Home Value Estimates

Charles Hyland 450411920

Bernice Zhu 460294434

Yiran Jing 460244129

Table of Contents

Part 1: Introduction and formulation of the business problem

The day-to-day policy decisions made by city government departments revolve heavily around knowledge of the property market. These have a direct impact on the amount of revenue that the government can collect so that it can continue to provide services to the community. A major portion of state and local governments' revenue is based on collecting property taxes and stamp duties. These must be calculated with knowledge of the current market prices. Current property prices are also a telling indicator of important demographic information that city governments can then use to make public policy decisions. For instance, property prices are indicative of the socioeconomic status of a particular area's residents, their age and their welfare needs which must often be provided by the public sector. Additionally, the feasibility of major construction infrastructure projects in a government area will turn on the government's capability to keep a close eye on the property market. This knowledge equips governments to negotiate with property owners and developers.

The pressing need for governments to remain closely aware of the state of the property market must be balanced against the economic reality that properties are illiquid assets; only a small fraction of properties will be sold at a given time, with no directly available data for most houses. We therefore seek to investigate the problem of predicting house prices for all properties within a town, to give the government estimations of the price that any particular house would be able to fetch if sold now (September 2017), even if that property is not placed on the market this year.

We have used regression models to predict the sale price (response variable) as a function of various predictors. These models were trained on a sample of 804 observations of houses sold this year and the prices they were sold for. We worked with a set of 78 potential predictors, which can be classified into the following categories:

- Structural variables (the number of rooms in each house, the condition of the house and materials, materials used to construct the house, roof, availability of air-conditioning, pools, garages)
- Neighbourhood variables (where was the house located in this town?)
- Environmental variables e.g. the proximity to various street conditions
- Other miscellaneous variables e.g. type of sale

The predictive task involved two main questions we aimed to address:

1. ***Model selection:*** Which estimation model was the most appropriate, taking into account our ultimate goal of predicting population data with only sample data available? This involved considerations of bias-variance trade-off and each model's performance using an absolute loss criterion.
2. ***Variable selection:*** Which predictors should be selected for use within the model we have chosen? This task was approached from the perspective of straddling the bias-variance trade-off to reach an optimal level of model complexity to give unbiased predictions without overfitting the model to the training data.

The success measure of our solution is trying to minimize the metric known as the mean absolute error (MAE). In layman's terms, this is simply the average of the error between our model's predicted sale price and the actual sale price. The formulaic interpretation of this is seen below:

$$MAE = \frac{1}{n}\sum |y_i - \widehat{y_i}|$$

Again, this is simply looking at the average of the absolute difference between predicted sale price and actual sale price.

Part 2: Exploratory data analysis and feature engineering

This section deals with our first issue of going through the observed dataset, cleaning it and preparing it for analysis by creating or transforming variables.

1. Intuitive analysis of the most important predictors

The traditional approach for conducting exploratory data analysis and feature engineering begins with an intuitive, manual identification of variables which are believed will strongly influence the value of the response. These have been identified below:

- **Overall quality of the house**: Intuitively, this is one of the first considerations which a potential home buyer will take into account when deciding whether to purchase, and how much they are willing to pay for the house.
- **Overall condition of the house**: This is a variable that is closely linked to the overall quality of the house, and is therefore also one of the most important determinants of sale price.
- **Total Basement Square Feet**: The area of the basement can determine whether the potential home buyer will have extra storage space and/or living space in their house, and having this extra space often means that the potential homebuyer will be willing to pay extra, increasing the price of the property.
- **Total area of the first floor**: The first floor is widely recognised as containing the most integral parts of a house, most typically containing the living room, kitchen, and other common living areas. Even though some houses may not have any other floors, all houses will have a first floor, and so the area of this space is intuitively an extremely important determinant of housing price. However, we note that this may be closely related to total basement square feet, as the first floor is constructed above the basement and therefore there may be a high positive correlation between these two variables.
- **GarageCars** (the number of cars that the garage can contain): People are often willing to pay more for a house purely because of the availability of extra parking space, to give them more privacy, allowing them not to infringe on street parking or find less convenient parking spaces with respect to their house location.
- **GarageArea** (the size of the garage in square feet): This variable is closely related to that of GarageArea, however is a slightly different measure, as it includes the area of a garage that may be used for storage or other uses, not necessarily for car parking. People are intuitively willing to pay more for this extra storage area. Because of this close relationship, we expect there to be a high positive correlation with GarageCars.
- **Gr Liv Area** (the above grade (ground) living area square feet): This is a total measure of the amount of living space possible, and is clearly and intuitively an important determinant of the price a house will fetch on sale.

These variables have been identified through research on what factors constitutes housing prices and was also arrived at through intuition. As a proxy, through research on real estate websites, we noticed that these variables were publicly stated alongside the going price for a house, thereby suggesting that these factors are what homebuyers are interested in when purchasing a house.

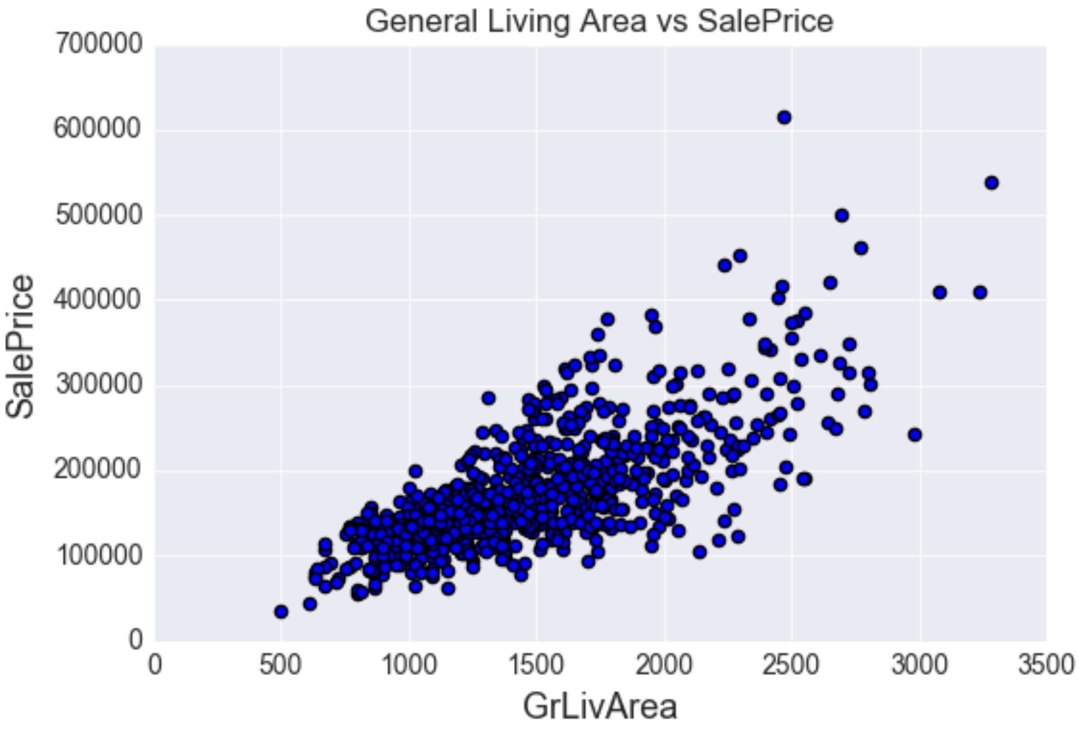
2. Removal of outliers

We examine whether there are any possible outliers that may affect the regression analysis. Outliers (extreme observations) have the potential to turn the regression line towards itself, so that the regression captures noise rather than the true pattern in the dataset (and the true pattern within the population).

The preliminary documentation for the dataset (written by the creator of the dataset for this town of Ames), noted the possibility of identifying outliers through plotting the variable of GrLivArea against sale price. Particularly, he noted that the presence of 3-4 houses with GrLivArea of above 3500, however having extremely low sale prices.

In light of this, we plotted this relationship below. The plot shows no apparent outliers, indicating that these observations have been removed from the training dataset prior to our accessing this for analysis. Other plots of other variables were also examined but also did not warrant the removal of outliers. Therefore, no observations have been removed on the grounds of being outliers.

Figure 2.1- Relationship between general living area and saleprice



3. **Deletion of variables and replacement of variable values**

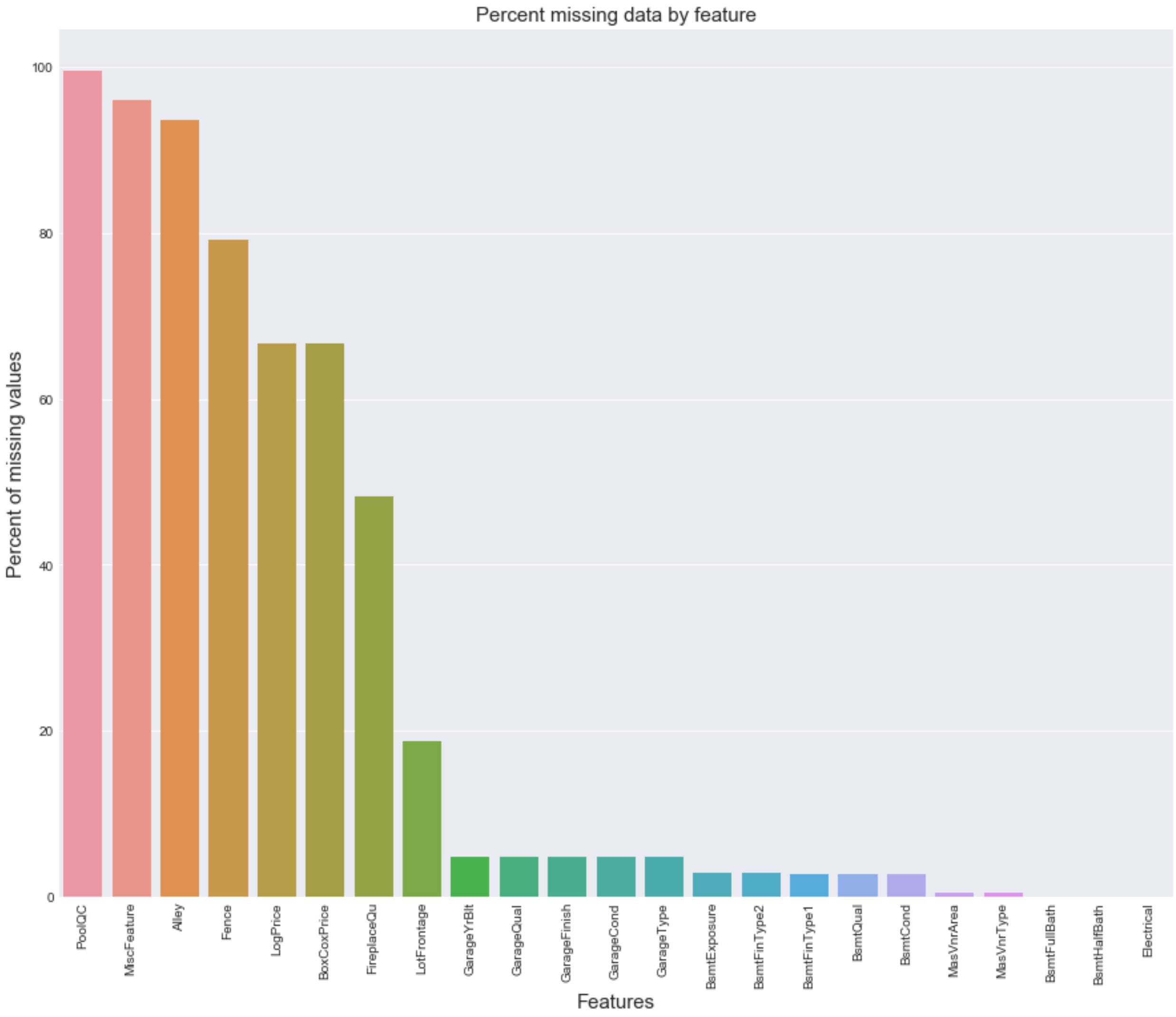
We deleted variables where there was little variation in the observations. This is because we do not have sufficient information to adequately fit or estimate a model to explain the variation in the population data as a function of that predictor. Furthermore, this helps to reduce the dimensionality of the dataset and also assist our model in ensuring they do not pick up random noise.
The following variables were deleted:

- All variables related to the pool, as there are only 3 observations out of the 804 in the dataset which have a pool
- Fence quality, as nearly 80% of the observations have that value missing
- Alley, as nearly 94% of the observations are missing this information.
- GarageYrBlt and GarageFinish, as these are highly correlated with the other Garage variables, and keeping all of them may lead to multicollinearity problems. The year a garage was built is likely to be closely related to its quality, and its finish is likely to be related to its type, etc.
- LandSlope, as this is very closely related to the land contour, and again would lead to multicollinearity problems.

- Utilities, as all the observations, except one, which was missing a sewer, had all the public utilities available. This is very minimal variation in the data and does not assist in predicting prices due to the homogeneity of the feature.
- Kitchens above grade: Deleted as this is highly related to the KitchenQual variable, which is arguably more informative than a binary response to the question of whether a kitchen is above grade.

It is worth noting that the variables relating to miscellaneous features were not deleted even though over 96% of the dataset had missing values for this variable. This is because the missing values are information, rather than lack of information; indicating a house has a particular extra feature, such as shed or extra garage, which is commonly a structural feature that is extremely relevant to house prices.

Figure 2.2- Plot of missing ratio (percentage of observations with missing values) for variables



We also note that there are variables where there were only a few observations with missing values. Where an empty observation merely meant an absence of information for that particular observation e.g. a house just does not have a garage (and so does not have a GarageArea value), then we replaced the empty observation with the mean or median of the variable, depending on the skew of the variable's distribution (with substantially skewed distributions, we used the median, but if the variable distribution did not substantially deviate from normal, we used the mean). In contrast, where a missing observation did convey some information e.g. that the house was only one storey and so did not have a 2nd floor, we used 0 as the value of the variable for that particular observation.

The replacement of values, rather than the alternative of deleting observations, was necessary to ensure that the models would capture the variation in the data for these particular observations. This had the effect of giving our models the maximum number of observations on which to fit or estimate, in an effort to more closely capture the patterns from the population data.

4. Use of dummy variables and binning

Remaining nominal and ordinal predictors were converted into dummy variables. It was necessary to deal with ordinal predictors in this manner, as keeping these in ordinal scale implies that the difference between different levels in the predictor is a consistent one i.e. the difference between a score of 2 and 3 is the same difference between a 6 and 7, or the difference between 'good' and 'average' is the same as the difference between 'good' and 'excellent'. While the construction of dummy variables does lead to higher dimensionality, increasing the number of predictors and the complexity of the model, this was necessary to prevent incorrect information from being incorporated into the model.

However, we note that we chose to exempt OverallQual and OverallCond variables from this treatment. These were retained in discrete form for ease of interpretability directly from a regression output (drawing upon two identical houses, if we increase the quality of 1 house by one point, we can expect on average an *x* difference in the sale price).

Furthermore, we employed a technique of binning for categorical variables if 2 or more dummy variables pertaining to the same category had less than 10 observations. These were instead combined to create a new dummy variable for that categorical predictor, labelled 'other'. This was done in an effort to reduce the model complexity by reducing the number of predictors that the model had to account for, in turn reducing the variance component of the prediction error. Furthermore, this helps to prevent the model overfitting on random noise which we do not have much information on.

The following categorical variables underwent the binning process, with dummy variables for which there were less than 15 observations combined together into the new 'other' dummy:

- **MSZoning**: only retaining residential low density and residential medium density as separate categories, combining observations from all other categories into an 'other' category
- **RoofStyle**: only retaining Gable and Hip as separate categories, all other categories combined into the 'other' category
- **RoofMatl**: only retaining RoofMatl_CompShg as a separate category, combining all other categories as RoofMatl_other
- **MiscFeature**: only retaining MiscFeature_Shed, combining all other categories as MiscFeature_Other, as overwhelmingly the most common miscellaneous feature that houses had was a shed

5. Transformation of variables

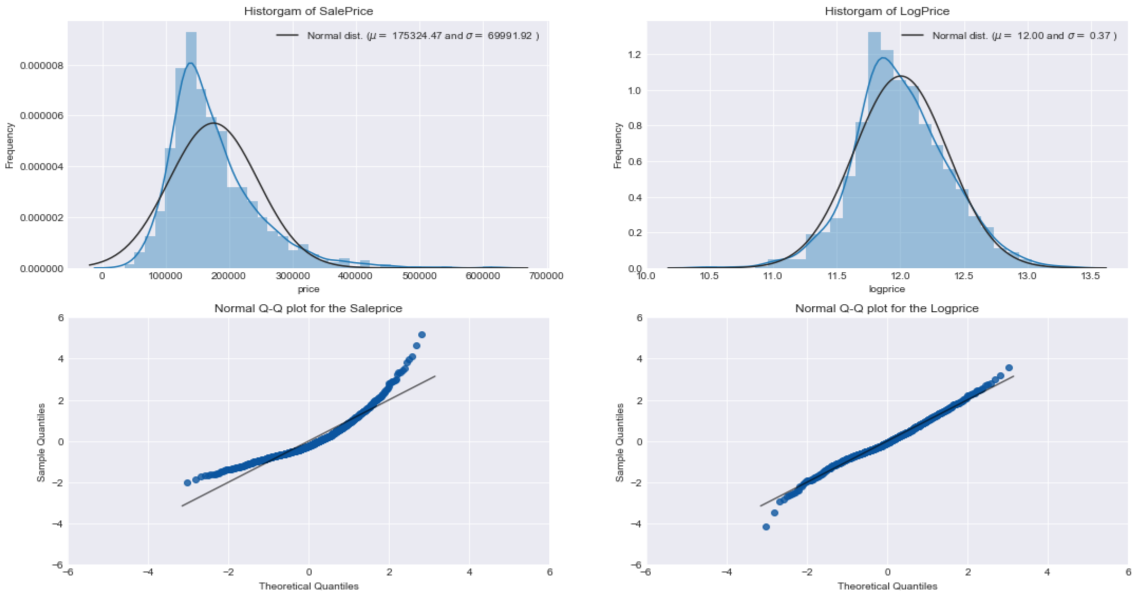
The models we have used in analysing this particular dataset, such as the kernel ridge regression, are designed to locate and pick up on nonlinear relationships between predictor random variables. Therefore, in favour of a less tedious process of feature engineering, we did not rigorously examine the scatter plots of individual predictors against the response to incorporate nonlinear effects e.g. splines, polynomial regression effects, or interaction effects.

This, however, does not necessarily mean that a transformation of the response variable of sale price should not be considered. In fact, we discovered that the transformation of the response from Price to log(Price) would be more appropriate for our estimation and prediction tasks.

Studying the univariate EDAs for the original variable and the transformed variable, we see that the distribution of the untransformed variable is significantly right-skewed. This suggests that regression techniques, such as OLS and its regularised forms, may not be suitable, as these are founded upon approximately normal data. After a log-transformation was applied, the distribution of the response variable was approximately normal.

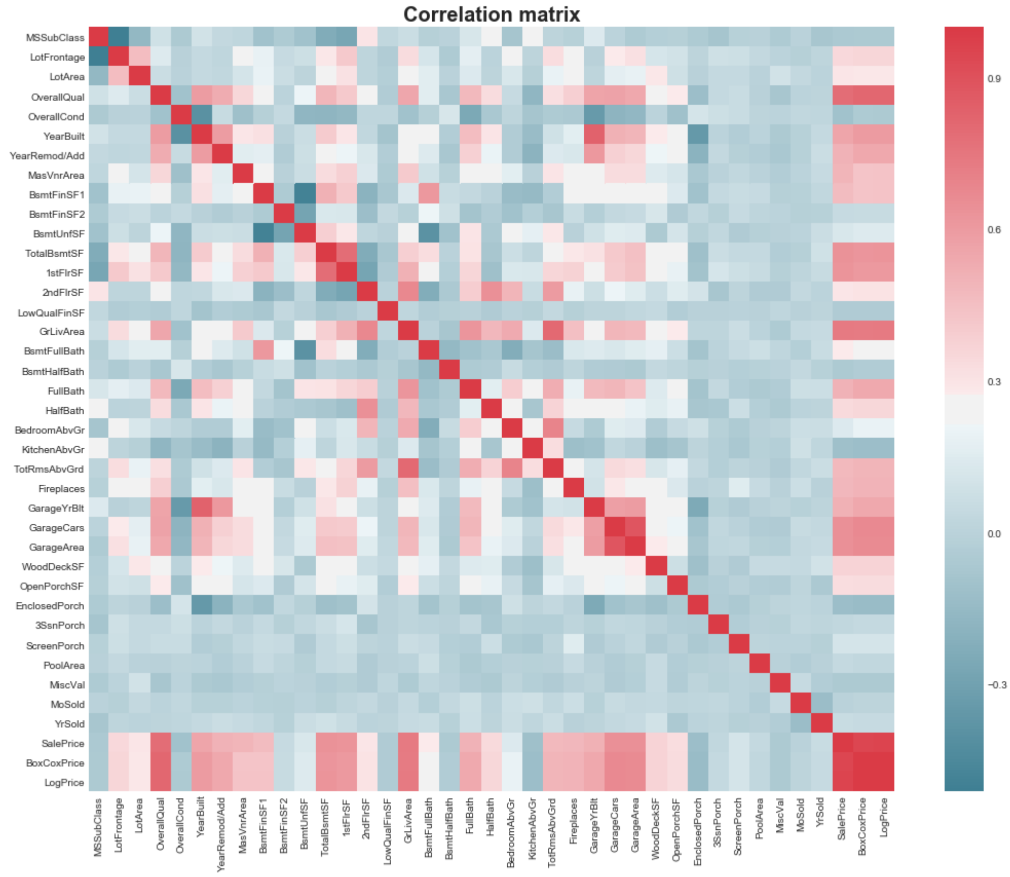
We note that performing a box-cox transformation of the SalePrice variable could also deal with the issue of skewness. In the interests of interpretability, however, we have adhered to the log-transformation throughout our analysis.

Figure 2.3- distribution of SalePrice and its log transformation



6. Combination of predictors

Figure 2.4- Correlation matrix of variables



The above correlation matrix was constructed with all the continuous predictors, and including the transformations of the SalePrice variable. It shows that multicollinearity is not too much of a problem here (as evidenced by the general lack of dark reds or dark blue squares). However, we note two particular areas of predictors where multicollinearity may present an issue. These are the variables to do with the areas of the floors (particularly the basement and first floor), and the variables to do with the garage size (the number of cars that can be parked, and the area of the garage in square feet).

With respect to the Garage variables, we chose to leave these as they were and include both in our analysis. This is in light of the different information that these variables convey, and the different uses of the garage that a home buyer may engage in. While a garage which allows more cars to be parked inside will clearly have a greater area, we note that a garage can also be used for storage, and because of these different uses, it does not make sense to delete one of these variables, or else to combine these variables in some way.

Some of the arguably most important predictors are the area of the basement, the area of the 1st floor and the area of the 2nd floor. However, it is interesting to note that not all houses will be built over 2 floors. Furthermore, what is often of most interest to a potential home buyer is not the area of individual floors, but rather the total area of the living space available within a house. To account for this real-life phenomena, we combined these three variables into a new variable, called TotalSF, computed by:

$$TotalSF = TotalBsmtSF + 1stFlrSF + 2ndFlrSF$$

An examination of the correlation matrix with each of these original predictors, the newly created predictor and the response shows that the original predictors are highly correlated with each other e.g. the area of the basement is highly correlated with the size of the first floor, intuitively, because these floors are built on top of each other and will therefore be similar in size. The creation of the new variable may have the effect of decreasing the model complexity as fewer predictors are used.

Table 2.1 – Descriptive statistics of square feet variables for the house storeys

	TotalBsmtSF	1stFlrSF	2ndFlrSF	TotalSF	SalePrice
TotalBsmtSF	1	0.79	-0.24	0.80	0.63
1stFlrSF	0.79	1	-0.28	0.77	0.63
2ndFlrSF	-0.24	-0.28	1	0.31	0.29
TotalSF	0.80	0.77	0.31	1	0.83
SalePrice	0.63	0.63	0.29	0.83	1

The correlation matrix shows that the top 5 predictors with the highest correlation with the sale price are OverallQual, GrLivArea, GarageCars, GarageArea and TotalSF (following the combination of the basement, 1st and 2nd floor area variables). It is worth noting that this corresponds with our initial intuitive analysis of the predictors, suggesting that the observations in this dataset closely reflect common logic surrounding which should be the most important determinants of sale price.

Plotting some of these variables, we can verify a strong linear relationship between the identified most important predictors and the response variables.

Figure 2.5 – box plot of OverallQual vs Logprice

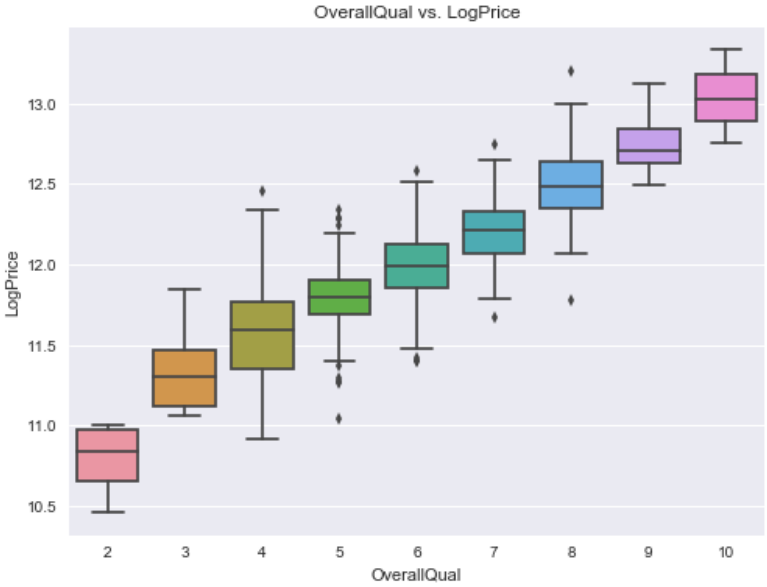


Figure 2.6 – relationship between GrlivArea and Price with their corresponding transformation term

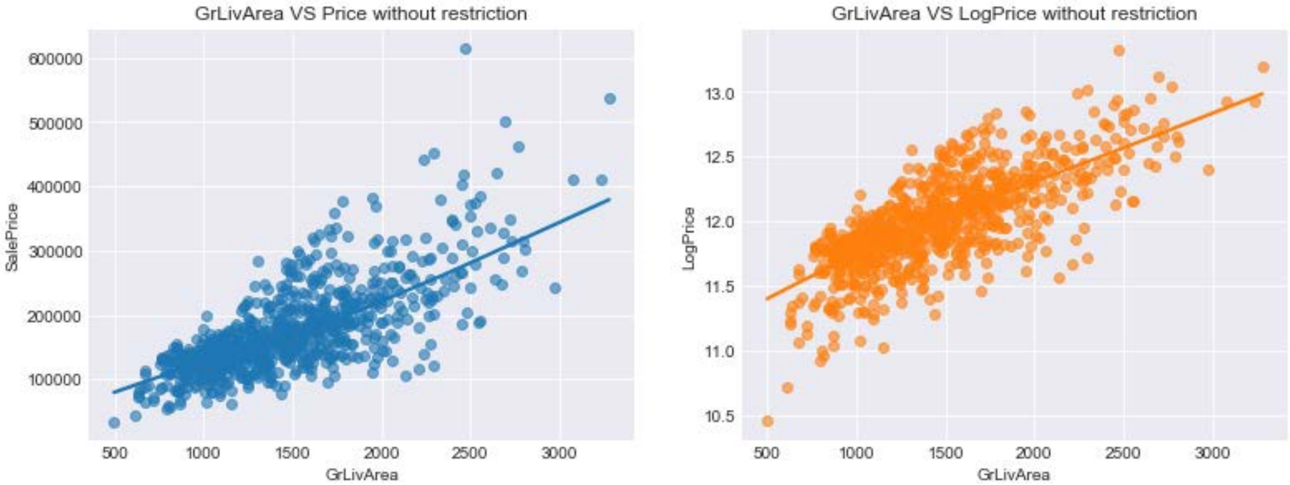


Figure 2.7 – relationship between TotalSF and Price with their corresponding transformation term

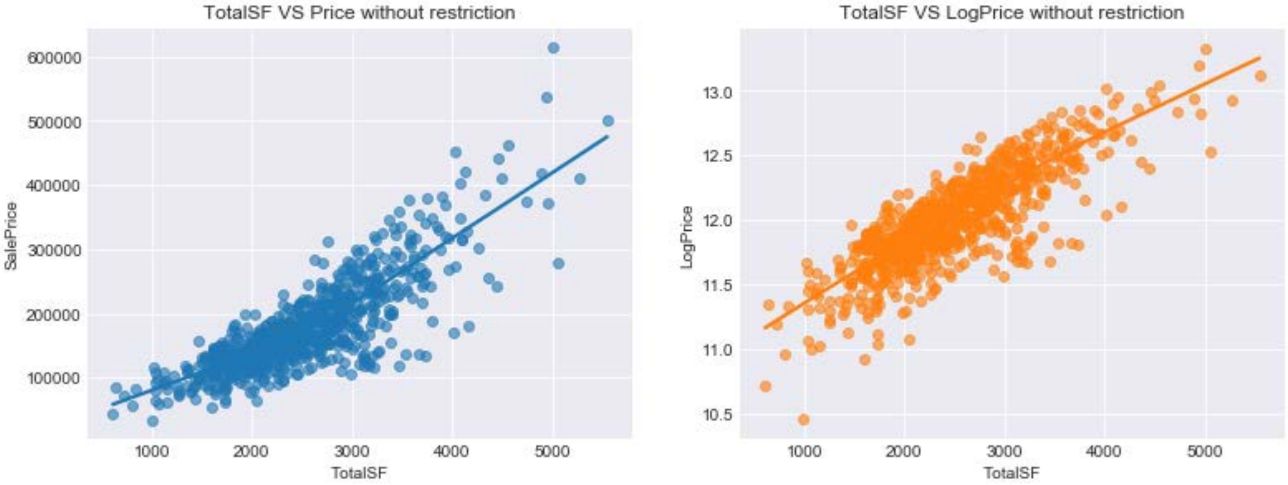
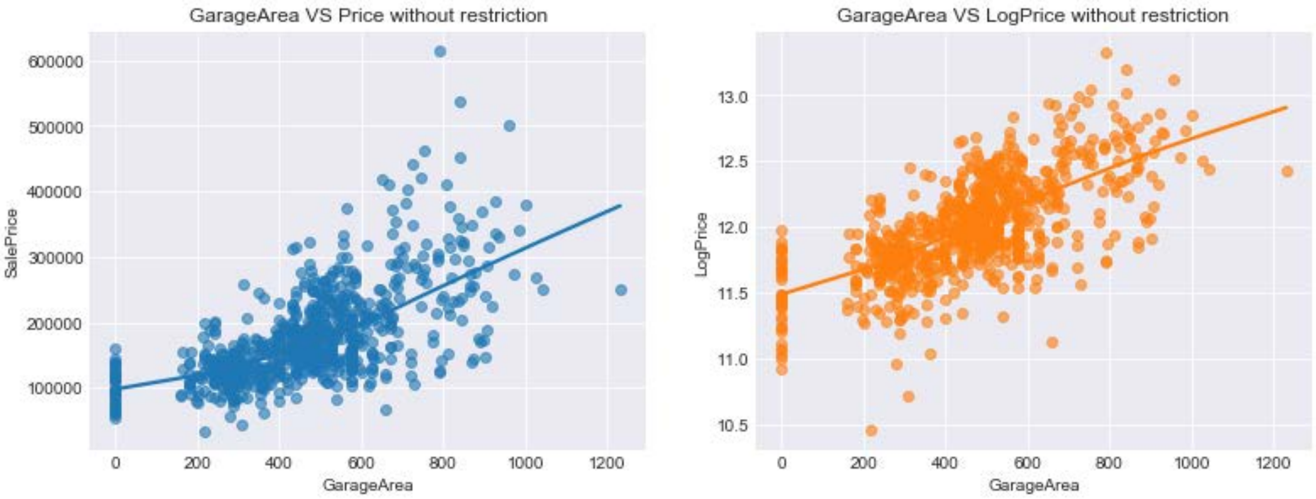


Figure 2.8 – relationship between GarageArea and Price with their corresponding transformation term



However, although our analysis with the correlation matrix suggests the appropriateness of examining linear relationships between the predictors and the response in our models, it is important to keep in mind that the correlation matrix looks at only linear relationships. We must be wary that there may be nonlinear relationships in the dataset, and even if we performed minimal variable transformation in light of the robustness of our models, there may nevertheless be prominent nonlinear relationships that a stronger predictive model would need to be and be able to capture.

7. Time series analysis

Figure 2.9 – Box plot of prices sorted according to year sold

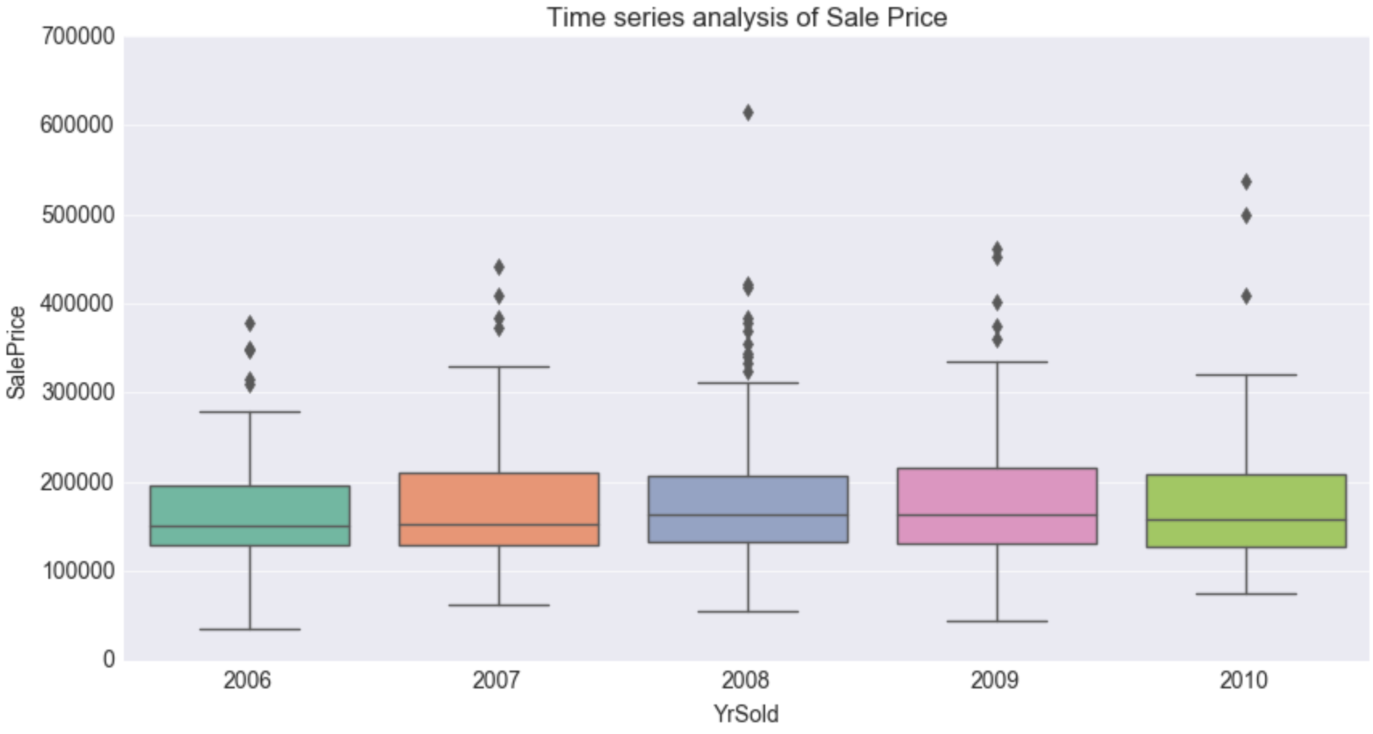
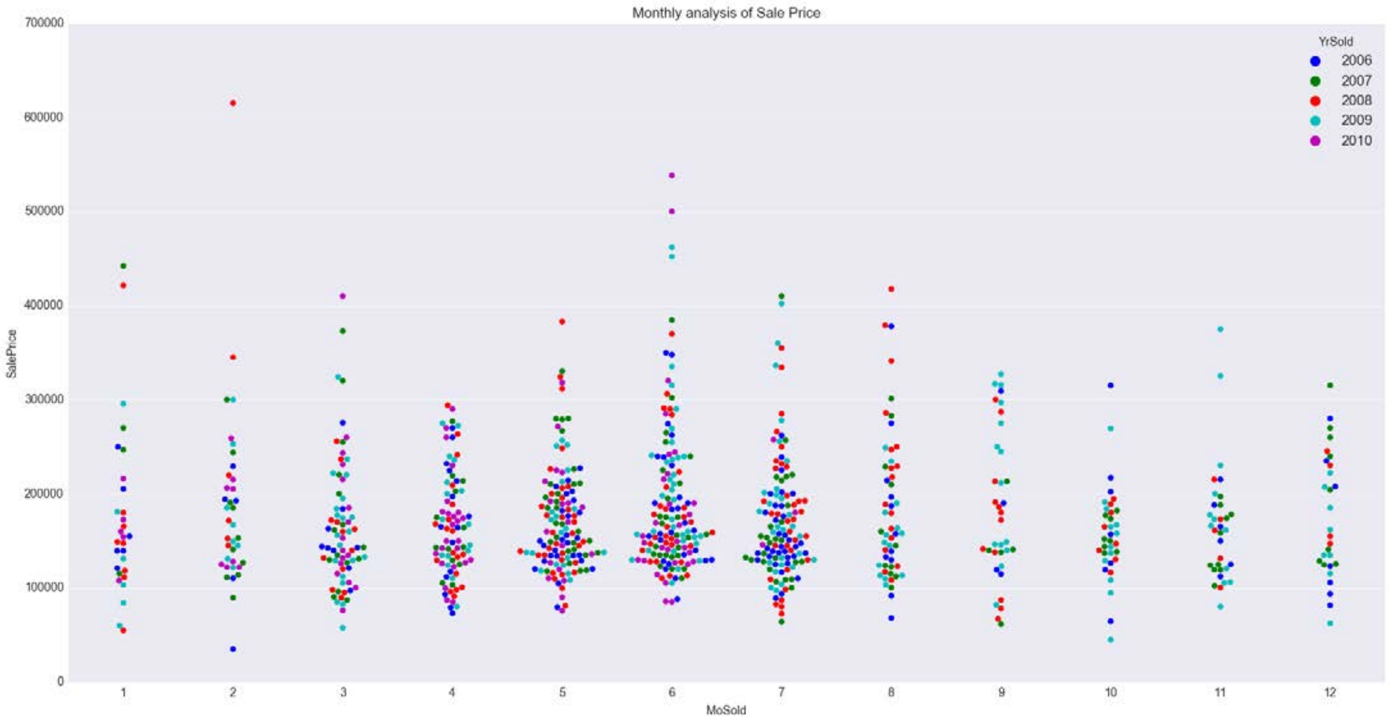


Figure 2.10 – swarm plot of SalePrice values according to month sold, colour-coded by year of sale



We examine the dataset for any potential time series effects, given the presence of variables indicating when each of these properties were sold. The presence of time series effects may violate the regression assumption of independence of observations.

However, the above plots indicate that housing prices remained stable over the 5 years 2006-2010, despite this period including the GFC, during which the USA's economy was severely affected. The anomalous stability of house prices across these time periods indicates there is no time-series effect present in the dataset. A further analysis of the sale price as a function of the month in which a house was sold indicates stability across months. However, we note the concentration of observations in the months between May and July, indicating that house sales are very high in the summer time in the USA (perhaps avoiding the concentration of holidays towards the end of the year).

8. Further diagnostics

There are various other tests that could be performed to fully diagnose and evaluate the dataset. These include the influence plot for outliers, and residual diagnostics for regression assumptions of linearity, normality, homogeneity and independence of assumptions. However, we do not include these here. This is because we will be extensively using non-parametric models in ensemble with parametric models (using the techniques of model stacking and ensembling). These are, by definition, models which have make very few assumptions about the functional form of the patterns in the dataset, and therefore come with very few diagnostic tests which should be performed other than the ones discussed above. Even if we will be retaining the advantages of parametric approaches by also including these in our models, their combination with non-parametric techniques means we only have to make very few assumptions about our data.

Part 3: Methodology

In light of the large number of potential predictors (over 200), and the high degree of complexity of the model if a standard OLS estimation model was used, we turned to techniques that would perform the interrelated tasks of model selection and variable selection with the goal of minimising variance whilst trying to achieve an accurate, unbiased prediction.

In doing so, we arrived at two final models, both of which were a combination of various model selection and variable selection techniques.

<u>Model 1:</u>

Ridge regression

Ridge regression is a modification to the basic multiple linear regression model. MLR estimates the coefficients for the predictors without any restriction on the values of these predictors. While the use of all the predictors at once allows the estimated model to closely fit the training dataset (with the training MAE lowered as predictors are added as more of the variation in the dataset becomes explained by the model), the basic MLR approach risks a high variance in the model's performance on the test dataset as the model is highly complex, with many predictors.

To prevent this problem and reduce variance, the ridge regression model shrinks the size of the estimated coefficients through the use of a selected tuning parameter λ.

The ridge regression model is:

$$\widehat{\beta}_{\text{ridge}} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 + \sum_{j=1}^{p} \beta_i x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

To select the value of the tuning parameter, we used cross validation, selecting its value by considering which had the lowest cross-validation MAE.

Forward selection

Forward selection is a method for variable selection, working to eliminate predictors that the algorithm does not consider relevant to the response. By eliminating certain predictors, we work to reduce the variance in the test MSE that arises from using a high number of predictors.

The forward selection starts with the null model which only contains the constant, and then predicts the p models which only contain one predictor, computing the MAE for each and selecting the one with the lowest MAE. It then considers all the models which add one predictor to the previously selected model, selecting this one and then repeating this same process of adding one predictor at a time until the model with all p predictors is considered. The algorithm then selects the best model amongst the p models according to cross-validation MAE.

LASSO

We also tried the alternative regularisation method of LASSO. Ridge regression is based on the same rationale as ridge, shrinking the value of predictor coefficients to reduce variance, however a crucial difference between LASSO and ridge regression is that LASSO has the variable selection effect, while ridge regression does not. That is, while ridge will lead to smaller coefficient values, these will never be exactly zero, while LASSO will shrink some coefficients to zero (thereby effectively eliminating them from the model). LASSO is particularly effective for datasets which are 'sparse' i.e. there are only a few relevant predictors to the response.

$$\widehat{\beta}_{\text{lasso}} = \underset{\beta}{\text{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_i x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

LASSO actually led to worse results in this case, as this was not a case where we had a sparse dataset. Forward selection suggested that nearly all of the predictors were relevant, therefore ridge was a more appropriate method here. Ridge adequately addressed the problem of high variance as model complexity increased, whilst simultaneously keeping the many relevant predictors in the model.

Random forests

Random forests was another technique employed into the creation of the model. Random forests are extensions of decision trees, whereby multiple decision trees are created through the process of taking a bootstrap sample from the training set. Unlike typical decision trees which allows for splitting along the entire feature space, random forests injects a stochastic element into the algorithm by forcing each decision tree to be constructed based on a randomly selected subspace of the features. This forces different decision trees to be constructed. This will construct more robust predictions as unlike the single decision tree case which can overfit due to some features, the fact that the random forest decision trees has less features to construct trees means that it is much more robust to overfitting.

Boosting

To further improve the models, the technique of boosting was employed. Boosting is a technique whereby 'weak learners' are combined to create a single 'strong learner'. The mechanism behind boosting is that you have multiple rules (weak learners) which must at least have a higher than 50% chance of being able to correctly predict values. Through this, boosting combines the numerous weak learners into a single strong learner through a plethora of methods such as taking the average or weighted average of the weak learners.

Gradient boosting

In particular, the boosting model employed was gradient boosting, whereby the algorithm trains multiple models sequentially. Each new model aims to minimise the loss function through the process of gradient descent, an optimisation technique by traversing the negative gradient of a function in order to find a local minimum of the loss function. Gradient boosting construct learners such that they will be maximally correlated with the negative gradient of the loss function and therefore allow for it to be an easier process to locate the minimum of the loss function.

Extreme gradient boosting

An extension of gradient boosting is known as extreme gradient boosting (xgboost) which was also employed in the final model. Extreme gradient boosting is identical to the gradient boosting algorithm except that now, xgboost uses a more formal approach to regularisation of the model to reduce over-fitting of the data. Furthermore, the underlying implementation of xgboost utilises systems optimisation and therefore allows for extremely computationally intensive calculations and allow for more computations.

Finally, with the 5 models in place, the technique of ensembling was used. Put simply, it is allowing for each model to predict the salesprice and then we combine their predictions into a single prediction by taking the weighted average.

From this, the weighted averages were as follows:

Table 3.1 – Weighted average of 5 models

XGBoost	Gradient Boost	Ridge Regression	Forward Selection	Random Forest
15%	35%	10%	35%	5%

Whereby the weighted averages were calculated through trial and error and by analysing each model's individual performance.

Model 2:

The second model incorporated the technique of model stacking. For model stacking, each individual model through cross validation, predicts values for all observations in the training set. Then each model computes their predictions based on the whole training set for the validation set. Then, a meta-learner or model is then constructed based on the predictions for the training set and attempts to use this information to assign weightings in order to predict the validation set. By looking at each model's prediction for the training set and comparing their predictions to the actual training set prices, the meta model will be able to identify in which scenarios do certain models tend to predict more accurately. The model then assigns different weightings to each model's predictions contingent on their performance. From using the training set predictions and price values, the meta model then calibrates the weightings each model should have for the final prediction in the validation set.

With this, the base models that were found to be the most optimal through cross validation and analysis of the mean absolute error were: Gradient Boosting, Elastic Net, and Kernel Ridge Regression. Furthermore, the meta model was found by iterating through all possible models to be used in conjunction with the base model. The strongest performing model was the LASSO, with a mean absolute error of 12483.72.

The base models we tried were as follows:

Gradient boosting: The underlying rationale for gradient boosting was explained in the previous section as part of the techniques we chose to employ to build the first model.

Elastic net

Elastic net is derived from the combined advantages of ridge regression and lasso regression. It incorporates the same principles of shrinking the coefficients of correlated predictors to reduce model complexity (as seen in ridge regression) and variable selection (as seen in the lasso). The equation for the elastic net regression is:

$$\widehat{\beta}_{\mathrm{EN}} = \underset{\beta}{\mathrm{argmin}} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_i x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \left(\alpha \beta_j^2 + (1-\alpha)|\beta_j| \right)$$

Kernel ridge regression

Kernel ridge regression is a further development of the ridge regression regularisation method (discussed above as part of the techniques of model 1). The algorithm for kernel ridge regression incorporates a further kernel function in addition to the basic ridge regression method. The mechanics of the kernel function are complex and will not be fully elaborated on within this report, however, we note that this is a non-parametric technique targeted at finding a non-linear relation between random variables. This could potentially address nonlinearity between the predictors while offering the flexibility of a non-parametric approach.

Extremely Randomised Trees

This method is quite similar to the random forest method, except even more randomness is added to the process. Similar to random forests, a random subset of candidate variables are selected but then discriminative thresholds are drawn randomly for each candidate feature and then the best randomly-generated discriminative threshold is used as the rule to split the data. This method was employed in order to attempt to offset more variance in the predictions at the cost of adding bias.

Adaptive Boosting

Adaptive boosting fits a sequence of weak learners (which are primarily models that are slightly better than random guessing) on repeatedly modified versions of the data. The final prediction then comprises of a weighted average of all the predictions made by the weak learners. In each iteration of modifying the data, a higher weight is given to observations that were predicted incorrectly which therefore allows for them to have another chance at being estimated.

From our model stacking, we then also used our ensemble technique from earlier and took a weighted average of the ensemble model and the stacking model's predictions in order to compute a final prediction value.

Overall, a lot of the models we employed (adaptive boosting, extremely randomised trees, kernel ridge regression, gradient boosting), were nonparametric in nature. These are extremely flexible models to the nature of the training dataset. However, some of the models e.g. the lasso and ridge regressions, which we attempted to use on the data were parametric, and are modifications of the ordinary least squares estimator which relies on approximately normally distributed variables and error terms to achieve reliable results. Therefore, even with the flexibility of our nonparametric models, it was necessary to take a log transformation of the price variable to ensure that we could still use parametric methods to our advantage.

Part 4: Analysis of results

In this section, we evaluate the performance of our chosen models. We note that the loss function for this task is an absolute error loss rather than a squared error loss, therefore, we use the mean absolute error as the measure for model evaluation.

<u>Model 1:</u>

The following table compares the cross-validated mean absolute error, and test R squared for each individual model approach we took, as well as the values for the final model achieved by ensembling.

Table 3.2 – model performance of model 1 base models and final combination

Model	MAE	R-square
Forward Selection	14042.658	0.884
Gradient Boosting	12968.305	0.923
Adaptive Boosting	18182.479	0.861
Extreme Gradient Boosting (XGBoost)	13619.232	0.915
Ridge Regression	12901.659	0.892
Random Forest	18237.747	0.857
XGBoost, GBoost, Ridge, Forward Selection, Random Forest. Weighting: 15,35,10,35,5.	12430.890	0.934

The results clearly show that the ensemble technique was substantially better for prediction as evaluated on the validation set. The ensemble technique dramatically improved our score for R-square whilst reducing the MAE.

Intuitively, we would expect that the combination of different models to achieve a better result in predictive performance as estimated by the validation set. This is because a combined approach prevents against the overfitting of any particular individual model to the training dataset, which could ultimately result in high variance in prediction.

<u>Model 2:</u>

The cross-validation scores for each of the preliminary models we tried with different hyper-parameters are given below:

Table 3.3 – cross-validation score of preliminary models

Model	CV - MAE
LASSO	0.0718
Extra Randomised Trees	0.0986
Gradient Boost	0.0795
Adaptive Boosting	0.105
XGBoost	0.0805
Kernel Ridge Regression	0.0793
Elastic Net	0.0719

We chose the models with the lowest cross-validation mean absolute errors: gradient boost, kernel ridge regression, elastic regression, LASSO and extreme gradient boosting.

We then constructed ensembles of different models to locate which particular ensemble we should utilise for our base models. This was achieved by generating a prediction with each of the base models still under consideration, then taking the average mean absolute error of each combination of these base models shown below:

Table 3.4 – ensemble model performance

Ensemble Model	**CV - Mean Absolute Error**
Adaboost, Gboost, XGBoost, LASSO, ENet, Kernel Ridge Regression, Extra Trees	0.0715
Adaboost, GBoost, XGBoost, LASSO, Enet, Kernel Ridge Regression	0.0720
Gboost, LASSO, ENet, Kernel Ridge Regression	0.0690
GBoost, ENet, Kernel Ridge Regression	0.0696

Whilst we see that the combination {GBoost, LASSO, Enet, KRR} has a lower mean absolute cross validation score, we chose to use the one standard deviation technique, picking the {GBoost, ENet, KRR} ensemble which has a nearly identical mean absolute error cross validation score yet is a simpler model as one of the models to consider weighting in the final stacked model.

Then, we attempted to find the optimal meta model to use for our stacked model technique by fitting all possible combinations of meta models onto our established base model. From this, we found that the LASSO was the best meta model with a mean absolute error of 12483.72.

We finally had to determine the weighting to assign to each of these averaged models, by predicting these on the validation set to obtain the following MAEs:

Table 3.5 – MAE of averaged models

Averaged Model	MAE
Gradient Boost, LASSO, Elastic Net, Kernel Ridge Regression	12320.82
Gradient Boost, Elastic Net, Kernel Ridge Regression	12258.01

Using a trial and error process, we tried different weightings for each of the averaged models, estimating an MAE on the validation set through submission to Kaggle. This led us to weight the averaged model {GBoost, LASSO, ENet, KRR} 80% and the averaged model {GBoost, ENet, KRR} 20%.

The validation scores for some of the other models we tried, obtained from Kaggle submissions, are given below:

Table 3.6 – validation score of selected models

Model	Validation score (MAE as evaluated on validation set) correct to 3 dp
Gradient boosting tree	12661.356
XGBoost	13172.835
Ridge regression	175559.037
Final Model 1	10945.497
Final Model 2	10972.019

The substantially lower MAEs for the final models, as evaluated through Kaggle, justifies them as the ones we have chosen to focus our analysis on, and to recommend to the government. The final models, both of which were combinations of individual models, outperformed individual models. It is particularly worth noting the extreme difference between the validation set MAE for the ridge regression and those of the other models. This highlights the predictive disadvantage of the parametric model which assumed a fixed, linear relationship between the response and the predictors (as we did not include any nonlinear effects in the predictor variables), suggesting that there may be nonlinearity present. The nonparametric models, which in contrast, assumed no fixed functional form of the relationship between predictors and response in the population data, were able to account for these possible nonlinear relationships and achieve predictions through capturing those particular variations in the predictor variables.

Part 5: Further directions and conclusions

Overall, we recommend to the government the use of model stacking and ensembling when aiming to obtain the most accurate predictions of house prices. The ensembling technique should ideally incorporate simple regularisation methods such the lasso and elastic net, as well as modern statistical modelling methods such as gradient boost and nonparametric considerations such as the kernel ridge regression.

However, we note the limitations of the approach we took, and recommend possible further directions to provide further guidance for government policy decisions.

Our final strong predictive performance was achieved by harnessing each model's strengths and limiting their weaknesses through the creation of ensembles. This guards against allowing any single model to predict badly on particular observations or parts of the dataset, as each method may have particular disadvantages precluding it from predicting well on certain observed values.

However, while we attempted to construct final models based on a combination of various base techniques to combine and compromise between the different advantages of each one, we note that it is worthwhile to attempt to construct models on datasets with more information, with less missing values for certain predictors.

With respect to feature engineering, it is possible that we could also attempt to model more nonlinear relationships in the dataset, including more nonlinear terms or transformations. Our final training dataset included only one variable transformation (log transformation of the price variable), with the strength of our prediction performance achieved by the sophistication of our base models and the synergy of using these in combination with each other rather than through rigorous cleaning of the data. As shown by the final validation set MAEs obtained from Kaggle, this step in feature engineering would be particularly important for analysis which emphasises parametric forms for interpretability rather than the extensive use of nonparametric techniques as we have done so here.

The validation set approach, while feasible here given the large size of the validation set, could nonetheless be replaced by different model selection methods. These include the cross-validation method (which gives us more data about the predictive power of our models, as prediction is performed on more observations).

We recommend that other models may be relevant and appropriate to the type of analysis we are conducting here. Our exploratory data analysis identified a small subset of important predictors shown to be highly relevant to the response, which could imply that a small number of predictors account for a large proportion of the variation in the response. Because of this, it may be relevant to perform to dimension reduction methods such as principal components regression and partial least squares, both of which prioritise first identifying the predictors or combinations of predictors which are the most relevant to the response.

Finally, while we have used a highly data-driven approach to recommend a method for predicting property prices, the government should note that this is far from the only technique that can be implemented for this task. All the models discussed in this paper have used sophisticated, rigorous statistical and mathematical techniques to identify and analyse the relationships between real-life phenomena, however, there is a wealth of econometric analysis available which relies on public policy experience, analysis of human behaviour and industry knowledge to achieve highly accurate predictions about the property market. Further analysis could incorporate this domain knowledge to a further extent to improve upon the findings made here.