value imply that the house does not have these facilities. Therefore, I did not remove any value but instead, I replace nan by NoPool, NoFence, NoMiscFeature etc. to indicate that these nan is not missing value but a meaningful category in these categorical variables.
For GarageYrBlt and MasVnrArea these kind of numerical data missing value, they are meaningful nan because they indicate those houses do not contain a garage or masonry veneer. I replace these value by the negative of mean value in these attribute to indicate they do not have these structures and emphasise the difference of having these structures and without these structures.
For column Utilities, all transactions have the same value AllPub. It has not prediction power or meaning in the data set. The column is dropped to reduce predictors to be determined in the regression model. It automatically screened in the low variance checking procedure.

b) Check whether we have outliers. Indicate the method you deal with it.

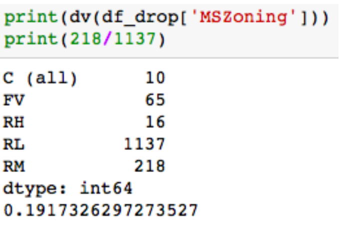
Outliers in LotArea are extremely large, by using .drop function in pandas dataframe, I remove transactions that contain large value in LotArea. I originally decided to drop Lotfrontage and MasVnrArea's outliers based on the box-plot. However, after studying the scatter-plot, I decided to keep some of these records. It is because there are obvious relationship can be observed and these data can be useful to judge whether the predictors they belonged to are useful or not.

After cleansing,



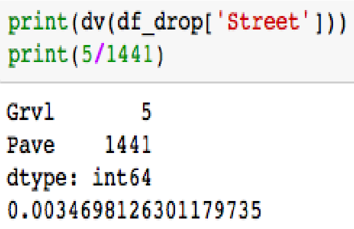


c) Predictors with very low variance offer little predictive power to models. Please find the ratio of the second most frequent value to the most frequent value for each predictor, and to remove variables where this ratio was less than 0.05. This only translates to dropping variables where 95% or more of the values are the same.



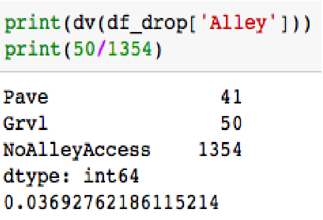
```
print(dv(df_drop['MSZoning']))
print(218/1137)

C (all)      10
FV           65
RH           16
RL         1137
RM          218
dtype: int64
0.1917326297273527
```



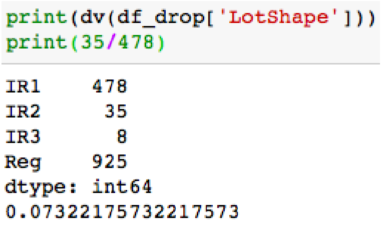
```
print(dv(df_drop['Street']))
print(5/1441)

Grvl       5
Pave    1441
dtype: int64
0.0034698126301179735
```



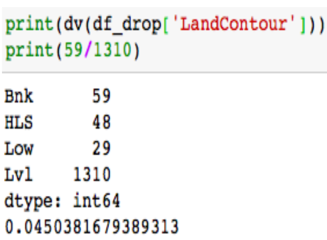
```
print(dv(df_drop['Alley']))
print(50/1354)

Pave                 41
Grvl                 50
NoAlleyAccess      1354
dtype: int64
0.03692762186115214
```



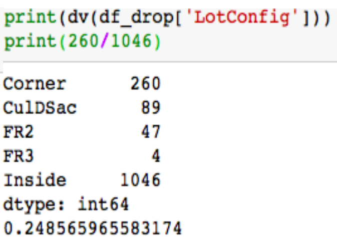
```
print(dv(df_drop['LotShape']))
print(35/478)

IR1     478
IR2      35
IR3       8
Reg     925
dtype: int64
0.07322175732217573
```



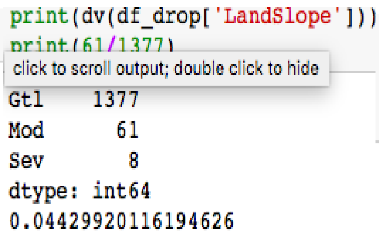
```
print(dv(df_drop['LandContour']))
print(59/1310)

Bnk      59
HLS      48
Low      29
Lvl    1310
dtype: int64
0.0450381679389313
```



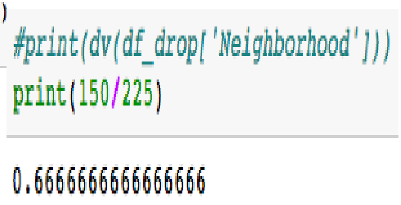
```
print(dv(df_drop['LotConfig']))
print(260/1046)

Corner      260
CulDSac      89
FR2          47
FR3           4
Inside     1046
dtype: int64
0.248565965583174
```



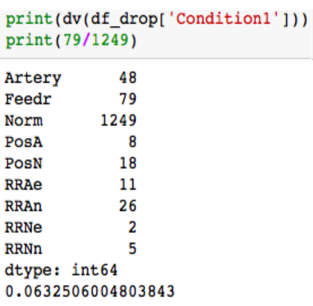
```
print(dv(df_drop['LandSlope']))
print(61/1377)

Gtl    1377
Mod      61
Sev       8
dtype: int64
0.04429920116194626
```



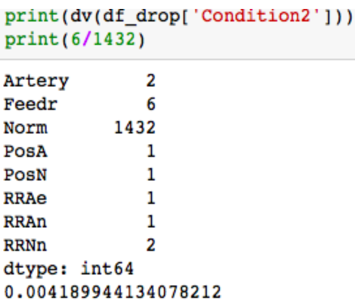
```
#print(dv(df_drop['Neighborhood']))
print(150/225)

0.6666666666666666
```



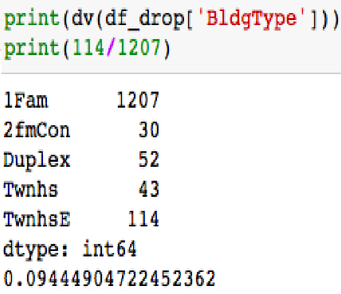
```
print(dv(df_drop['Condition1']))
print(79/1249)

Artery      48
Feedr       79
Norm      1249
PosA         8
PosN        18
RRAe        11
RRAn        26
RRNe         2
RRNn         5
dtype: int64
0.0632506004803843
```



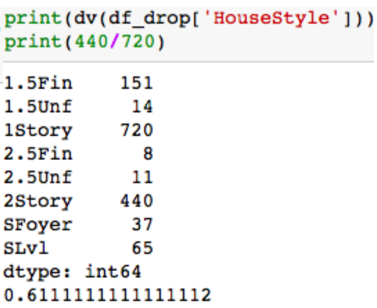
```
print(dv(df_drop['Condition2']))
print(6/1432)

Artery       2
Feedr        6
Norm      1432
PosA         1
PosN         1
RRAe         1
RRAn         1
RRNn         2
dtype: int64
0.004189944134078212
```



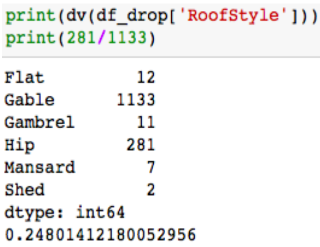
```
print(dv(df_drop['BldgType']))
print(114/1207)

1Fam      1207
2fmCon      30
Duplex      52
Twnhs       43
TwnhsE     114
dtype: int64
0.09444904722452362
```



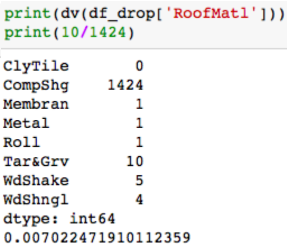
```
print(dv(df_drop['HouseStyle']))
print(440/720)

1.5Fin    151
1.5Unf     14
1Story    720
2.5Fin      8
2.5Unf     11
2Story    440
SFoyer     37
SLvl       65
dtype: int64
0.6111111111111112
```



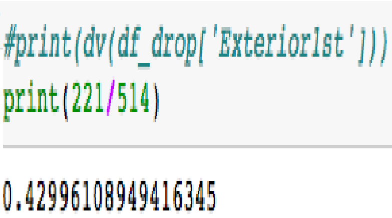
```
print(dv(df_drop['RoofStyle']))
print(281/1133)

Flat         12
Gable      1133
Gambrel      11
Hip         281
Mansard       7
Shed          2
dtype: int64
0.24801412180052956
```



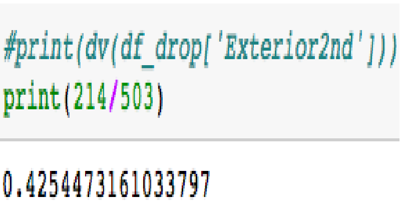
```
print(dv(df_drop['RoofMatl']))
print(10/1424)

ClyTile        0
CompShg     1424
Membran        1
Metal          1
Roll           1
Tar&Grv       10
WdShake        5
WdShngl        4
dtype: int64
0.007022471910112359
```



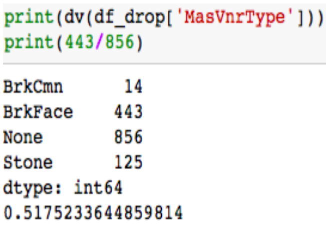
```
#print(dv(df_drop['Exterior1st']))
print(221/514)

0.42996108949416345
```



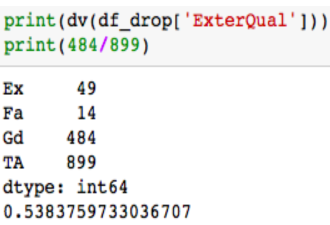
```
#print(dv(df_drop['Exterior2nd']))
print(214/503)

0.4254473161033797
```



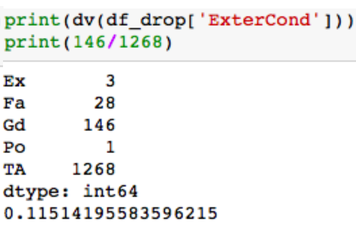
```
print(dv(df_drop['MasVnrType']))
print(443/856)

BrkCmn      14
BrkFace    443
None       856
Stone      125
dtype: int64
0.5175233644859814
```



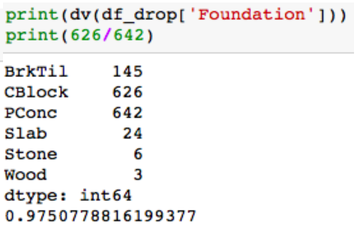
```
print(dv(df_drop['ExterQual']))
print(484/899)

Ex     49
Fa     14
Gd    484
TA    899
dtype: int64
0.5383759733036707
```



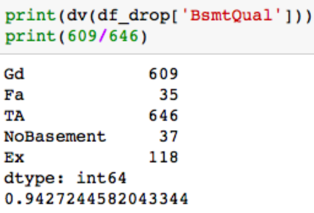
```
print(dv(df_drop['ExterCond']))
print(146/1268)

Ex       3
Fa      28
Gd     146
Po       1
TA    1268
dtype: int64
0.11514195583596215
```



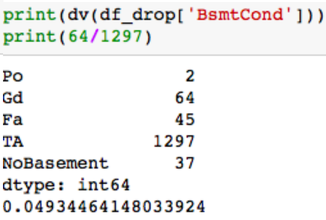
```
print(dv(df_drop['Foundation']))
print(626/642)

BrkTil    145
CBlock    626
PConc     642
Slab       24
Stone       6
Wood        3
dtype: int64
0.9750778816199377
```



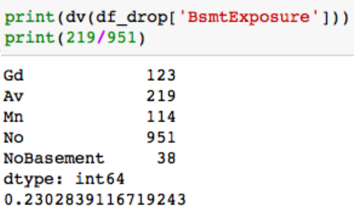
```
print(dv(df_drop['BsmtQual']))
print(609/646)

Gd            609
Fa             35
TA            646
NoBasement     37
Ex            118
dtype: int64
0.9427244582043344
```



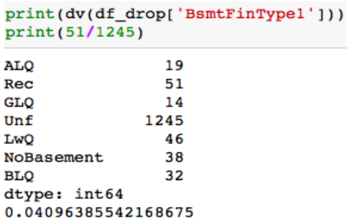
```
print(dv(df_drop['BsmtCond']))
print(64/1297)

Po               2
Gd              64
Fa              45
TA            1297
NoBasement      37
dtype: int64
0.04934464148033924
```



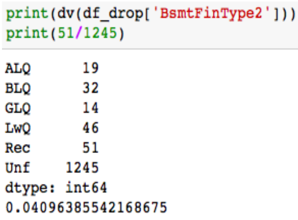
```
print(dv(df_drop['BsmtExposure']))
print(219/951)

Gd            123
Av            219
Mn            114
No            951
NoBasement     38
dtype: int64
0.2302839116719243
```



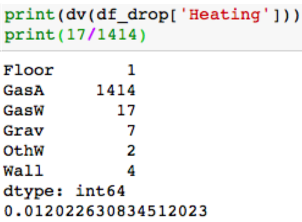
```
print(dv(df_drop['BsmtFinType1']))
print(51/1245)

ALQ             19
Rec             51
GLQ             14
Unf           1245
LwQ             46
NoBasement      38
BLQ             32
dtype: int64
0.04096385542168675
```



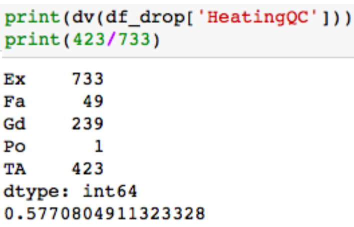
```
print(dv(df_drop['BsmtFinType2']))
print(51/1245)

ALQ      19
BLQ      32
GLQ      14
LwQ      46
Rec      51
Unf    1245
dtype: int64
0.04096385542168675
```



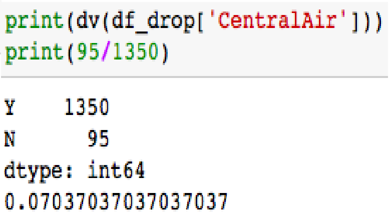
```
print(dv(df_drop['Heating']))
print(17/1414)

Floor       1
GasA     1414
GasW       17
Grav        7
OthW        2
Wall        4
dtype: int64
0.012022630834512023
```



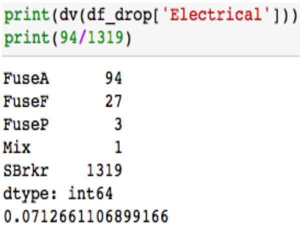
```
print(dv(df_drop['HeatingQC']))
print(423/733)

Ex    733
Fa     49
Gd    239
Po      1
TA    423
dtype: int64
0.5770804911323328
```



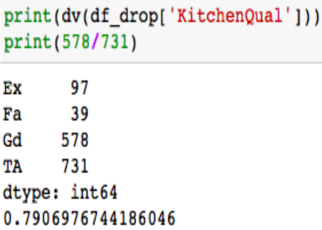
```
print(dv(df_drop['CentralAir']))
print(95/1350)

Y    1350
N      95
dtype: int64
0.07037037037037037
```



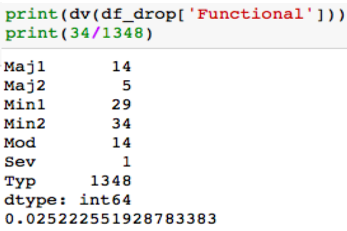
```
print(dv(df_drop['Electrical']))
print(94/1319)

FuseA      94
FuseF      27
FuseP       3
Mix         1
SBrkr    1319
dtype: int64
0.0712661106899166
```



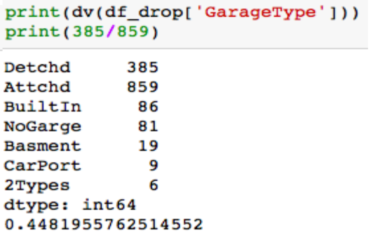
```
print(dv(df_drop['KitchenQual']))
print(578/731)

Ex     97
Fa     39
Gd    578
TA    731
dtype: int64
0.7906976744186046
```



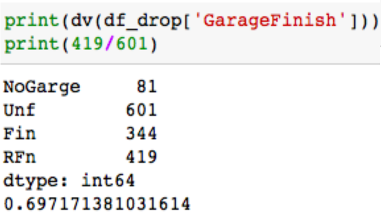
```
print(dv(df_drop['Functional']))
print(34/1348)

Maj1      14
Maj2       5
Min1      29
Min2      34
Mod       14
Sev        1
Typ     1348
dtype: int64
0.025222551928783383
```



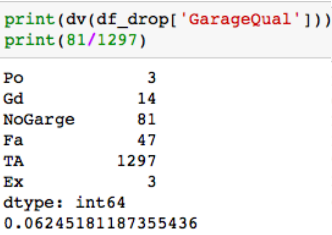
```
print(dv(df_drop['GarageType']))
print(385/859)

Detchd     385
Attchd     859
BuiltIn     86
NoGarge     81
Basment     19
CarPort      9
2Types       6
dtype: int64
0.4481955762514552
```



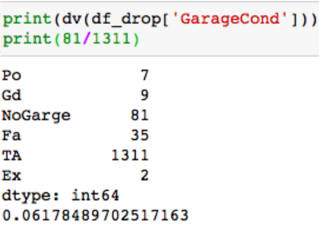
```
print(dv(df_drop['GarageFinish']))
print(419/601)

NoGarge     81
Unf        601
Fin        344
RFn        419
dtype: int64
0.697171381031614
```



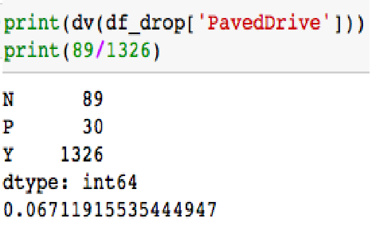
```
print(dv(df_drop['GarageQual']))
print(81/1297)

Po            3
Gd           14
NoGarge      81
Fa           47
TA         1297
Ex            3
dtype: int64
0.06245181187355436
```



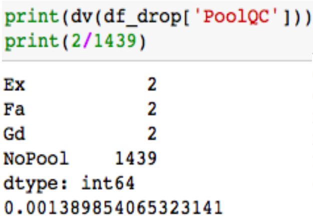
```
print(dv(df_drop['GarageCond']))
print(81/1311)

Po            7
Gd            9
NoGarge      81
Fa           35
TA         1311
Ex            2
dtype: int64
0.06178489702517163
```



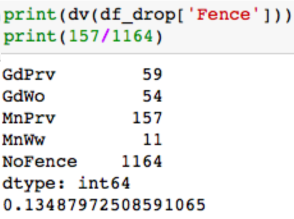
```
print(dv(df_drop['PavedDrive']))
print(89/1326)

N      89
P      30
Y    1326
dtype: int64
0.06711915535444947
```



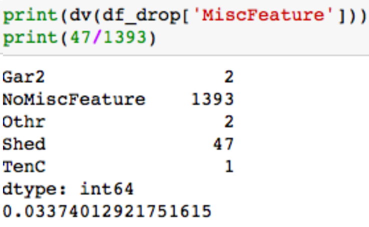
```
print(dv(df_drop['PoolQC']))
print(2/1439)

Ex           2
Fa           2
Gd           2
NoPool    1439
dtype: int64
0.001389854065323141
```



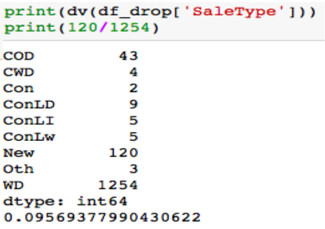
```
print(dv(df_drop['Fence']))
print(157/1164)

GdPrv         59
GdWo          54
MnPrv        157
MnWw          11
NoFence     1164
dtype: int64
0.13487972508591065
```



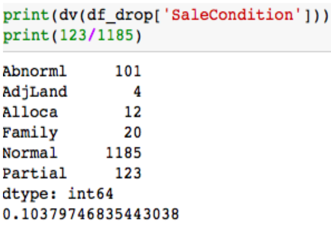
```
print(dv(df_drop['MiscFeature']))
print(47/1393)

Gar2                2
NoMiscFeature    1393
Othr                2
Shed               47
TenC                1
dtype: int64
0.03374012921751615
```



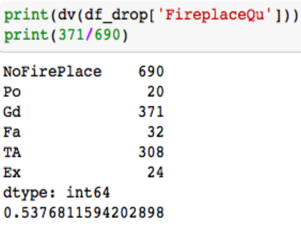
```
print(dv(df_drop['SaleType']))
print(120/1254)

COD        43
CWD         4
Con         2
ConLD       9
ConLI       5
ConLw       5
New       120
Oth         3
WD       1254
dtype: int64
0.09569377990430622
```



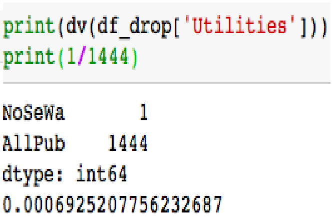
```
print(dv(df_drop['SaleCondition']))
print(123/1185)

Abnorml     101
AdjLand       4
Alloca       12
Family       20
Normal     1185
Partial     123
dtype: int64
0.10379746835443038
```



```
print(dv(df_drop['FireplaceQu']))
print(371/690)

NoFirePlace    690
Po              20
Gd             371
Fa              32
TA             308
Ex              24
dtype: int64
0.5376811594202898
```



```
print(dv(df_drop['Utilities']))
print(1/1444)

NoSeWa       1
AllPub    1444
dtype: int64
0.0006925207756232687
```

By definition, ratio = categorical variables' most frequent value over second most frequent value. Whereas, the alpha (rejection threshold is 0.05). If ratio < alpha, the variables will be rejected.
'Street', 'Alley', 'LandContour', 'LandSlope', 'Condition2', 'RootMat1', 'BsmtCond', 'BsmtFinType1', 'BsmtFinType2', 'Heating', 'Functional', 'PoolQC', 'MiscFeature' and 'Utilities' are dropped.

d) Transform the variables if necessary.

I have transform all categorical variables into indicator variables by using get_dummies function and dropping the first column of dummies to avoid

For year related predictors, I centralized all these value because 1999 and 2008 have no actual meaning. But how new or old the building is, relative to the sample, will have more interpretations or meanings behind. What is more, I standardize all numeric variables that have comparatively large range compare to other attributes. I managed to use min-max standardizing to limit most of the numerical variable ranged between 1 and -1, but leaving integer value range less than 10 with no change.

After all of these, I split my data set to a training data set and a testing data set randomly in 70-30 proportion. This step is essential as I have to compare the mean square error of prediction of multiple linear regression model against the multiple regression tree model.

Model building

a) Please use Multiple linear regression to fit the training set.

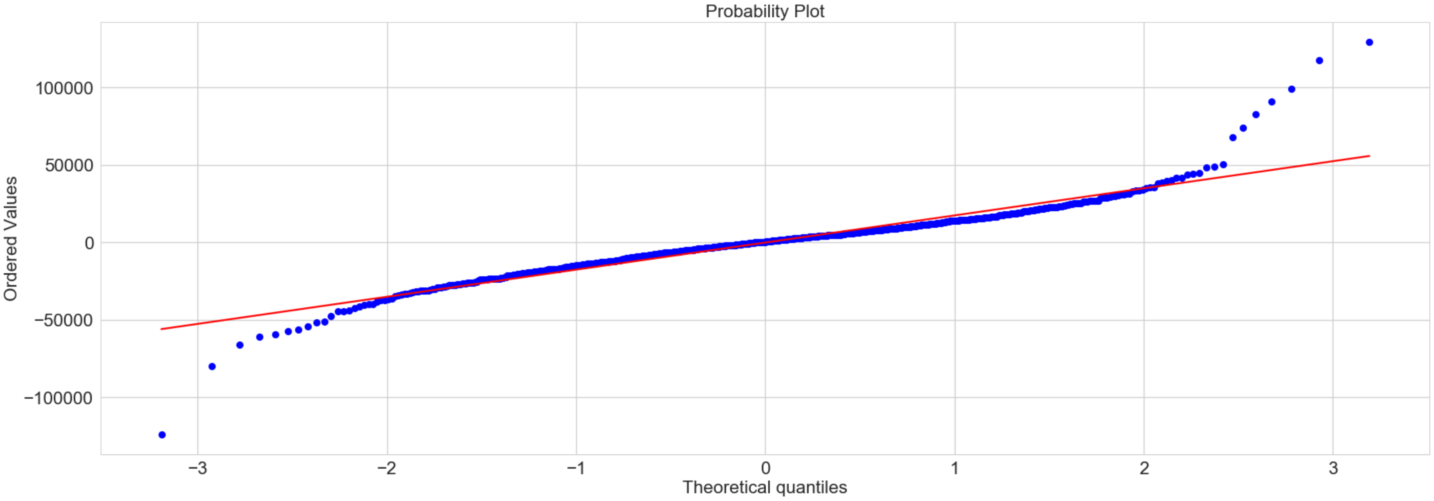
I. Use AIC to perform variable selection.

I first fit the previous data and their relative SalePrice to a OLS regression model. By carrying out significant testing, predictors with p-value higher than 0.05 and VIF higher than 10 are dropped. X_train3 is remained with only 33 variables left. Then, using this data set to perform step-wise variables selection method associated with their AIC.

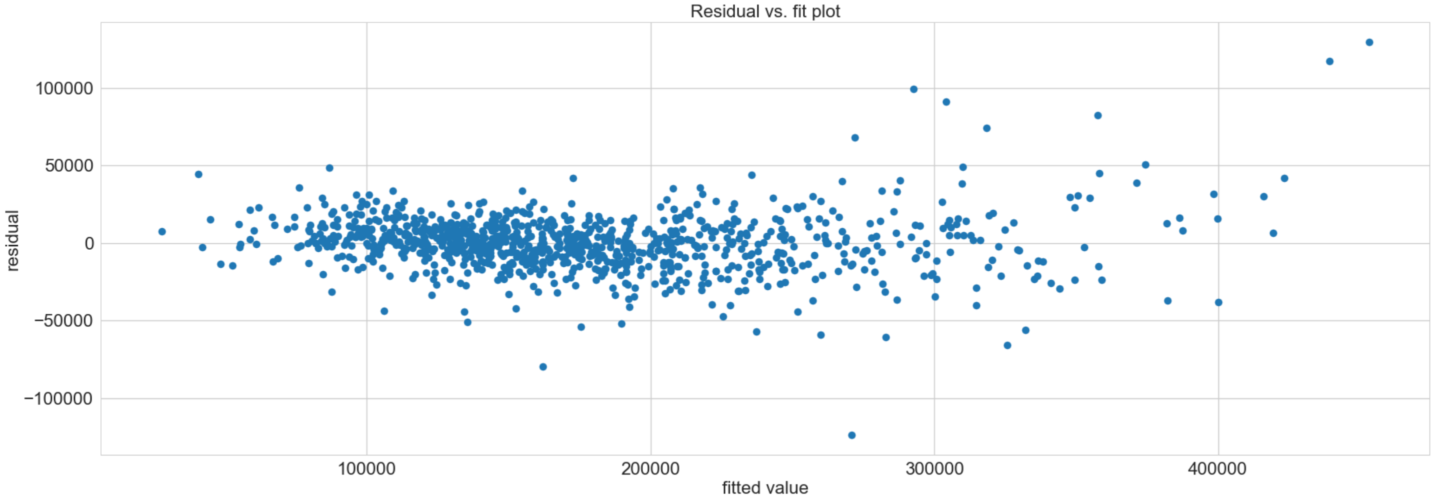
```
['Foundation_PConc', 'LotArea', 'BsmtFinSF1', 'BsmtUnfSF', '2ndFlrSF', 'SaleType_New
', 'BsmtQual_TA', 'BsmtQual_Gd', 'BsmtQual_Fa', 'ExterQual_Gd', 'BsmtExposure_Gd', 'H
ip', 'PoolArea', 'StoneBr', 'NoRidge', 'Edwards', 'MSZoning_RM', 'Mitchel', 'WoodDeck
SF', 'KitchenQual_Fa', 'SaleCondition_Normal', 'Condition1_RRAe', 'Timber', 'Fence_Mn
Prv', 'Condition1_Norm', 'BsmtFullBath', 'Mansard']
```

These are the variables that generate the best fit result in which minimise the AIC and in suitable complexity.

II. Check for model assumption and multicollinearity. Interpret the output.

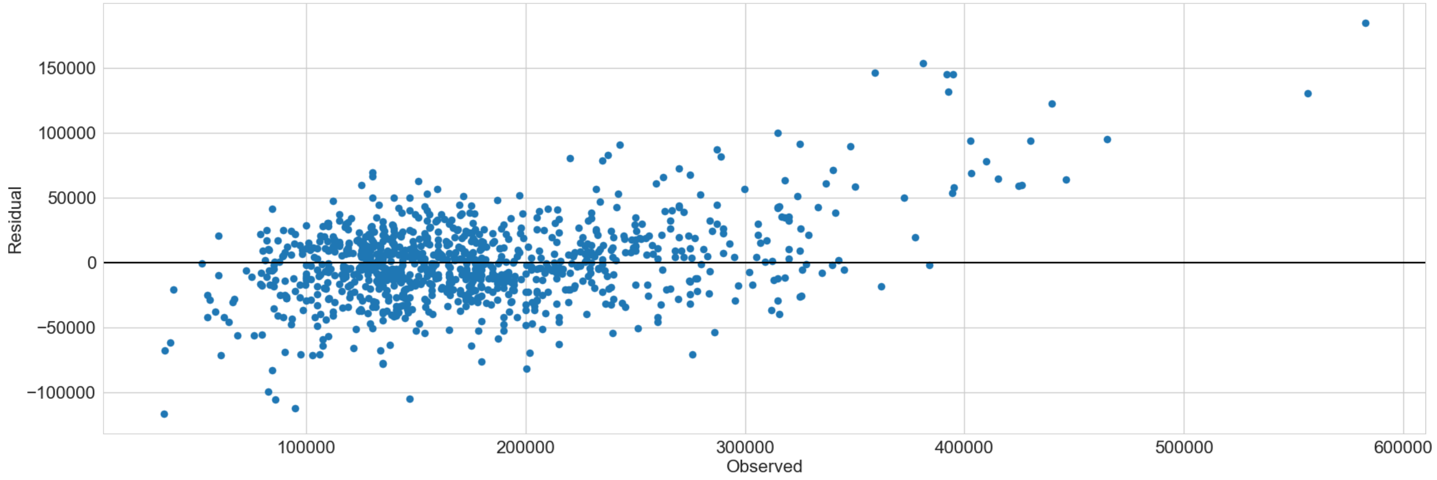


Above is the normality test, performing Q-Q test using the residuals. Majority of data lie on the normal line, the normality assumption is not violated.



There are few outliers but majority of residuals evenly spread above and below residual = 0. Residuals are believed to cancel out each other such that E[residual] will be 0. Thus, mean-zero assumption is not violated.

No significant trend can be observed from the plot, meaning residuals will not variate as fitted value changes. Constant variance assumption is valid.



The residual-observation plot is to check independence of data, most of the residuals fall within the 95% confidence interval of around mean = 0. The independence assumption is not violated.
Check the VIF of variables again, all below VIF = 10 threshold, meaning no multicollinearity exist.

III. Indicate your final model for explanation objective.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      0   R-squared:                       0.822
Model:                            OLS   Adj. R-squared:                  0.817
Method:                 Least Squares   F-statistic:                     157.7
Date:                Mon, 15 Oct 2018   Prob (F-statistic):               0.00
Time:                        00:52:46   Log-Likelihood:                -11533.
No. Observations:                 983   AIC:                         2.312e+04
Df Residuals:                     954   BIC:                         2.327e+04
                    Df Model:                          28
                    Covariance Type:            nonrobust
===================================================================================
                coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
Foundation_PConc  1.351e+04   3210.612      4.208      0.000    7210.768    1.98e+04
LotArea           7.961e+04   7996.747      9.955      0.000    6.39e+04    9.53e+04
BsmtFinSF1        1.483e+05   8291.580     17.892      0.000    1.32e+05    1.65e+05
BsmtUnfSF         1.226e+05   7033.369     17.435      0.000    1.09e+05    1.36e+05
2ndFlrSF          1.043e+05   5617.027     18.569      0.000    9.33e+04    1.15e+05
KitchenQual_TA   -6.068e+04   5596.015    -10.844      0.000   -7.17e+04   -4.97e+04
MasVnrType_Stone  2.048e+04   3822.184      5.358      0.000      1.3e+04     2.8e+04
StoneBr           4.511e+04   9667.412      4.667      0.000    2.61e+04    6.41e+04
KitchenQual_Fa   -7.674e+04   8014.995     -9.574      0.000   -9.25e+04    -6.1e+04
KitchenQual_Gd   -3.999e+04   5193.477     -7.700      0.000   -5.02e+04   -2.98e+04
BsmtQual_TA       -3.04e+04   3709.151     -8.195      0.000   -3.77e+04   -2.31e+04
BsmtQual_Gd      -2.873e+04   3552.686     -8.088      0.000   -3.57e+04   -2.18e+04
BsmtExposure_Gd   2.086e+04   3995.267      5.221      0.000      1.3e+04    2.87e+04
Edwards          -2.689e+04   4257.588     -6.315      0.000   -3.52e+04   -1.85e+04
MSZoning_RM      -1.506e+04   3341.646     -4.508      0.000   -2.16e+04   -8506.289
WoodDeckSF        2.352e+04   6496.211      3.620      0.000    1.08e+04    3.63e+04
ScreenPorch       3.277e+04   8812.493      3.718      0.000    1.55e+04    5.01e+04
OpenPorchSF       3.126e+04   9002.865      3.472      0.001    1.36e+04    4.89e+04
SaleType_CWD      6.323e+04    2.18e+04      2.896      0.004    2.04e+04    1.06e+05
Condition1_Norm   7657.2725   2929.937      2.613      0.009    1907.407    1.34e+04
Mitchel          -2.055e+04   5947.045     -3.456      0.001   -3.22e+04   -8880.173
PoolArea          5.184e+04    2.04e+04      2.542      0.011    1.18e+04    9.19e+04
Condition1_RRAe  -3.224e+04    1.21e+04     -2.673      0.008   -5.59e+04   -8572.649
NAmes            -8177.0015   3250.508     -2.516      0.012   -1.46e+04   -1798.030
BsmtFullBath      5523.5027   2634.450      2.097      0.036     353.517    1.07e+04
Gilbert          -9785.7148   4913.363     -1.992      0.047   -1.94e+04    -143.466
MSZoning_RH      -1.816e+04    1.04e+04     -1.746      0.081   -3.86e+04    2256.621
ExterQual_Gd      5134.4297   3184.664      1.612      0.107   -1115.327    1.14e+04
const             1.245e+05   7725.764     16.114      0.000    1.09e+05     1.4e+05
==============================================================================
Omnibus:                      169.969   Durbin-Watson:                   1.966
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              695.464
Skew:                           0.760   Prob(JB):                    9.59e-152
Kurtosis:                       6.830   Cond. No.                         41.7
==============================================================================
```

b) Can we use Poisson regression or Negative Binomial regression to predict housing price? Why or why not?

As our dataset are normally distributed and independent, it is not suitable to use Poisson regression or Negative Binomial regression to carry out the prediction of housing price. There are very few counting related variable, only OverallQual, OverallCond, BsmtFullBath, BsmtHalfBath, FullBath, HalfBath, BedroomAbvGr, KitchenAbvGr, Fireplaces, GarageCars and MoSold are counting data whereas their significant are low, most of these counting variables are dropped in p-value or VIF test. Thus, it is not suitable to use Poisson or Negative Binomial regression to perform forecasting.

c) Use regression tree to fit the training set. Use cost-complexity to prune the tree.

To limit the complexity, maximum depth of regression tree should be limited to a certain threshold n. By performing alpha-fold Cross Validation, I seek for the number alpha for performing the cost complexity analysis of my regression tree.

Alpha	5	6	7	8	9	10	11	12	13
Avg MSE	660531997	606642888	611884920	608074196	607497381	610612504	603188921	616130835	614551915
Alpha	14	15	16	17	18	19	20	21	22
Avg MSE	598388775	582757969	589365377	593289459	604455141	619151330	620473858	618413071	610073393
Alpha	23	24	25	30	35	40	50	100	200
Avg MSE	609940423	624922443	617383223	583387835	593987397	613528742	615556981	607014542	610776422

15-folds cross-validation minimise the Cross-Validation Error which is the average MSE for each Cross-Validation model in an alpha-folds splits and a depth of 15 will generate the best sub-tree.

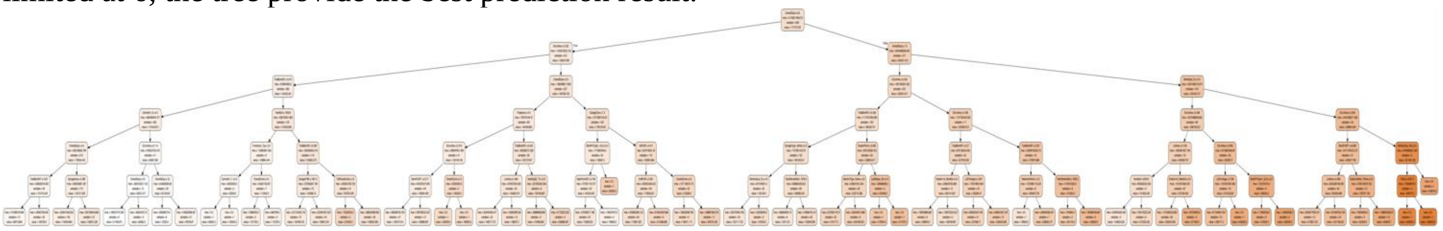
As $C_{\alpha}(T) = Error(T) + \alpha \times (\# Terminal\ Nodes)$, large alpha will be penalised for complex model while small alpha will create large sum square of error. By balancing the complexity and minimising the error generated, depth of tree n should equal to alpha.

As my regression tree too large to be fit in this word document.
Please refer to my codes for a whole picture of my regression tree at depth = 15.

d) Please compare the prediction performance for multiple linear regression and regression tree using test set MSE. Our goal is to predict housing price, which model should you choose?
For max_depth = 15,

MSE for multiple linear regression is 408341154.90
MSE for regression tree is 1963014670.57

I would choose multiple linear regression for prediction purpose as its MSE is much lower meaning the prediction power is stronger as it makes less error when encountering new data. Due to the high MSE, I decided to check whether my regression tree model are over-fitted. By try and error, my regression tree has low prediction power to unseen data, thus, I further prune the tree and minimise the MSE. When depth of tree is limited at 6, the tree provide the best prediction result.



MSE for regression tree with depth = 6 is 1642149748.94

However, the MSE for this new regression tree is still a lot larger than MSE of multiple linear regression, thus, as a predictor of house price, I would definitely choose multiple linear regression model.

4. Please write a short summary (less than 1 page) about this case. Please Indicate the business implication.

The most significant attributes are basement area, second floor area, lot area and kitchen quality. Every unit change on these attributes contribute to most significant changes in sale price of houses. Adjusted R square are over 80%, meaning sample data are very good fit of the model.

To begin with, the positively influencing attributes which leads to the increase of house price are LotArea, PoolArea, BsmtFinSF1, BsmtUnfSF and 2ndFlrSF. All of these predictors are area related ones, higher lot area have higher price is normal to be observed as you have to pay more if you are buying a larger house. But for the others, it is clear that house with big basement, large swimming pool and second floor would cost more. Not all houses has a basement, a pool and second floor, even in America. I would describe these house as villas or luxury houses which makes sense that villas cost higher in the market.

Followed by SaleType_CWD, StoneBr, WoodDeckSF, SaleCondition_Normal, ExterQual_Gd, BsmtExposure_Gd, Foundation_PConc. SaleType_CWD means the sale of house is warrant deed in cash, as performing warrant in a lump sum of cash has high interest rate, repayment for house is believed to be higher. StoneBr is a neighbourhood. This is very straight forward that houses in Stone Brook cost higher. WoodDeckSF is the area of wood deck of the house, although it is not as luxury as pool or second floor, normal speaking, only relatively large house will have large area of wood deck. While screen porch and open porch are the exterior structures gives an open view and block sunshine. BsmtExposure_Gd refers to walkout or garden level walls quality. It can be deemed as an attribute to indicate the appearance of garden. Houses having wood deck, garden, screen or open porch for sure are tended to be more expensive as houses that contain these structures are big and luxury houses in often. ExterQual_Gd refers to the materials to build exterior are good quality, better the exterior material used, more appealing the house will be and also its price. MasVnrType_Stone, Condition1_Norm and BsmtFullBath also did contribute to the increase of house price but in less proportion. It is always good to have masonry veneer to provide privacy and having walls implies that there has a garden as well. It is normal to cost more with a wall and garden. People also in favour of a full bathroom in basement maybe the convenience it brings appeals buyers who want to design basement as a working or entertainment area. People prefer their house to place at a normal place, do not enjoy living near railroad, off-site features or high-capacity urban roads. Perhaps is because of the noise pollution will be high living near to these features. Therefore, Condition1_Norm is related to those house which is not located near a busy traffic that generate heavy noise pollution.

BsmtQual_Fa and BsmtQual_Gd are all related to the height of basement. All basement height lower than 100 inches will bring negative impact to the price of the house. I think the reason behind is because if the height of basement is low, there are many limitation to optimise the use of basement and even useless to have these kind of basement. MSZoning_RM also lower the price of the house as medium residential density are semi-rural, semi-urban area in which are far away to workspace and cannot offer good living environment whereas people also not in favour of high density area. People do not like living in the neighbourhood Edwards, North Ames, Gilbert and Mitchell. Besides, kitchen quality contribute to the price decrease as well, if the house' kitchen has quality not to standard, house price will drop as cooking food is essential for all family, it is necessary to have a decent kitchen. Or else the sale price has to drop in order to attract buyers. Houses located adjacent to East-West railway will decrease in price maybe due to the high frequency of east-west train resulted in heavy noise pollution near the railway.

If I were a real estate developer, after consider this multiple linear regression model, in order to maximise the sale price of a new construction project, I would propose a project to build villas. First, all villa should equipped with a pool, basement with height higher than 100 inches, wood deck, porch and a second floor and using good quality materials to be used in the exterior of building. The neighbourhood of villas should located near Stone Brook and avoid Mitchell, North Ames, Gilbert and Edwards. Located near rural area and far away from East-West railway and last to ensure to have a very high quality kitchen be built within.