QBUS2820 – Predictive Analytics

*Assignment 2*

Anosh S

Task A – Moneyball

# Introduction

# Predictive analytics can empower investor decision making in the real estate market with knowledge that can result in more strategic plays to maximise return. To be more specific, through investors being able to predict the price of a house based on particular characteristics, an evaluation of risks and rewards can be derived from predicted returns. Therefore, the benefits of machine learning model predictions is that it works in favour for both housing buyers and sellers.

# Typically, investor evaluations of houses have required a natural intuition and manual brute force comparison of housing characteristics to understand the value of particular houses. Thus, an individual trying to enter the real estate market can be burdened and easily intimidated by the large quantities of data available, for self-research. Hence, it is the appeal of predictive analytics in being able to automate most housing research. As a result, this should ease evaluations overall through greater efficiency, and offering potentially more reliable and accurate means to determining house prices.

# Overall, four different models were constructed, each producing varying results. The performance difference between these models were relatively large, but understandable when examining the underlying concepts of each of the models. Therefore, the predictive capabilities of the best model could be used to its greatest potential, and its benefits fully expounded upon, if the limitations of the model are understood and the appropriate domain knowledge is applied when drawing any conclusions.

# Exploratory Data Analysis

Insight into the training and test dataset, and the training set has 1570 observations while the test set has 1210 observations. Moreover, there are 80 different possible features that can be used to predict “SalePrice”.

Train dataset has 1570 rows and 81 columns.

Test dataset has 1210 rows and 80 columns.

(figure 1)

Examining into whether NA values would be a problem gave that with none being present it would not be an issue of concern. This should effectively ease the analytical process with less cleaning required.

There are a number of columns with missing values. Once the best features have been selected the observations containing the NA values will be identified and dealt with accordingly.

There are 24 columns in train dataset with missing values.

(figure 2)

Identifying the datatypes of each column, there are a number of qualitative variables that may require dummy variables to be created for these. This will be applied after an individual inspection of each of the variables in feature engineering.

MS SubClass int64

MS Zoning object

Lot Frontage float64

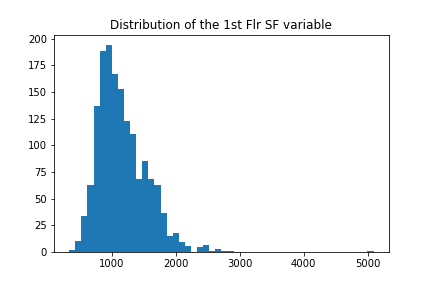
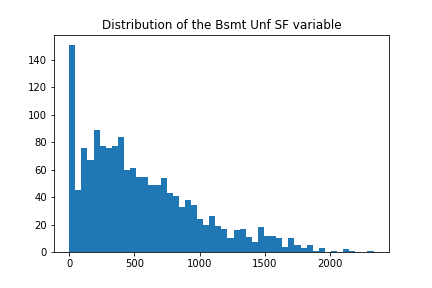
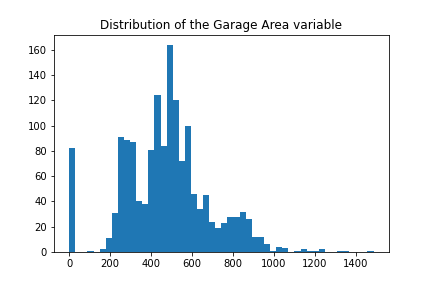
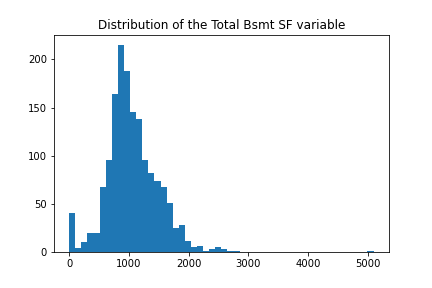
Lot Area int64

Street object

Alley object

…

(figure 3 - view full output in appendix i)

Before identifying the most ideal features, we will first delve into each of the variables to ensure that the data is logically consistent i.e. the values correspond logically with the variable that the data is trying to describe.

(figure 4 – refer to all other plots in appendix ii)

Nothing too unusual is noticeable in the data, however there is some skewness to a number of the features.

Using an IQR approach to identify outliers, these should be noted.

The dataset has 966 outliers.

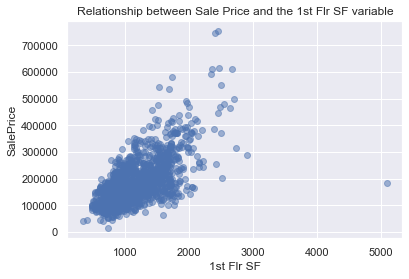
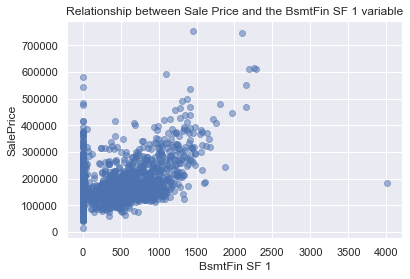
(figure 5)

Since, there were 966 observations that contain at least one outlier under one of its features, this represents a significant proportion of the train dataset and thus were not ignored in training.

# Feature Engineering

First, we will examine for the most relevant numerical variables in the dataset. Later, the best selected categorical features will be converted to dummies so that they can be usable for predictive modelling.

Subtracting the year it was sold in to the year it was constructed the new variable ‘Age’ was created. Also, subtracting the last year it was remodeled or had additions made to the house to the year it was sold was added as the “Last remod/add” feature. The same was also done for "Garage Yr Blt".

The following is a series of scatter plots of the numerical features against “SalePrice” to understand what relationships may be present between these variables.

(figure 6 – refer to all other plots in appendix iii)

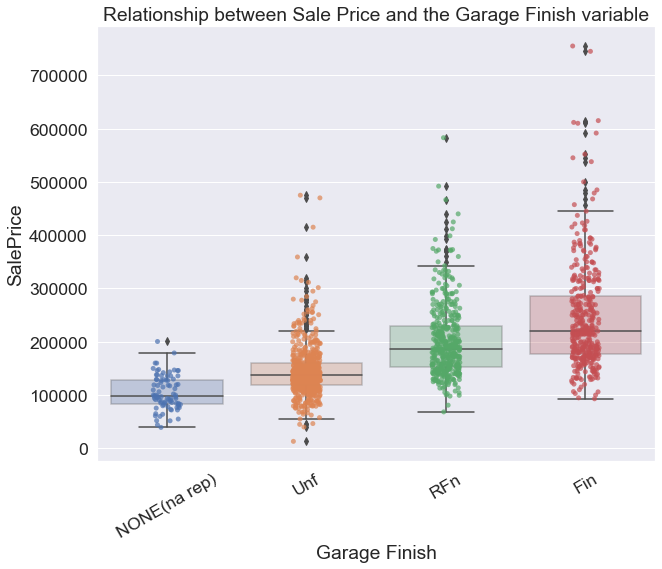
Viewing the scatter plots the best variables that were considered were "Gr Liv Area", "Garage Area", "Age", "Last remod/add", "Total Bsmt SF", "1st Flr SF", '2nd Flr SF', “BsmtFin SF 1”, “Full Bath”, “TotRms AbvGrd”, “Fireplaces” and “Garage Cars”. This is due to these showing some form of a non-linear relationship.

Since the scatter plots of “Age” and “Garage Age” appeared to have very similar scatter plots these variables were compared for multicollinearity to ensure linear models were not impacted in model construction.

0.8281

(figure 7)

Given that the correlation between the two was found to be relatively high the potential for these to cause potential multicollinearity issues for linear models meant these were removed for linear model feature subsets.

To see if any relationship is present between the categorical variabels and the “SalesPrice” predictor, box plots were made between these.

(figure 7 – refer to all other plots in appendix iv)

The best categorical features would be "Kitchen Qual", "Garage Finish" and "Overall Quality". This was on the basis that there was a sufficient number of observations between the different categories, a clear pattern was evident and that the variation (the whiskers of the boxplots) between groups did not overlap excessively. "Alley" would have also been considered, however, the presence of NA values that did not show a pattern or a distinct variation between the other categories meant NA values had little meaning.

Before confirming the final subset of features, to ensure that there were no issues in training and evaluation the features that contained missing values were to be dropped or imputed with an average.

The missing values under "Total Bsmt SF" and "BsmtFin SF 1" were replaced with their respective mean, since only one observation was found to contain these missing values under the features. Moreover, provided that only one observation in the test set had an NA value for "Garage Area" and "Garage Cars" it was best to impute the values of this single row with their means rather than removing the entire "Garage Area" and “Garage Cars” variables. Also, it was noticed that during model evaluation some of the observations in the test dataset under "Kitchen Qual" contained observations that had a category not defined in the training set. Hence, “Kitchen Qual” was disregarded from both sets.

Therefore, the following features that were considered for model construction and evaluation were "SalePrice", "Gr Liv Area", "Garage Area", "Age", "Last remod/add", "Total Bsmt SF", "1st Flr SF", "Kitchen Qual", "Overall Qual", "Garage Finish", '2nd Flr SF', 'BsmtFin SF 1', 'Full Bath', 'TotRms AbvGrd', 'Fireplaces' and 'Garage Cars'.

To finish feature engineering the train and test narrowed dataframes were finalised with its numerical variables being standardised. Typically the categorical variables would have been dummified here too, however, to simplify the model construction workflow this was done just before the subset of features were used to train the model.

# Methodology and Modelling

## kNN Regression

As the third best model and one that was studied during the unit, the kNN regression method achieved reasonable results. It is a non-parametric supervised learning approach that requires no distribution assumptions. The model predicts for some input point as:

where is the training sample i.e. . Therefore, the model predicts based on the sample average of the response values for the number of training observations that are closest to the point .

Using the kNN model formulation best subset selection was done to determine the best model from the best features derived from feature engineering. However, given that conducting best subset selection across fifteen features is highly time consuming, only different combinations of the features were trained for models containing twelve, thirteen, fourteen or fifteen features.

-------------------------------------------

-------------------------------------------

------OVERALL BEST MODEL BY CV-RMSE--------

-------------------------------------------

-------------------------------------------

Overall Best kNN model uses the features ['Gr Liv Area', 'Garage Area', 'Age', 'Last remod/add', 'Total Bsmt SF', '1st Flr SF', 'Garage Finish', '2nd Flr SF', 'Full Bath', 'Fireplaces', 'Garage Cars', 'Garage Age']

CV RMSE: 32680.7791

Number of neighbours: 6

(figure 13 – full results in appendix vi)

After formulating each of the different model possibilities the best model used features 'Gr Liv Area', 'Garage Area', 'Age', 'Last remod/add', 'Total Bsmt SF', '1st Flr SF', 'Garage Finish', '2nd Flr SF', 'Full Bath', 'Fireplaces', 'Garage Cars' and 'Garage Age' with the k set to 6. This was decided as the most optimal, since it achieved the lowest cross-fold validation RMSE of $32680.7791.

Since, the kNN model is non-parametric there are no assumptions made on the underlying data distribution, and so it is possible to accept the above model as a valid alternative means to predict housing sale prices.

## Random Forest Regression

As an extension of decision tree regression, this formulation is able to improve markedly on performance through removing much of the over fitting that comes from fitting a single tree. Essentially, it is a robust supervised learning method that uses averaging across a number of randomly fitted decision trees to make a final prediction. Therefore, given a number of trees in a random forest, to predict some point we take the average of the predictions made by the number of trees,

where is the final prediction of the random forest and is the prediction given by the tree of the random forest.

In a random forest trees are formed from sub-samples of the data that are derived from bootstrapping aggregation techniques. This allows for reduced variance whilst keeping bias the same, optimising generalisation ability.

Utilising the same best subset selection methods in kNN regression, random forest regression also trained various models based on different combination feature subsets of the best fifteen variables. However, like before this was only of subsets containing twelve, thirteen, fourteen or fifteen features.

-------------------------------------------

------OVERALL BEST MODEL BY CV-RMSE--------

-------------------------------------------

Overall Best RF model uses the features ['Gr Liv Area', 'Age', 'Last remod/add', 'Total Bsmt SF', '1st Flr SF', '2nd Flr SF', 'BsmtFin SF 1', 'Full Bath', 'TotRms AbvGrd', 'Fireplaces', 'Garage Cars', 'Garage Age']

CV RMSE: 31138.1507

-------------------------------------------

-------SECOND BEST MODEL BY CV-RMSE--------

-------------------------------------------

Second Best RF model uses the features ['Gr Liv Area', 'Garage Area', 'Age', 'Last remod/add', 'Total Bsmt SF', '2nd Flr SF', 'BsmtFin SF 1', 'Full Bath', 'TotRms AbvGrd', 'Fireplaces', 'Garage Cars', 'Garage Age']

CV RMSE: 31244.5667

(figure 14 – full results in appendix vii)

The best two models were formulated with random forest regression. The lowest attaining CV RMSE was $31138.1507 using the features 'Gr Liv Area', 'Age', 'Last remod/add', 'Total Bsmt SF', '1st Flr SF', '2nd Flr SF', 'BsmtFin SF 1', 'Full Bath', 'TotRms AbvGrd', 'Fireplaces', 'Garage Cars' and 'Garage Age'. The next best model with random forest regression achieved a CV RMSE of $31244.5667 with the same set of features but with ‘Garage Age’ replaced with ‘1st Flr SF’.

As a non-parametric model, similar to kNN regression, random forest regression has no formal distributional assumptions. Hence, the above best two selected models are valid.

## Linear Regression

Applying the ordinary least squares method, the linear regression model used to explain housing sale prices was derived through selecting the coefficient values that minimise the residual sum of squares,

Again, we used the same method of best subset selection as the other models. However, as discovered earlier, “Garage Age” was highly correlated with “Age”, and as a means of avoiding potential multicollinearity issues the “Garage Age” variable was excluded from subset selection.

While the above may be the best models using linear regression, there are underlying assumptions that must be satisfied in order for it to be valid for use in predictions. Therefore, to make any conclusions from the above linear models, a checking of assumptions would be first required.

## Polynomial Regression

As an extension of decision tree regression, this formulation is able to improve markedly on performance through removing much of the over fitting that comes from fitting a single tree. Essentially, it is a robust supervised learning method that uses averaging across a number of randomly fitted decision trees to

# Model Evaluation

|  |  |  |
| --- | --- | --- |
|  | CV RMSE | Test RMSE |
| MLR | 3.4782 | 4.0660 |
| kNN | 3.6202 | 4.0645 |
| SVR | 3.7372 | 4.0768 |

(figure 15)

Having performed model construction, the following delves into the ability for the each of the best models to generalise with unseen data i.e. test data. Overall, each of the best models of each of the types, achieved an above target test RMSE of $4.1 million. Also, it should be noted that the bias-variance tradeoff is addressed most optimally with the use of cross validation. These are good results, however, the relatively large difference between the cross validation and test performance across all the models brings some concerns. One way of possibility as to why this was the case is the fact that the split of data between the training and test sets does not use the conventional 80 to 20, but rather a 50 to 50 split. Hence, the models could have generalised better if it had a larger training set that made them less prone to unfamiliar observations that inflate the test RMSE.

The multiple linear regression model achieves the best performance overall, attaining the lowest cross validation RMSE and close to lowest test RMSE. The kNN model does slightly worse in cross validation, however, it is important to recall that MLR produced only the best model once outliers were excluded, while kNN achieved above target results even with outliers. Therefore, the higher cross validation RMSE can be accounted to these outliers and also the tendency of kNN models to be typically sensitive to noisy data. Otherwise, its achieving of an above target RMSE is indicative of good generalising ability. Also, since the kNN model is non-parametric and does not make any assumptions on the distribution of the data and the fact that it performed similar, if not, just as well as MLR, it is a testament to kNN being an effective alternative to predicting NBA salaries.

It was expected that the SVR model would be robust to outliers and thus was trained with the data that included them. This robustness is evident in the results above. Its cross validation RMSE is still relatively worse than the other two, however, it was still able to achieve above target and thus implies that it can generalise well. Thus, it is recommended that further formulation testing that adjusts to different values for the margin of tolerance could potentially allow for us to derive a more improved model.

Therefore, the kNN model is the best performer overall, since the MLR’s validity is questionable given some of its assumptions could not be satisfied confidently and that SVR achieved a slightly worse performance in terms of cross validation and test RMSE.

# Conclusion

In conclusion, the outcomes of the research in predictive modelling of NBA salaries are quite promising. Each of the models achieved above target test RMSEs and gave results that were consistent with what was expected. While these reasons support the practical implementation of these models in assisting NBA teams, there are limitations that need to be noted. Improvements to consider is that the data provided could have been organised such that the training and test split was more conventional, so a better generalisation performance could have been achieved. Moreover, the data could have been more complete with a wider selection of metrics on players, so that they could have expanded the predictive abilities of the models, through a more accurate representation of NBA players overall. Also, while the predictors selected were posited to have some correlation with salary it is important to recognise that these are associations with salary and not causal relationships. Given these limitations, important considerations for future research is that the provided data is more complete with a better train-test split, the SVR model is reformulated against different margins of tolerance to potentially derive a more improved model, and that other models are experimented to explore other potentially better alternatives.

Therefore, while these models are definitely useful tools, these should not be the sole reasons to any final decisions. This requires that the users of these tools makes use of domain knowledge and takes into account other factors that may not be considered by them, such as the constantly changing circumstances of salary conditions in the NBA league.

Task B – Gradient Ascent

# Exploratory Data Analysis

The column and row dimensions of the dataset are as follows.

The dataset has 506 rows and 14 columns.

(figure 16)

Determining the number of NA values, none were present.

There are 0 columns in train dataset with missing values.

(figure 17)

Each of the features are of numerical datatypes with only one binary categorical variable, that being “chas”, as reported on the data description document and can be confirmed in the distribution plots (refer to appendix ix).

crim float64

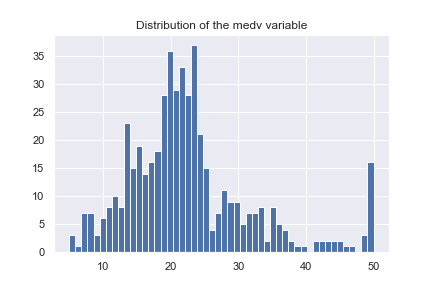
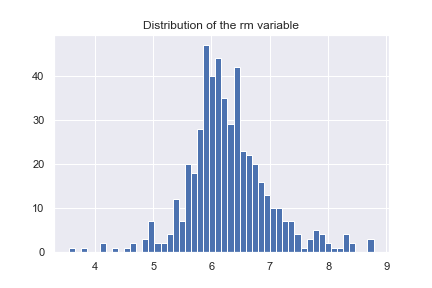
zn float64

indus float64

chas int64

…

(figure 18 - view full output in appendix viii)

Ensuring that all inputted values of the dataset are logically consistent, the distribution plots below will give insight.

(figure 19 – refer to all other plots in appendix ix)

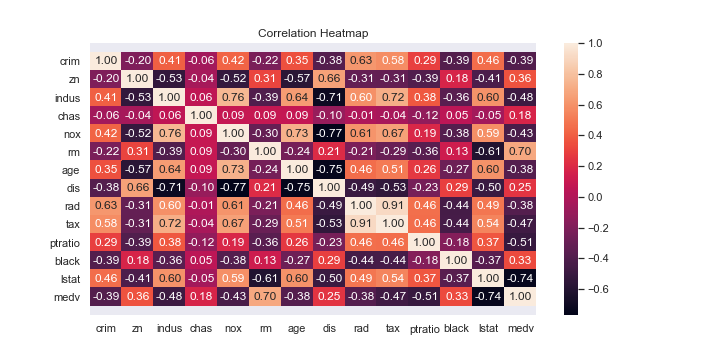
Viewing the distributions and comparing them to the data descriptions provided, the variables appear to be logically consistent.

In detecting for outliers, using the IQR method, 288 were found.

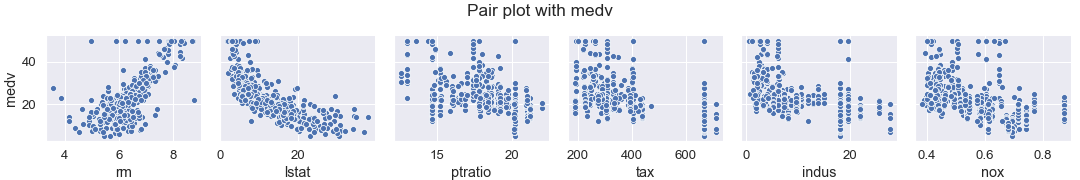
The dataset has 288 outliers.

(figure 20)

# Feature Engineering

To derive a good feature set for linear regression, we will conduct dimensionality reduction using a correlation heatmap to select the most relevant features.

(figure 21)

Referring to the above plot, the top 3 most correlated features with “medv” are “rm”, “lstat” and “ptratio”. “rm” is positively correlated while the other two are correlated negatively. To confirm the strength and linearity of these relationships a pair plot will be examined. The next three most correlated features will also be included in the pair plot to see if these variables may be better at predicting "medv".

(figure 22 – for complete pair plot refer to appendix x)

The pair plot shows that “rm” has a relatively strong linear relationship with “medv”, while “lstat” seems to follow a rough exponential form. However, “lstat” would remain to be an appropriate feature to explain “medv” in the linear model. The rest seem to share similar patterns in a relationship with “medv”, and so on the basis of attaining the highest correlation, “ptratio” remains as the third best feature to include.

It should be noted however that, “lstat” and “ptratio” share a -0.61 correlation and thus introduces a concern for possible multicollinearity. This will be assessed further with the VIF.

VIFs with "rm", "lstat", "ptratio": [1.6534, 1.6794, 1.1981], \max\_vif: 1.6794

(figure 23)

The results show that no multicollinearity is present between the best three features. Therefore, it is possible to conclude that “rm”, “lstat” and “ptratio” are the best to use in a linear regression model to predict “medv”.

# Gradient Ascent Implementation

Having constructed the gradient ascent algorithm, the algorithm is run through a leave one out cross validation across different values of alpha, whilst keeping the number of iterations fixed. This is because, it is expected that the performance of gradient ascent increases with greater values in the number of iterations. Thus, in assessing for different values of alpha whilst keeping the number of iterations fixed would be a sufficient method to deriving the maximum likelihood. Therefore, we fixed this value at 10 so to ensure algorithm had iterated enough to reach the optimum whilst retaining computational efficiency.

------------------|-------------------

ALPHA | CV RMSE

------------------|-------------------

0.001 | 0.9925

0.01 | 0.9322

0.1 | 0.7035

0.2 | 0.6743

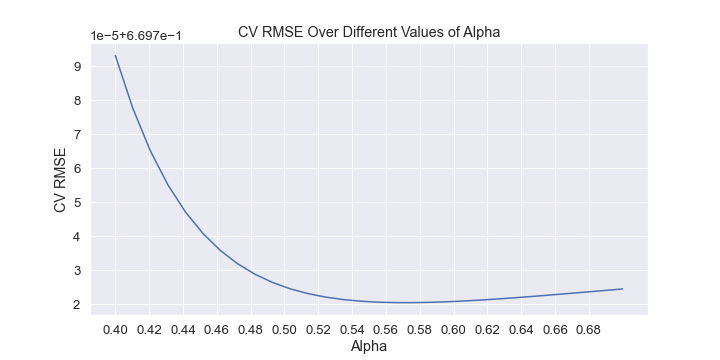
0.5 | 0.6697

1.0 | 0.6702

3.0 | 796553.9641

6.0 | 2572815658.9413

(figure 24)

Considering the provided output, the best alpha was 0.5, since it achieved the lowest cross validation RMSE. Now, it is possible to find a more precise value for alpha through running a cross validation over alphas close to 0.5.

(figure 25)

Therefore, from viewing the plot, the most optimal value of alpha is 0.57. This is since cross validation RMSE is depicted to be the lowest at this point as shown above.

References

<https://medium.com/swlh/random-forest-and-its-implementation-71824ced454f>

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html>

<https://www.math.mcgill.ca/yyang/resources/doc/randomforest.pdf>

# Appendix

## Appendix i

MS SubClass int64

MS Zoning object

Lot Frontage float64

Lot Area int64

Street object

Alley object

Lot Shape object

Land Contour object

Utilities object

Lot Config object

Land Slope object

Neighborhood object

Condition 1 object

Condition 2 object

Bldg Type object

House Style object

Overall Qual int64

Overall Cond int64

Year Built int64

Year Remod/Add int64

Roof Style object

Roof Matl object

Exterior 1st object

Exterior 2nd object

Mas Vnr Type object

Mas Vnr Area float64

Exter Qual object

Exter Cond object

Foundation object

Bsmt Qual object

Bsmt Cond object

Bsmt Exposure object

BsmtFin Type 1 object

BsmtFin SF 1 float64

BsmtFin Type 2 object

BsmtFin SF 2 float64

Bsmt Unf SF float64

Total Bsmt SF float64

Heating object

Heating QC object

Central Air object

Electrical object

1st Flr SF int64

2nd Flr SF int64

Low Qual Fin SF int64

Gr Liv Area int64

Bsmt Full Bath float64

Bsmt Half Bath float64

Full Bath int64

Half Bath int64

Bedroom AbvGr int64

Kitchen AbvGr int64

Kitchen Qual object

TotRms AbvGrd int64

Functional object

Fireplaces int64

Fireplace Qu object

Garage Type object

Garage Yr Blt float64

Garage Finish object

Garage Cars int64

Garage Area int64

Garage Qual object

Garage Cond object

Paved Drive object

Wood Deck SF int64

Open Porch SF int64

Enclosed Porch int64

3Ssn Porch int64

Screen Porch int64

Pool Area int64

Pool QC object

Fence object

Misc Feature object

Misc Val int64

Mo Sold int64

Yr Sold int64

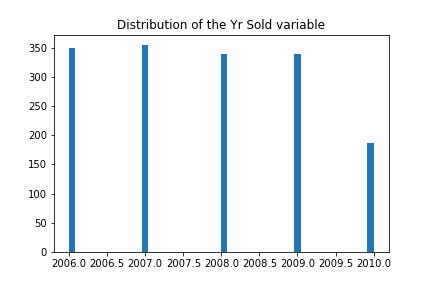
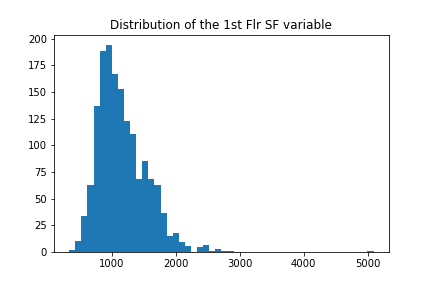
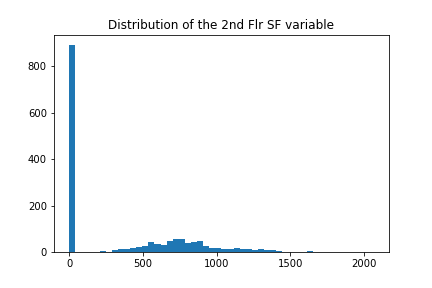
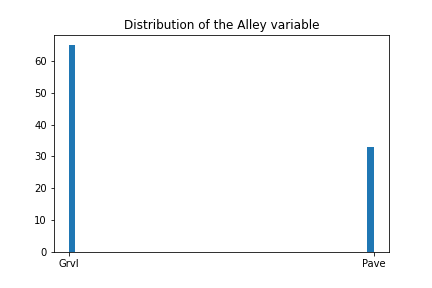
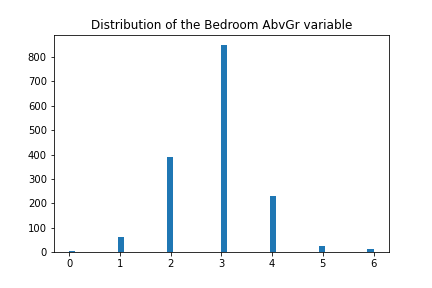
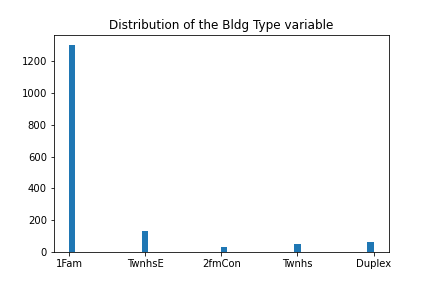
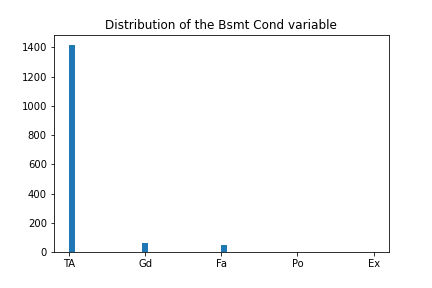
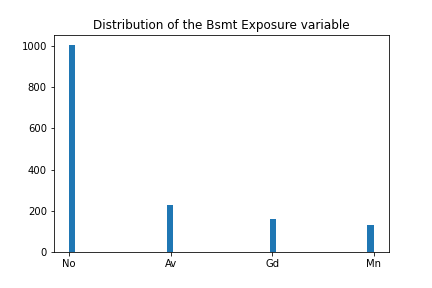
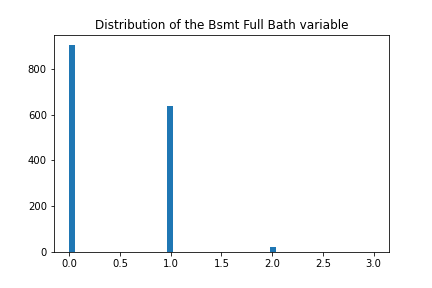
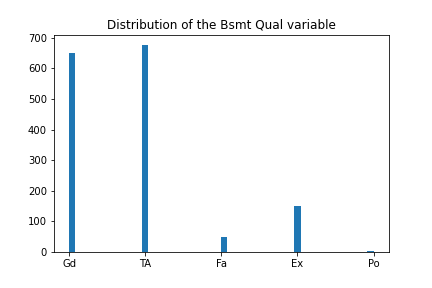
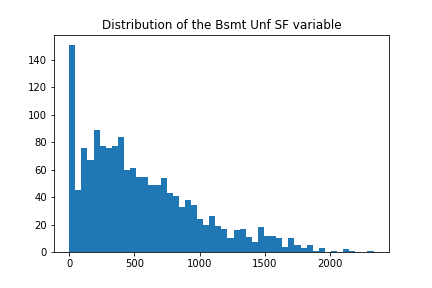
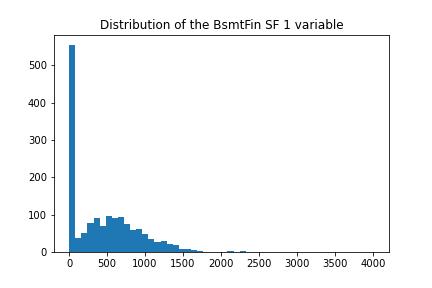
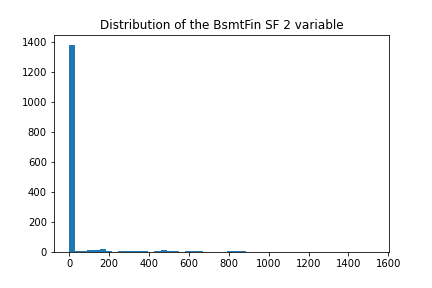
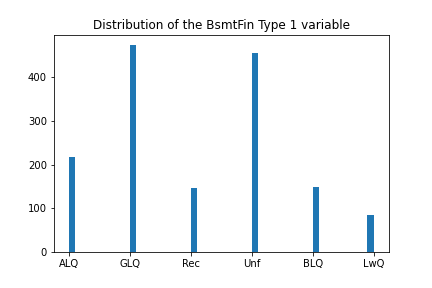
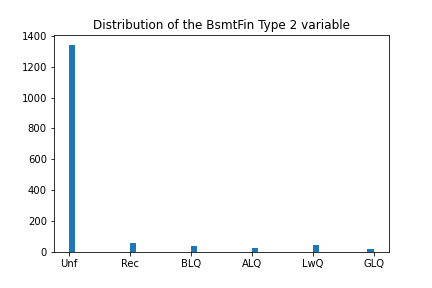
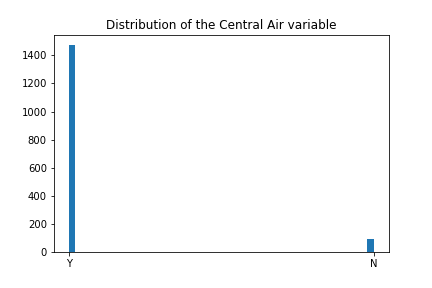
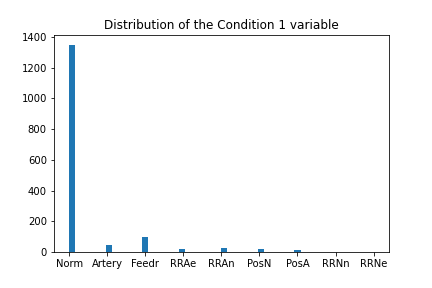
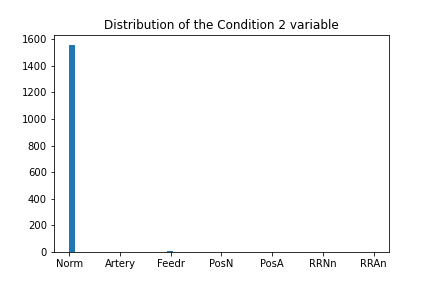
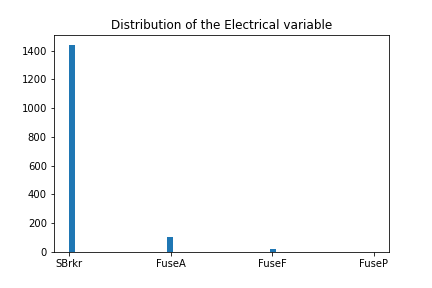
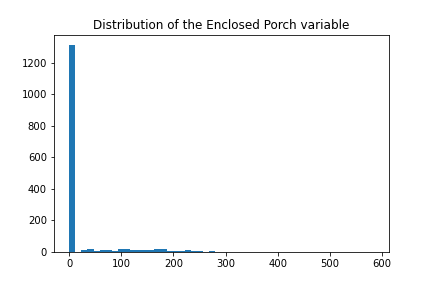
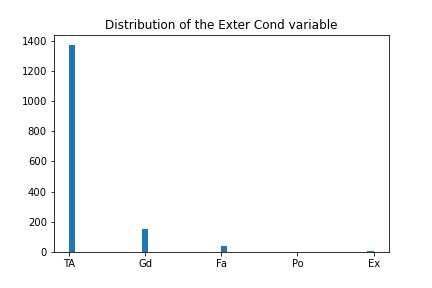
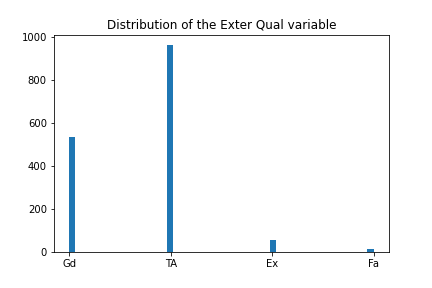
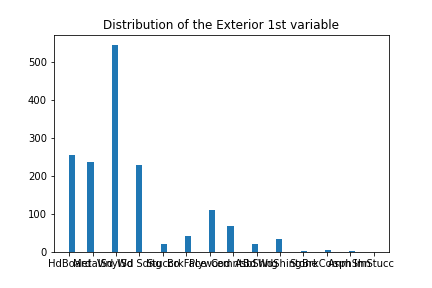
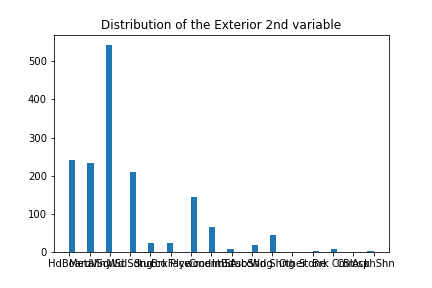
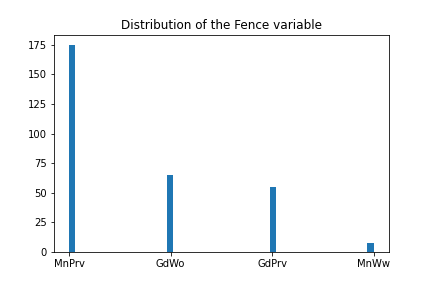
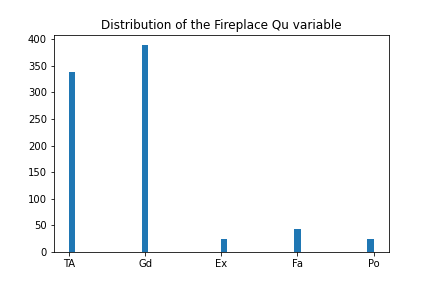
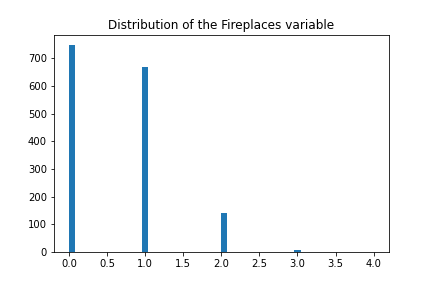
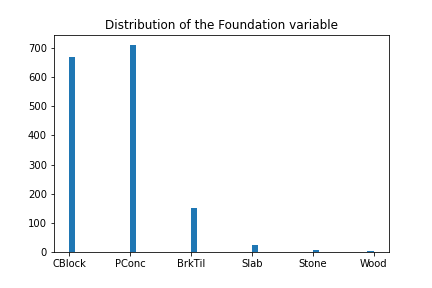
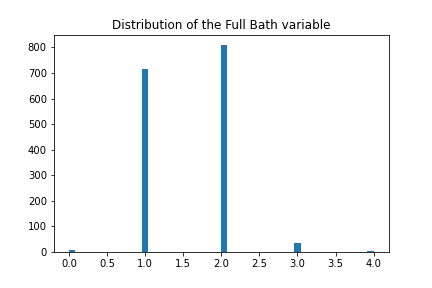
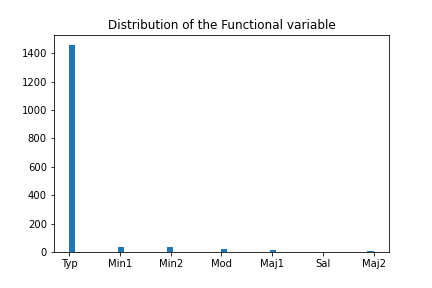
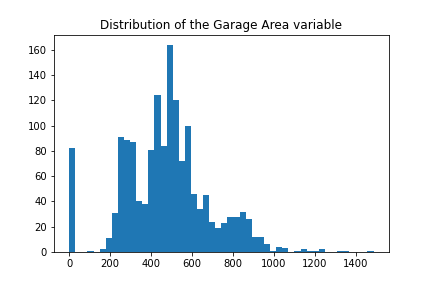
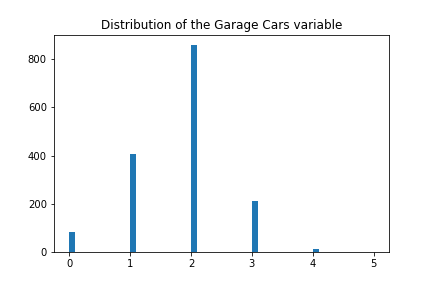
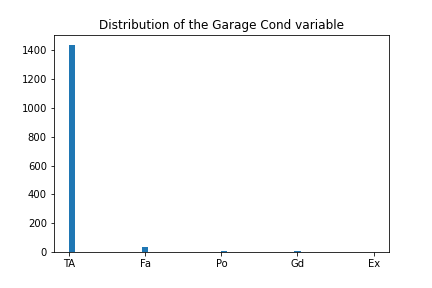
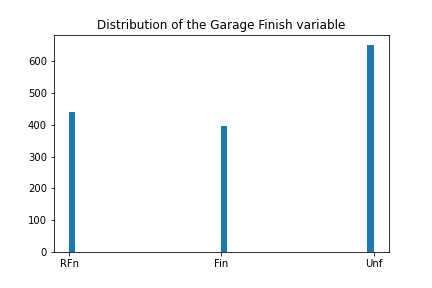
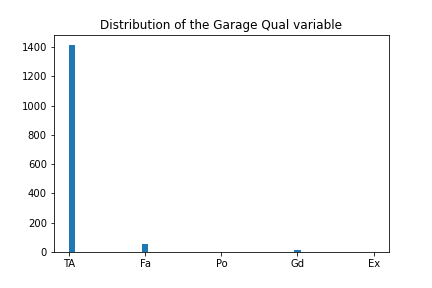
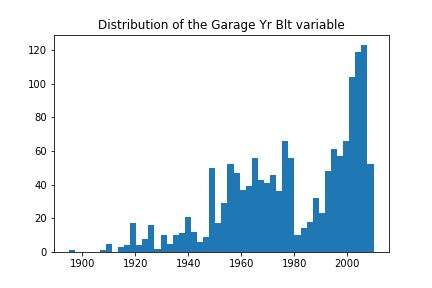
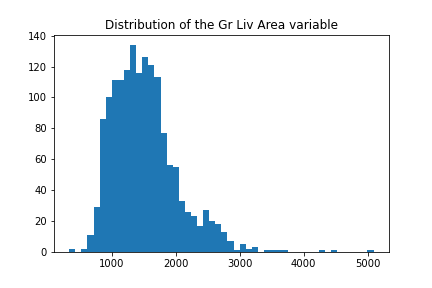
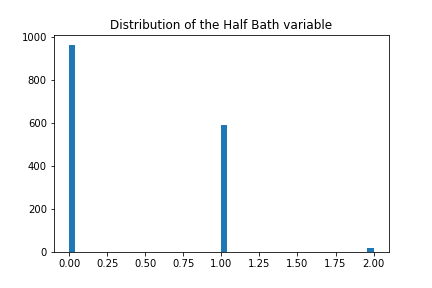
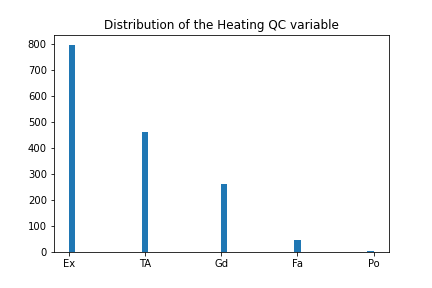
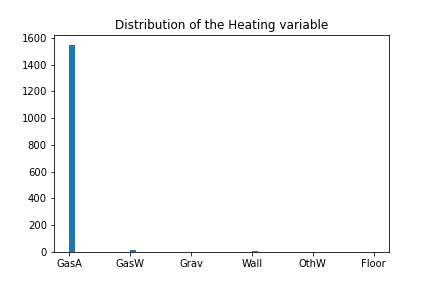
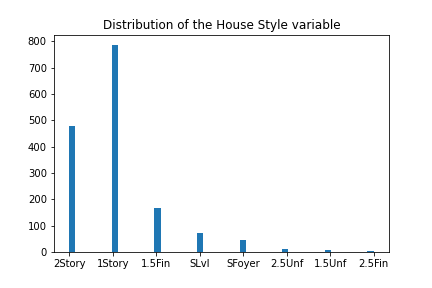
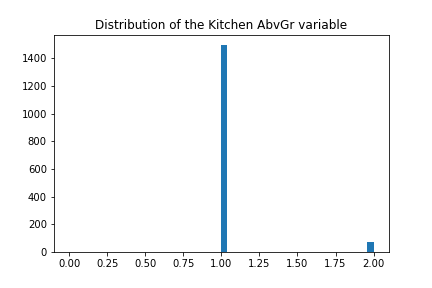
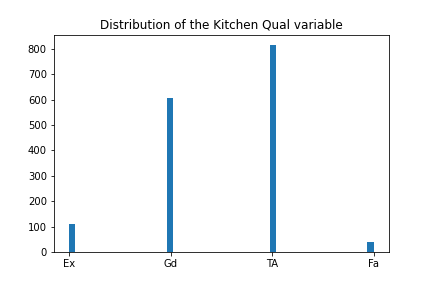
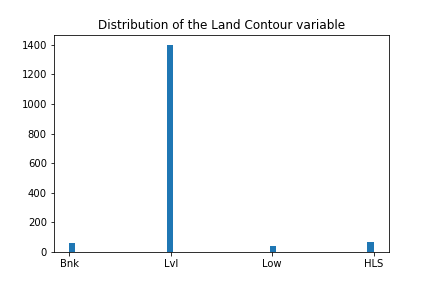
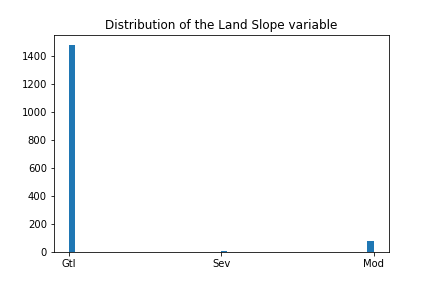
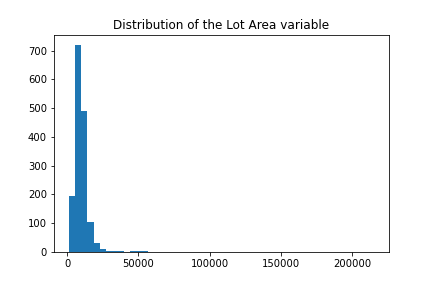
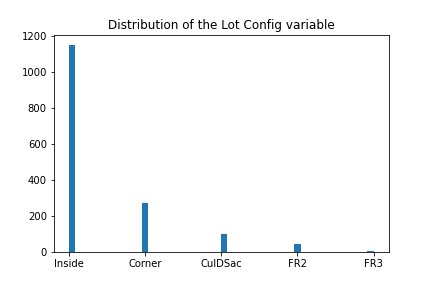
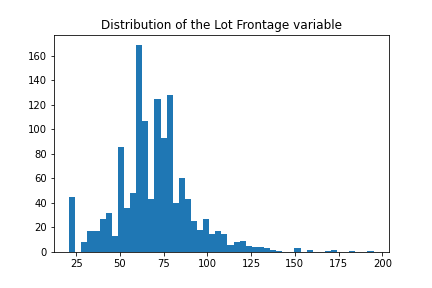
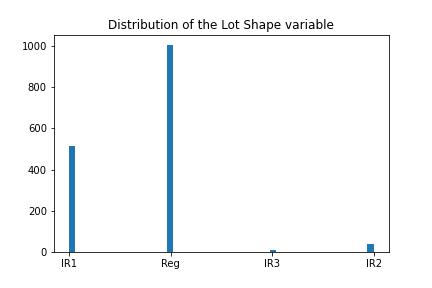
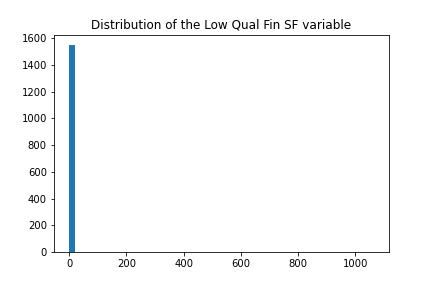
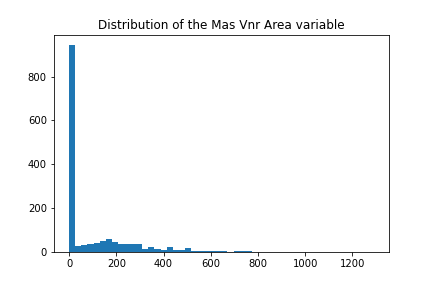
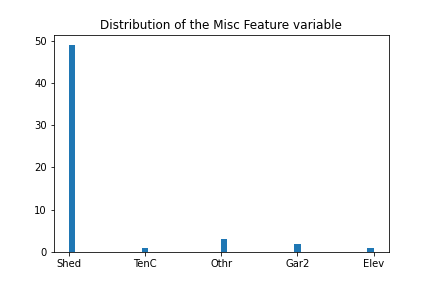
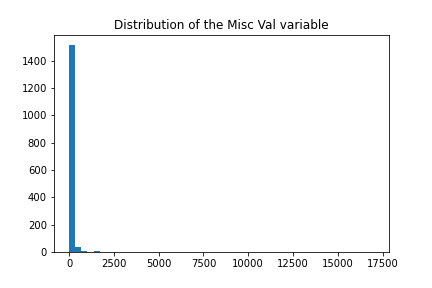
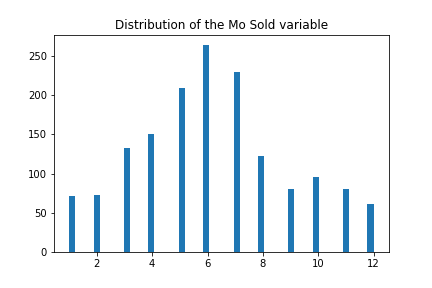
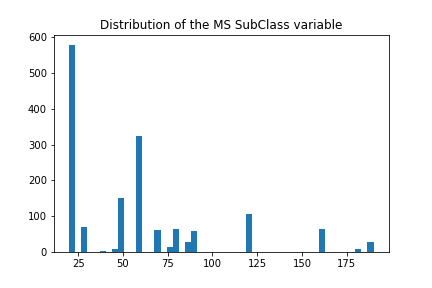
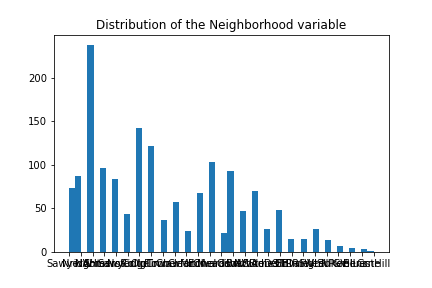
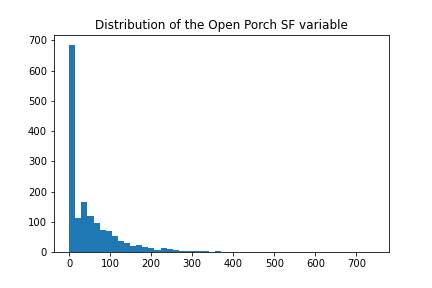
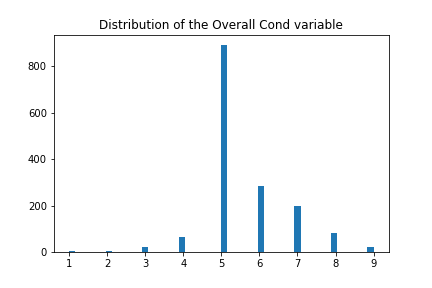
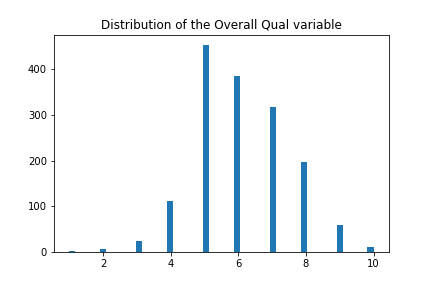
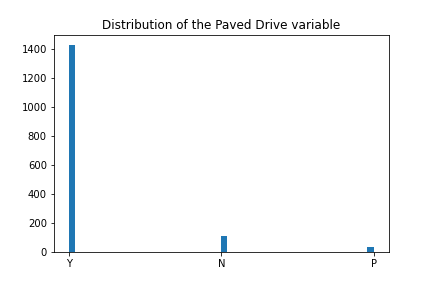
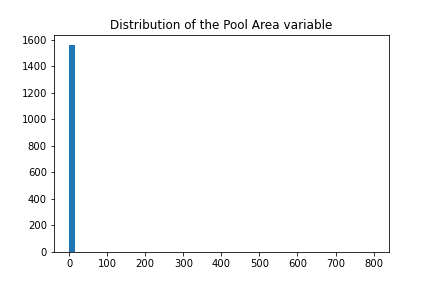
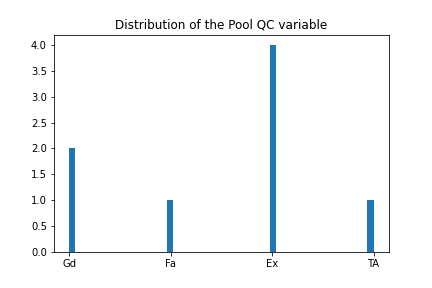
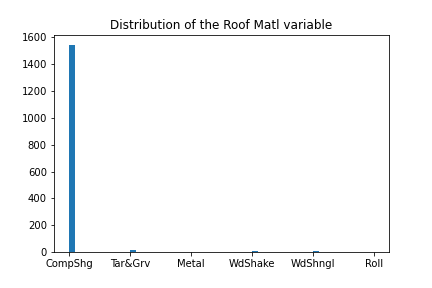
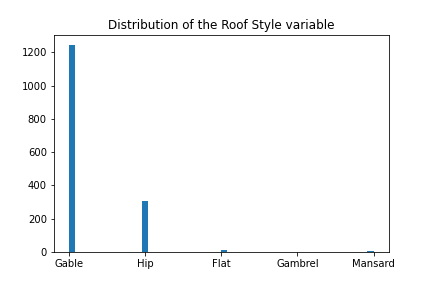
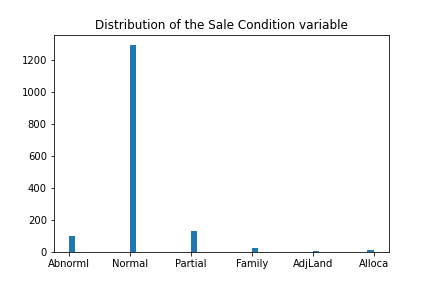
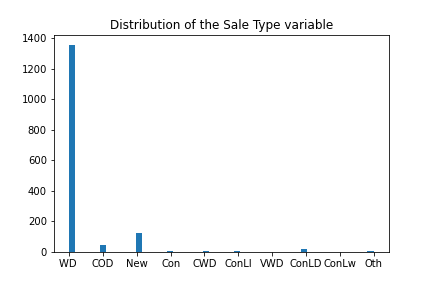
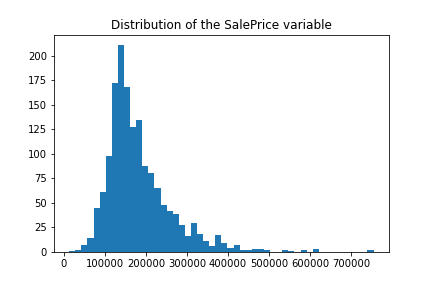
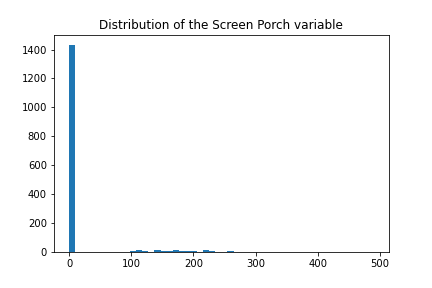
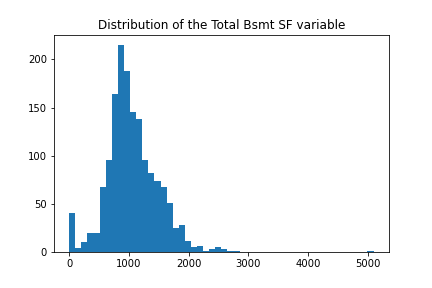
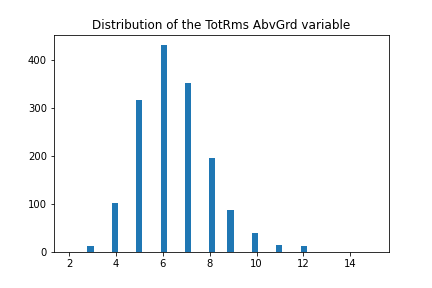
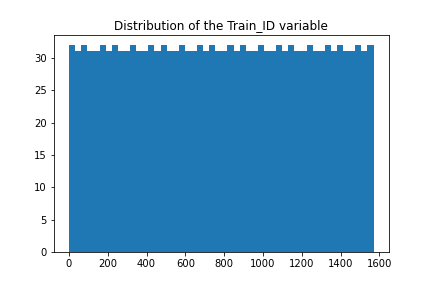
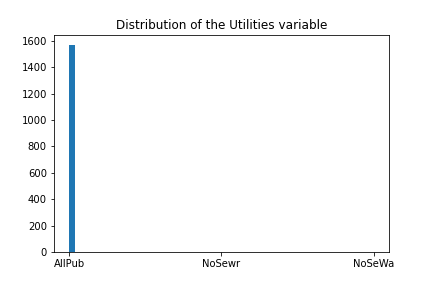
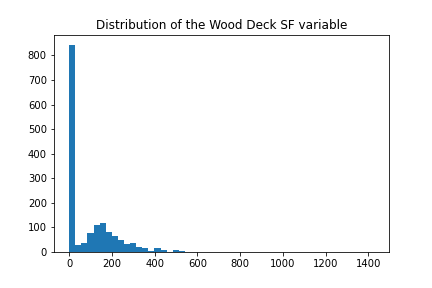
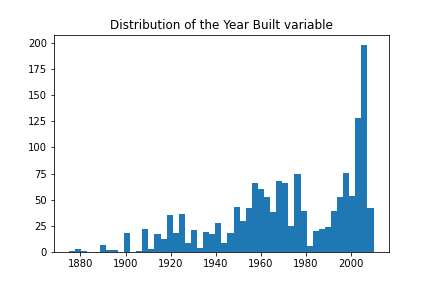
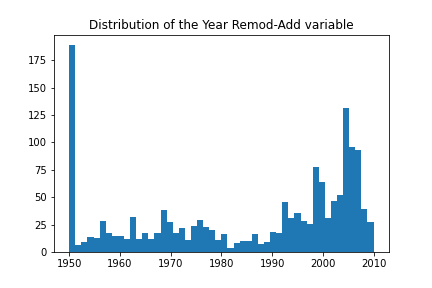
Sale Type object

Sale Condition object

SalePrice int64

dtype: object

## Appendix ii



## Appendix iii

## Appendix iv

## Appendix v

-------------------------------------------

---------------WITH OUTLIERS---------------

-------------------------------------------

------OVERALL BEST MODEL BY CV-RMSE--------

-------------------------------------------

-------------------------------------------

Overall Best mlr model uses the features ['PER', 'DWS', 'Minutes', 'USG']

CV RMSE: 3.7293

Test RMSE: 4.1146

---------------WITH OUTLIERS---------------

-------------------------------------------

------OVERALL BEST MODEL BY TEST-RMSE------

-------------------------------------------

-------------------------------------------

Overall Best mlr model uses the features ['WS', 'Minutes']

CV RMSE: 4.0987

Test RMSE: 3.9841

…

-------------------------------------------

-------------WITHOUT OUTLIERS--------------

-------------------------------------------

------OVERALL BEST MODEL BY CV-RMSE--------

-------------------------------------------

-------------------------------------------

Overall Best mlr model uses the features ['WS', 'Minutes', 'USG']

CV RMSE: 3.4782

Test RMSE: 4.0660

-------------WITHOUT OUTLIERS--------------

-------------------------------------------

------OVERALL BEST MODEL BY TEST-RMSE------

-------------------------------------------

-------------------------------------------

Overall Best mlr model uses the features ['WS', 'Minutes']

CV RMSE: 3.8000

Test RMSE: 3.9761

## Appendix vi

-------------------------------------------

------OVERALL BEST MODEL BY CV-RMSE--------

-------------------------------------------

-------------------------------------------

Overall Best kNN model uses the features ['PER', 'DWS', 'Minutes', 'USG']

CV RMSE: 3.6202

Test RMSE: 4.0645

Number of neighbours: 11

-------------------------------------------

-------------------------------------------

------OVERALL BEST MODEL BY TEST-RMSE------

-------------------------------------------

-------------------------------------------

Overall Best kNN model uses the features ['PER', 'WS', 'Minutes']

CV RMSE: 3.9774

Test RMSE: 3.9741

Number of neighbours: 16

## Appendix vii

-------------------------------------------

------OVERALL BEST MODEL BY CV-RMSE--------

-------------------------------------------

-------------------------------------------

Overall Best svr model uses the features ['PER', 'DWS', 'Minutes', 'USG']

CV RMSE: 3.7372

Test RMSE: 4.0768

-------------------------------------------

------OVERALL BEST MODEL BY TEST-RMSE------

-------------------------------------------

-------------------------------------------

Overall Best svr model uses the features ['PER', 'WS', 'Minutes']

CV RMSE: 4.0052

Test RMSE: 4.0079

## Appendix viii

crim float64

zn float64

indus float64

chas int64

nox float64

rm float64

age float64

dis float64

rad int64

tax int64

ptratio float64

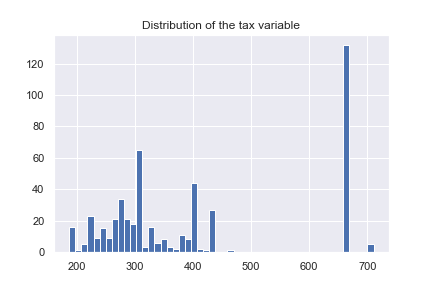
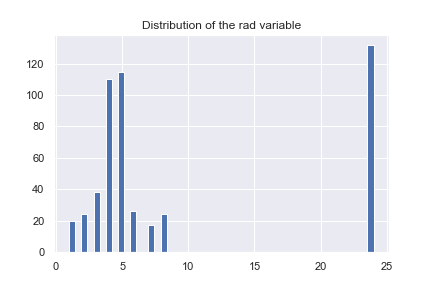
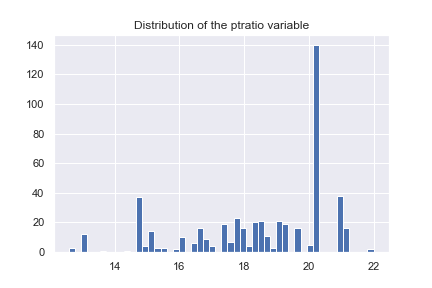
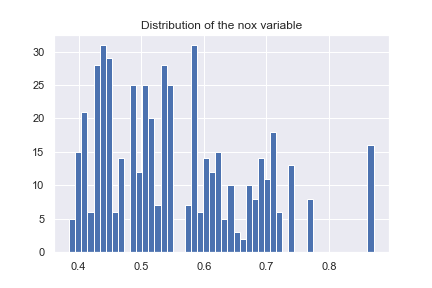
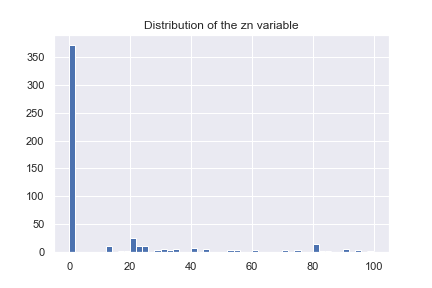
black float64

lstat float64

medv float64

dtype: object

## Appendix ix



## Appendix x

