London House Evaluation

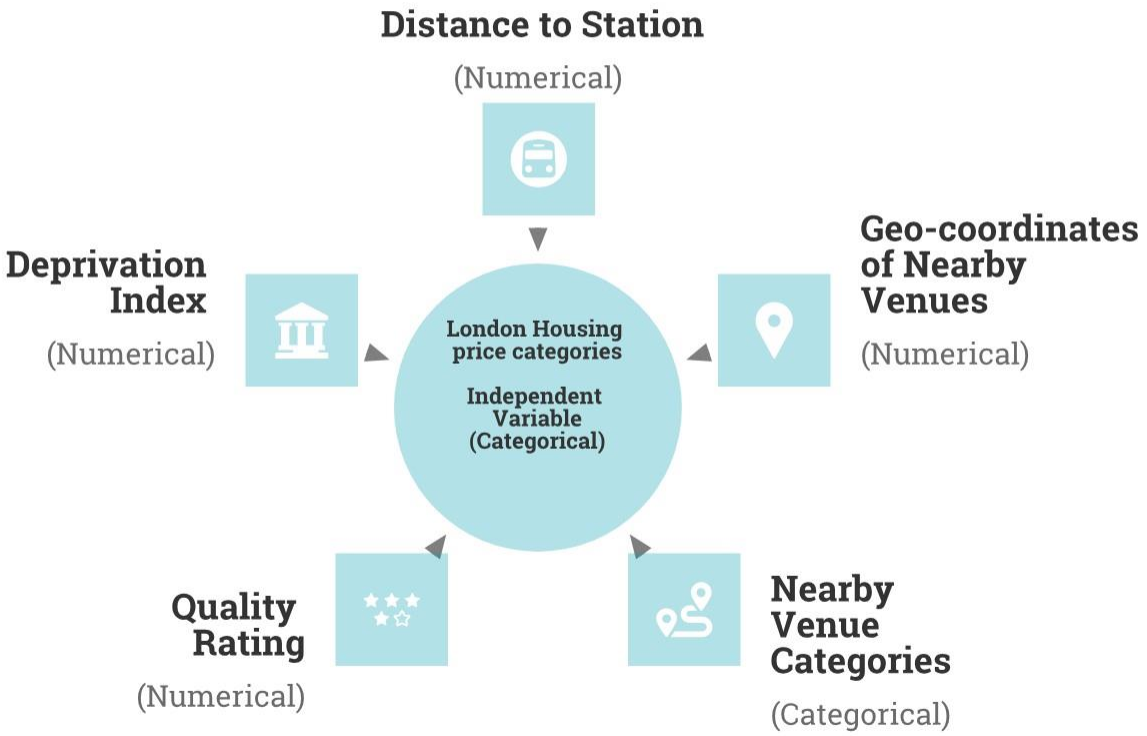
Coursera_DataSceince_CapstoneProject

Introduction

The current study

The current study attempts to predict the London housing price categories based on a set of locational features.

Target Variables and Candidate Feature Variables



Research Aims & Rationale

1

Traditional research primarily used ordinary least squares (OLS) for housing price prediction. However, recent research on house prediction in Sandiago has shown that machine learning algorithms, especially Random Forest have yielded superior performance than the traditional approach.

First Research Aim:

Extend the literature by examining whether the superior performance of Random Forest algorithm is replicable in the London housing market.

2

Literature review: housing prices are often modeled using regression models, whereby structural and locational attributes are used to explain variances in housing prices. This is justified on the basis that houses often share similar strucutral features with their nearby properties (Basu & Thibodeau, 1998). In addition, poverty index has also been used to identify at-risk housing (Margulis, 1998).

Second Research Aim:

Explore the predictive power of deprivation index, quality rating, distance to station, the venue categories and the geographical coordinates of the nearby properties.

Method

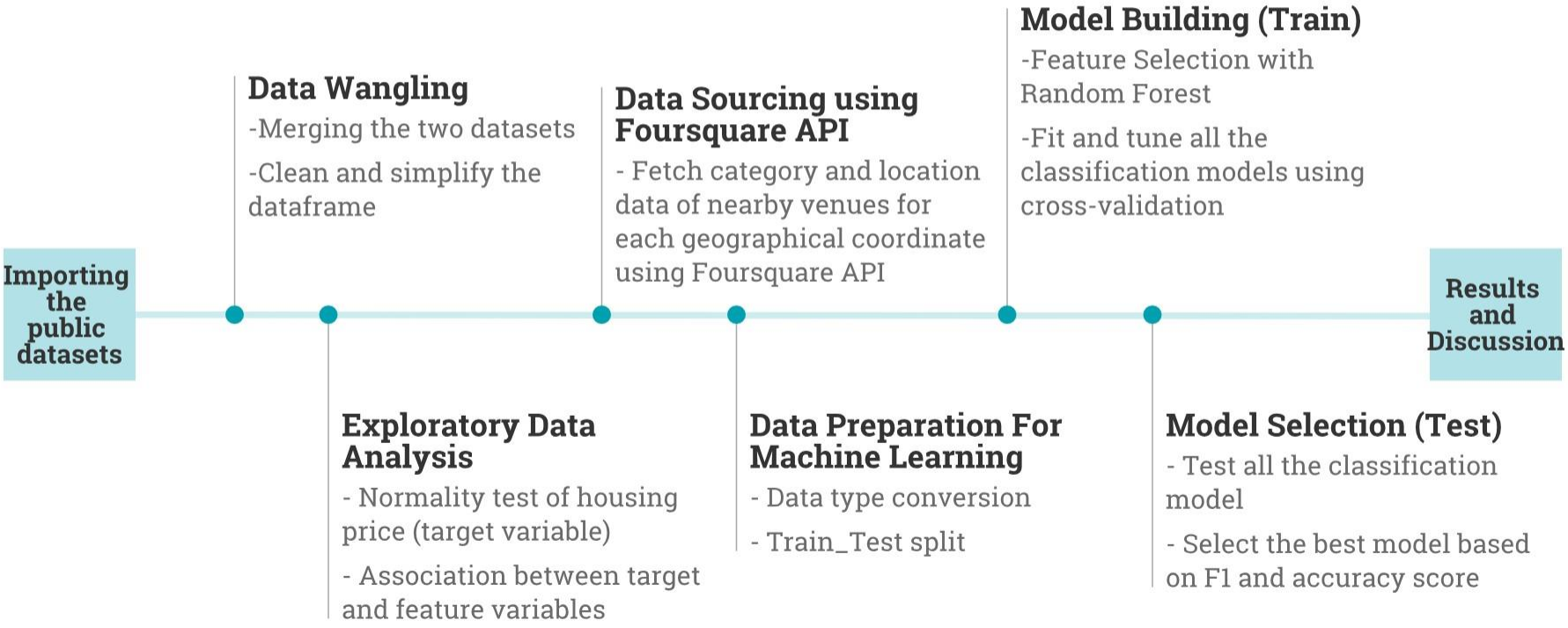
Data Sources

- **Public Dataset**
  - London Housing Price (numerical)
  - London Geographical Coordinates (numerical)
  - Deprivation Index (numerical)
  - Distance to Station (numerical)
  - Quality Rating (numerical)

- **FourSqaure API**
  - Nearby Venue Categories (Categorical)
  - Geographical Coordinates of the nearby venues (Numerical)

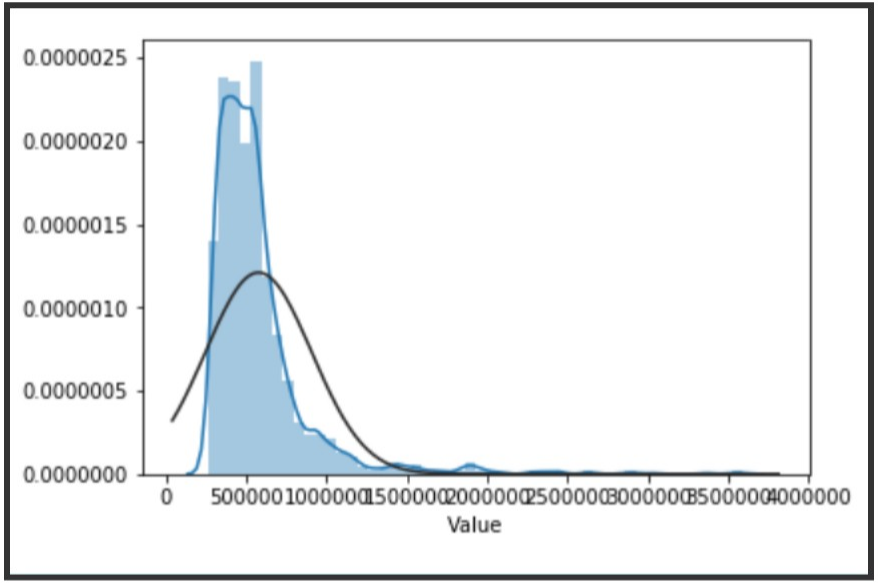
Method

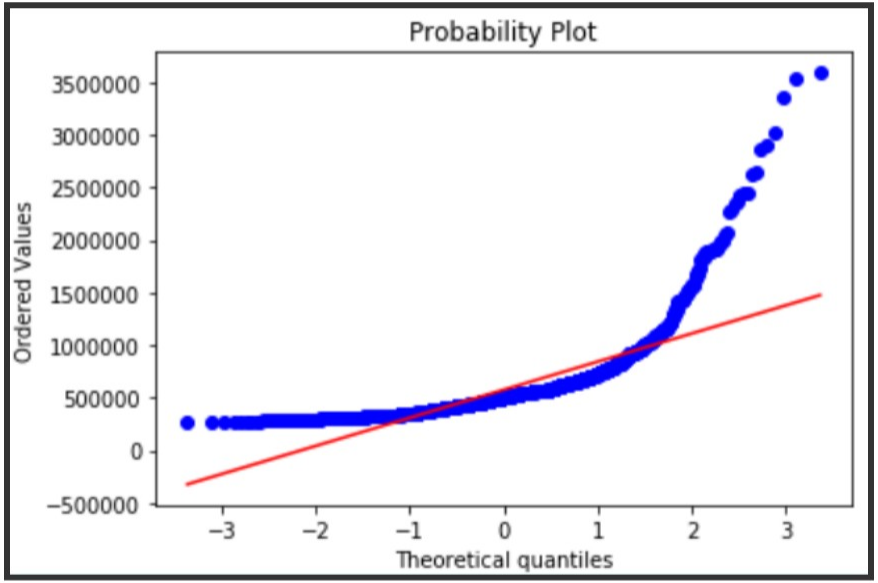
Procedure



Exploratory Data Analysis

Normality testing of Target Variable_Housing Price



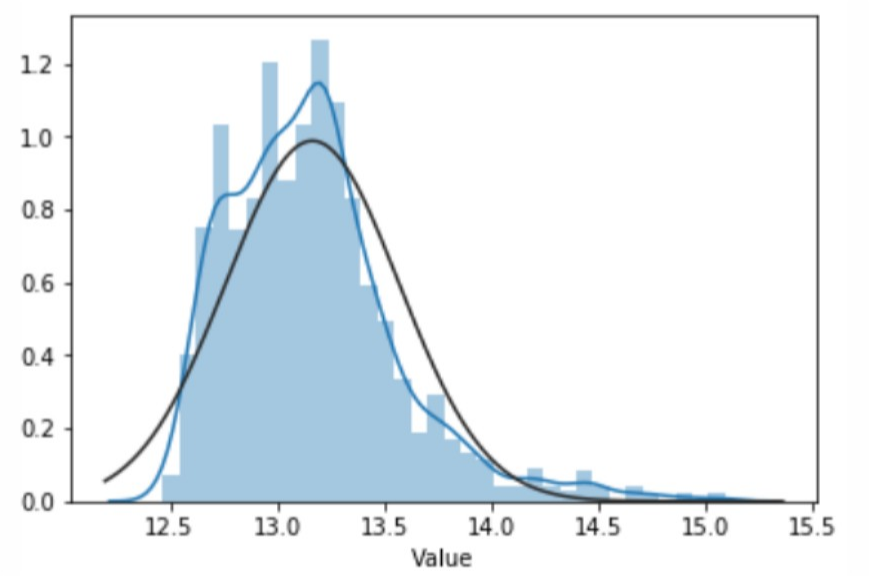


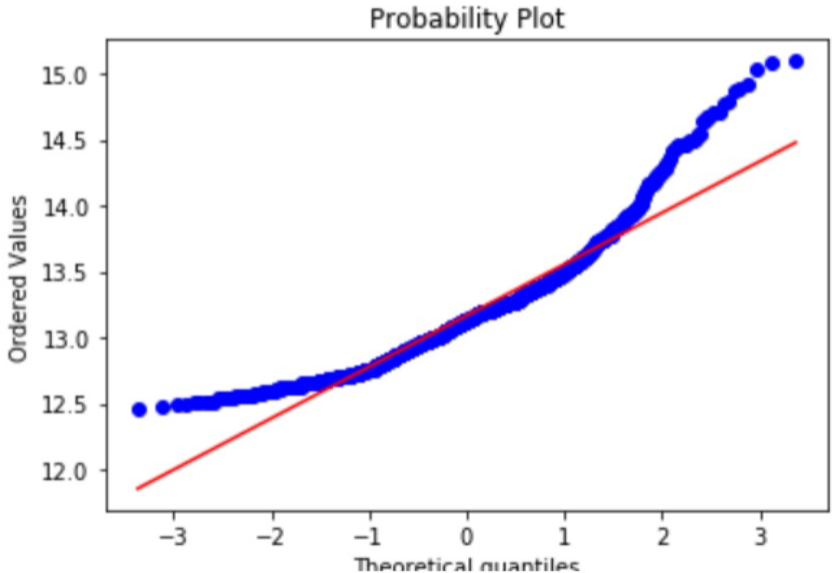
From the above distribution plot, we could see the price values deviate from normal distribution with positive skewness and high peakedness

Calculated skewness :3.790724
Calculated kurtosis: 20.985509

Exploratory Data Analysis

Log_transformation



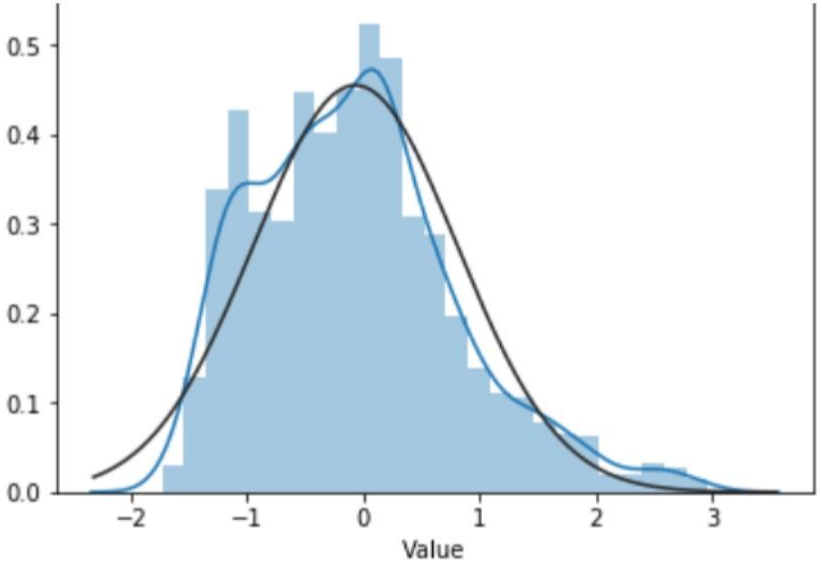


Log-transformation has reduced the skewness by quite a large margin. However, there still seem to be a few outliers at the outer range

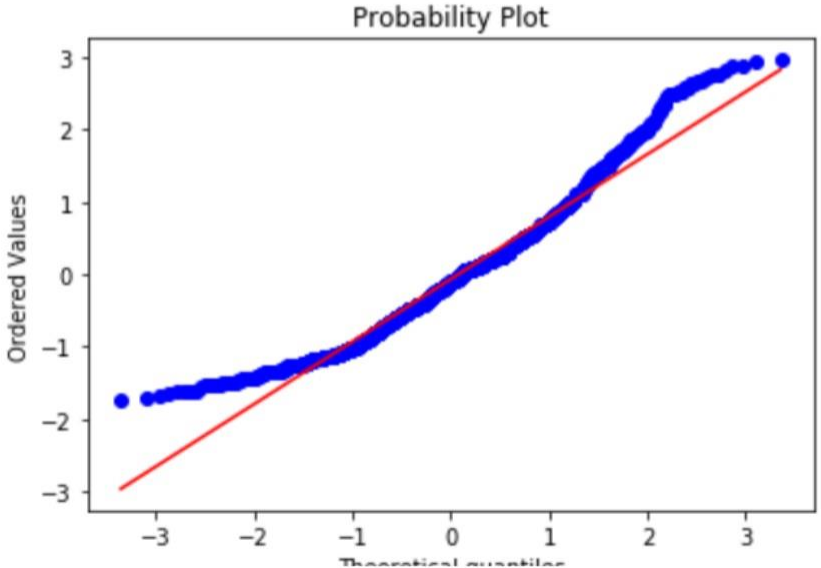
Calculated skewness :1.203352
Calculated kurtosis: 2.357252

Removing the outliers

1.Normalise the data using standardscaler()
2.The data point will be considered as outliers if its z score is greater than three.



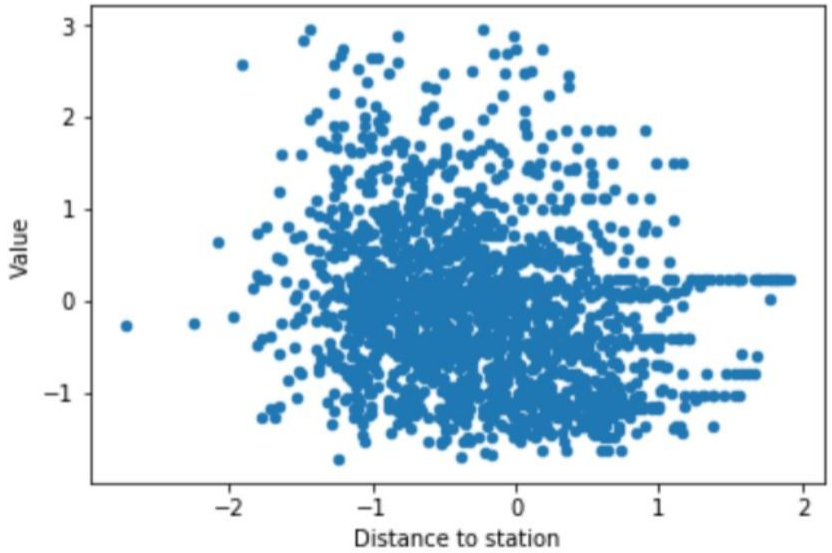
By removing the outliers, we have greatly reduced the skewness and kurtosis.



Calculated skewness :0.674409
Calculated kurtosis: 0.391779

Exploratory Data Analysis

Association between target variable and 1st candidate feature variable (Distance to Station)

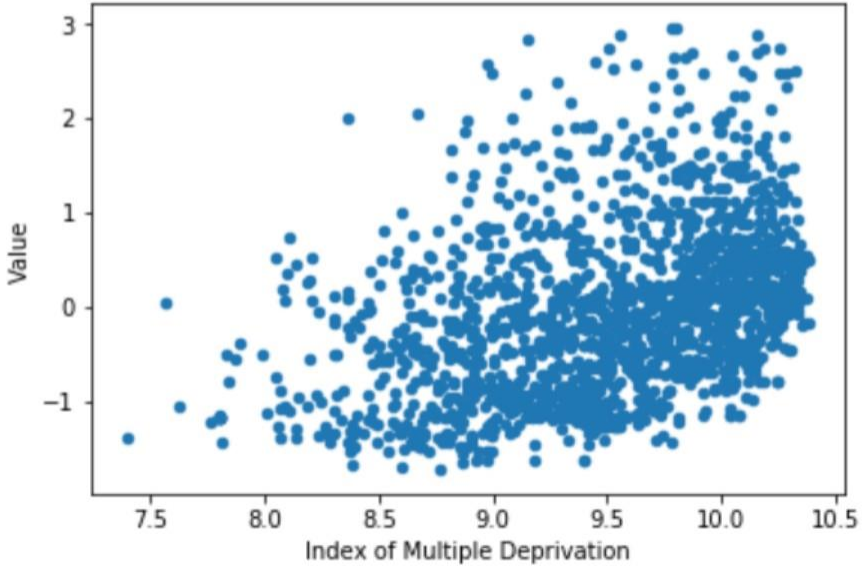


slope:-0.17426126742054926
pvalue=2.340117754964424e-20

Distance to station and London housing price are negatively and weakly associated. The association has reached statistical significance. Therefore, it will be included at the modeling stage.

Exploratory Data Analysis

Association between target variable and 2nd feature variable (Index of Multiple Deprivation)

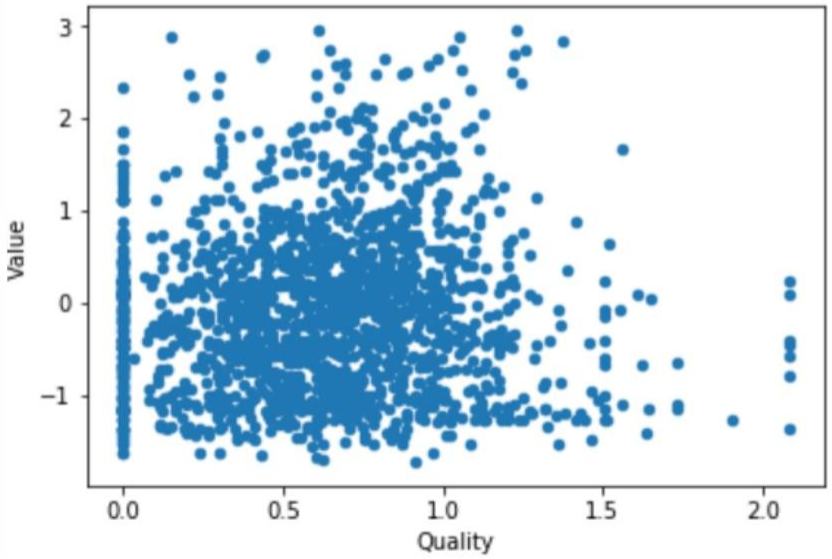


slope=0.2546380349955337
pvalue=3.652040757425847e-66

Deprivation index and London housing price are positively and weakly associated. The association has reached statistical significance. Therefore, it will be included at the modeling stage.

Exploratory Data Analysis

Association between target variable and 3nd feature variable (Quality Rating)

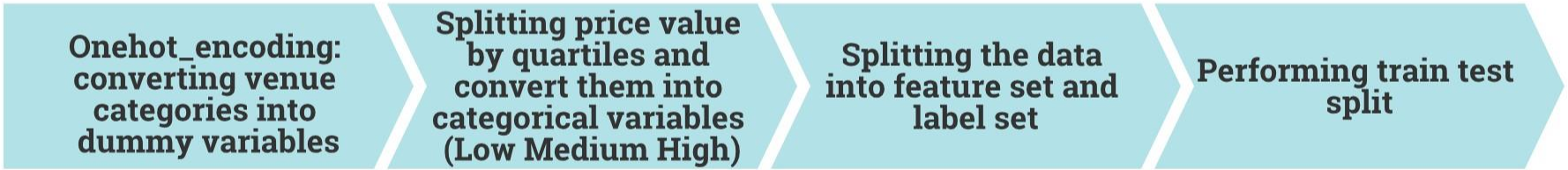


slope=0.026379471806030147
pvalue=0.011371166340648218

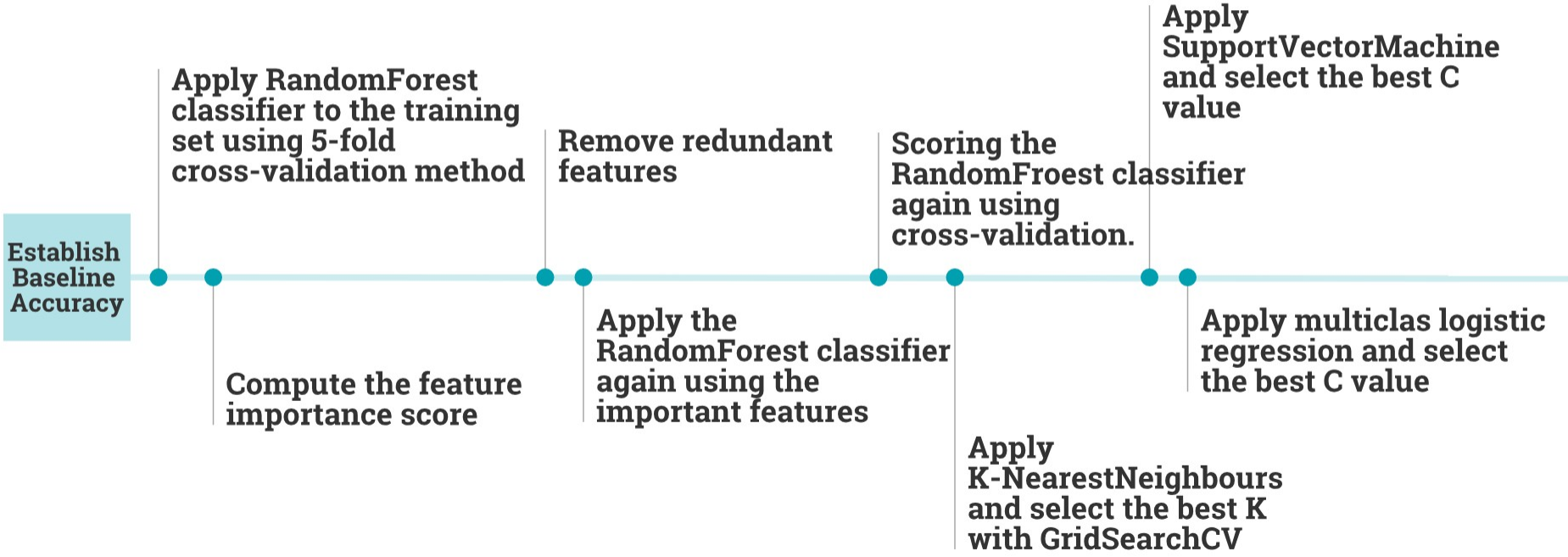
The association between quality rating and London housing price are extremely low, and it did not reach statistical association. Therefore, it will be excluded at the modeling stage.

Further Data Sourcing

Data Preparation for Machine Learning



Model Building



Model Building

- **Rationale for using Random Forest instead of Decision Tree**

  Although decision tress are more interpretable, they are prone to overfiting and tend to have lower predictive accuracy. Random Forest overcome this problem by splitting the bootstrapped training sample using random subset of predictors as split candidates.

- **Null Accuracy: The accuracy that could be achieved by always predicting the most frequent class**

  Calculated baseline accuracy: 0.338285

- **Rationale for using accuracy score as the evaluation criteria at the model building stage:**

  As shown on the left hand side, the three classes are quite balanced. Accuracy score works well on balanced data. Therefore, it is appropriate to be used as the scoring method for cross-validation.

  ROC curve is not provided as it is most suitable for binary classification problems.

```
#Checking the class balance
ld_onehot['PriceCategory'].value_counts()
```

```
medium    5941
low       5932
high      5898
Name: PriceCategory, dtype: int64
```

Model Building

Applying Random Forest to the entire training set

```
#Randomforest __Cross-validation with all feature variables
rf=RandomForestClassifier(n_estimators=1000, n_jobs=-1)
rf_accuracy_1=cross_val_score(rf,x_train_1,y_train_1,cv=5,scoring='accuracy').mean()
rf.fit(x_train_1,y_train_1)
rf_accuracy_1
```

```
0.9684119492826323
```

```
len(London_venues['Venue Category'].unique())
```

```
454
```

When all features are included, the random forest model yielded reasonably high 96.84% accuracy. However, there are 454 venue categories, it is likely that not all of them are relevant in predicting housing price categories. We will use Random Forest's feature importance measure for feature selection.

Justification for using Random Forest for Feature Selection

- **Potential alternative: Recursive Feature Elimination**

  --- Eliminates worst performing features on a model one after one until the best subset of features are known.

  --- Could be combined with cross-validation which also computes the best number of selected feature (Sklearn)

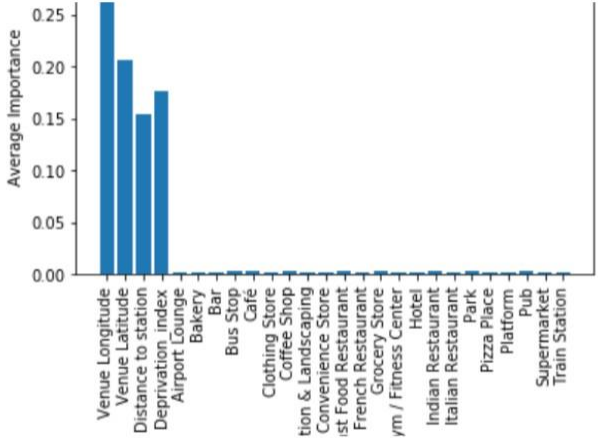
  --- Superior to the arbitrary cut-off point adopted in the current study.

***However, since it is computationally very expensive, Recursive Feature Elimination is not suitable for large number of features which is the case of the current study. Therefore, the use of Random forest feature importance measure is justified.

Model Building

Feature Selection: Feature importance measure calculated using 5 fold cross-validation

Venue Longitude	0.309194
Venue Latitude	0.205520
Deprivation_index	0.176735
Distance to station	0.153510
Park	0.003367
Fast Food Restaurant	0.003133
Pub	0.003001
Café	0.002787
Bus Stop	0.002679
Grocery Store	0.002666



As we could see, the feature importance values dropped quite significantly towards the end. We shall an arbitrary cut-off threshold to remove the redundant variables.

Model Building

Scoring Random Forest with selected features (importance index>0.0015)

```
#Cross-validating the randomforest with the new selected features
rf.fit(x_train_2,y_train_1)
rf_accuracy_2=cross_val_score(rf,x_train_2,y_train_1,cv=5,scoring='accuracy').mean()
rf_accuracy_2
```

```
0.990620937565063
```

The model performance increased by over 2% as result of feature selection. Now we will apply KNN, SVM and Logistic regression to the selected feature sets.

Model Building

K-nearest Neighbor

```
#Using GridSearchCV to selct the number of K
grid_params={'n_neighbors':[1,2,3,4,5,6,7,8,9,10]}
gs=GridSearchCV(KNeighborsClassifier(),
                grid_params,
                verbose=1,
                 scoring='accuracy',
                cv=5)
knn=gs.fit(x_train_2,y_train_1)

#Accuracy score_KNN
print(knn.best_params_)
print(knn.best_score_)
```

```
Fitting 5 folds for each of 10 candidates, totalling 50 fits
{'n_neighbors': 1}
0.8404861944777912
```

Cross-validation on the training set: KNN model yielded the best model performance when k=1.

Accuracy=84.95%

Model Building

Support Vector Machine

```
param_grid={'C':[0.1,1,10,100]}
gs_svc=GridSearchCV(SVC(decision_function_shape='ovo'),
                    param_grid,
                    scoring='accuracy',
                    cv=5)
svc_1=gs_svc.fit(x_train_2,y_train_1)
#Best C value
print(svc_1.best_params_)
#Mean accuracy score of the selected C
print(svc_1.best_score_)
```

```
{'C': 100}
```

One-against-one_Accuracy: 59.75%

```
param_grid={'C':[0.1,1,10,100]}
gs_svc=GridSearchCV(svm.LinearSVC(),
                    param_grid,
                    scoring='accuracy',
                    cv=5)
svc_2=gs_svc.fit(x_train_2,y_train_1)
#Best C value
print(svc_2.best_params_)
#Mean accuracy score of the selected C
print(svc_2.best_score_)
```

```
{'C': 0.1}
```

One-against-the rest_Accuracy:47.12%

Unlike KNN and RandomForest, SVM is intrinsically two-class. There are two ways in which we could modify the algorithims for multi-class prediction:the SVC 'one-against-one' approach and the LinearSVC 'one-vs-the rest' approach.

For our case, the one-against-one approach yielded better accuracy score. Therefore, it is selected for the later model comparison stage.

Model Building

Logistic Regression

```
lg= LogisticRegressionCV(cv=5,multi_class='multinomial')
lg=lg.fit(x_train_2,y_train_1)
lg_accuracy_score=cross_val_score(lg,x_train_2,y_train_1,cv=5,scoring='accuracy').mean()
lg_accuracy_score
```

```
0.49069702487576905
```

Despite the fact that the accuracy score of our logistic regression is quite low, we will not adjust the probability threshold as the dataset is quite balanced.

Model Evaluation

Classification Report

Accuracy Score_rf:0.991885

	precision	recall	f1-score	support
high	1.00	0.99	1.00	1476
low	0.99	0.99	0.99	1520
medium	0.99	0.99	0.99	1440
avg / total	0.99	0.99	0.99	4436

Random Forest

Accuracy Score_knn:0.878269

	precision	recall	f1-score	support
high	0.89	0.91	0.90	1476
low	0.90	0.87	0.88	1520
medium	0.85	0.85	0.85	1440
avg / total	0.88	0.88	0.88	4436

K-Nearest Neighbor

Accuracy Score_svm:0.613774

	precision	recall	f1-score	support
high	0.57	0.82	0.67	1445
low	0.78	0.56	0.65	1503
medium	0.54	0.47	0.50	1495
avg / total	0.63	0.61	0.61	4443

Support Vector Machine

Accuracy Score_lg:0.488859

	precision	recall	f1-score	support
high	0.50	0.59	0.54	1445
low	0.52	0.56	0.54	1503
medium	0.43	0.32	0.36	1495
avg / total	0.48	0.49	0.48	4443

Logistic Regression

The prediction performance of all the above models exceeded the null acuracy on both cross-validation set and the test set, with Random Forest model yielded the highest accuracy and F1 score

Result Evaluation

- **Possible explanation for the discrepancy in model performance (lower F1 and accuracy score of Linear SVM and logistic regression)**

  Both Linear SVM and logistic regression are linear classifiers, it is possible that the decision boundary of the current dataset is non-linear as the both non-linear classifiers (KNN and RandomForest) performed optimally.

- **Why KNN yielded the best predictive accuracy when K=1?**

  when K=1, which is often considered as a low bias but very high variance model, which means it might be sensitive to outliers which consequently lead to overfiting. However, in our case, the outliers are largely removed. It is possible that K=1 is the optimal parameter as it also yielded high accuracy and F1 score on the test set.

Discussion

- **Linking back to the 1st research aim:**

  Consistent with studies conducted

  - in Onondaga County (Yoo, Jungho & Wagner, 2012)

  - Saint-Petersburg (Antipov & Pokryshevskaya, 2012)

  - Santiago (Masías et al., 2016)

  Strengthened the evidence for the superior predictive performance of RF for explaining variances in housing price.

- **Linking back to the 2nd research aim:**

  The housing price of a given set of London geographical coordinates could be predicted based on its distance to the station, deprivation index, and the venue categories and geographical coordinates of its nearby properties.

  Order of predictive strength:

  Venue Latitude > Venue Longitude > Deprivation Index > Distance to Station > Venue categories

Possible Explanations for the Insignificant Predictive of Venue Categories

- **1st:**

  Weak relationship between housing price categories and its nearby venue categories.

- **2nd:**

  The venue category data is limited to its functional categories that provide little information about its associated expenditure categories, which could potentially provide more useful information about housing price.

Recommendations Based on the Research Findings

- **High predictive power of geographical coordinate data:**

  Potentially be used for overseas property investors who are not familiar with London geography to have a basic understanding of the housing price variations.

- **Low predictive power of venue categories:**

  Future research that moves beyond the nearby categories that solely reflects its functional utility and focuses on the ones that reflect household expenditure level is encouraged.