**Executive Summary**

This paper explores the relationship between various property attributes and the price of said piece of property by applying the random forest algorithm. The randomForest package in R was applied to the [USA Housing Dataset](https://www.kaggle.com/gpandi007/usa-housing-dataset) sourced from Kaggle to determine the most impactful features on the sale price of a residential property. Three implementations were tested: a basic out-of-the-box implementation of random forest, a random forest using manual validation, and a random forest with a specified validation dataset. Across all three forests, OverallQual (overall quality) was the most important predictor variable. The other nine of the top ten most important predictors also remained consistent between forests, although their ranking varies slightly. These variables were GrLivArea (above ground living area), neighborhood, GarageCars (size of garage in car capacity), ExterQual (quality of the material on the exterior), TotalBsmtSF (total square feet of basement area), 1stFlrSF (first floor square feet), 2ndFlrSF (second floor square feet), GarageArea (size of garage in square feet), and BsmtFinSF1 (Type 1 finished basement square feet). It is worth noting that many of these variables were highly correlated and interdependent, and further studies may benefit from exploring the impact of either decorrelating these variables or further limiting the dataset to reduce dependencies between features.

**Data Source and Definitions**

This analysis utilized the [USA Housing Dataset](https://www.kaggle.com/gpandi007/usa-housing-dataset) sourced from Kaggle. The dataset consists of 1460 observations of 81 features related to individual residential properties, including lot area, neighborhood, overall quality, year built, etc. A full list of features and their data types can be found in Appendix A. The dataset was narrowed down to 69 features (Appendix B) by removing variables that were not relevant to the analysis. Of the remaining features, the majority were categorical. Those that were continuous generally described square footage for an attribute of the house (ex. Garage square footage). The outcome variable, Sale Price, was also continuous.

**Exploratory Data Analysis**

The first step of Exploratory Data Analysis conducted was replacing null values. Due to the nature of the dataset, it was expected that there would be numerous null values – not every residential property is expected to have each one of the dataset features. Upon initial examination, 13 features contained null values (Appendix C). These were a mix of categorical and continuous variables. Null values in categorical features were replaced with a “None” or “Unknown” value to indicate that the property did not contain that feature, while null values in continuous features were replaced with a 0, indicating that the property contained 0 square feet of that feature.

Once the dataset was sufficiently cleaned, the next step was to examine the distribution of the various features and ensure there were no anomalies. Since the reduced dataset contained 69 features, it would not be prudent to discuss the distribution of every feature. However, of note is the fact that the outcome variable SalePrice was somewhat normally distributed with a right skew due to the presence of outliers on the upper end of the price range (Appendix D). The mean value was $180,921.20, the median was $163,000.00, and the standard deviation was $79,442.50.

**Primary Method – Random Forest**

The method used in the analysis of the dataset was the random forest algorithm. At its core, the random forest algorithm consists of a large collection of uncorrelated decision trees which individually leverage random subsets of the predictor variables and observations to generate predictive models. The random forest algorithm has many benefits which make it a popular predictive tool, including the fact that there is no preprocessing of data required or specific data requirements in order to use the method. The random forest algorithm can be utilized for classification or regression applications. In R, the randomForest package contains the most basic implementation of this algorithm. The package also features built in cross-validation, keeping a random set of the observations out-of-bag (OOB) to validate the forest’s predictions and generate a mean square error (MSE) estimate.

The basic concept behind a random forest is *strength in numbers* or *wisdom of the crowd*. The algorithm is built off the idea that a prediction from a large group of models is likely to be more accurate than from a single model. A random forest consists of a large number of decision trees, each of which performs a classification or regression task (depending upon the desired outcome of the random forest) by recursively asking simple true or false questions which split the data into subgroups. The random forest then makes a decision based on the collective predictions of the decision trees – for regression, the result is an average of all the decision trees’ predictions, while for classification, the result is the most popular of the decision trees’ predictions.

As well as generating predictions for the outcome variable based on the models created by each decision tree in the forest, the random forest algorithm also measures variable importance by tracking a statistic called node purity. In the simplest terms, node purity is the degree of homogeneity in the outcome variable of observations in a given node. When a decision tree splits the data into subgroups using any given predictor variable, the importance of that variable will be based on how much purer the resulting nodes are than their parent node. The node purity increase from splitting using a particular variable is tracked across all trees and averaged to determine the importance of that variable. In the randomForest package for R, node purity is measured in terms of the Gini index for classification and the residual sum of squares for regression.

It is important to note that the random forest algorithm relies on its forest of decision trees being uncorrelated to work. This is achieved using two methods: bagging and feature randomness. Bagging is a combination of bootstrapping and aggregating – each decision tree is built using a random subset of the data with replacement. The effect of this is that each decision tree will have a unique dataset it is trained on, resulting in different trees. Feature randomness entails limiting the number of features a tree can consider, further resulting in diverse trees. For classification, the default number of features per tree is the square root of the number of predictor variables, whereas for regression, the default is the number of predictor variables divided by three.

**Application of Primary Method**

*Data Requirements*

Since the random forest algorithm can generally be applied to classification or regression problems, any dataset that would be appropriate for performing a regression analysis would satisfy the otherwise lax data requirements of the method. Given that the USA Housing Dataset selected for this analysis has only categorical and continuous predictor variables and a continuous response variable suitable for generating a linear regression model, it does satisfy the data requirements for the application of the random forest algorithm.

*Methodology*

Ultimately, three different random forests were created and validated using different randomly-selected subsets of the dataset. The first forest was trained and validated using only the random forest algorithm and its built-in OOB cross-validation feature; as such, the algorithm was given the full dataset of 1460 observations to determine its own train-validate split. The second forest was generated using the training set of an independently-created 70%-30% train-validate split of the dataset. The remaining 30% was then used to perform an independent validation of the resulting forest’s prediction error. Much like the second forest, the third forest was also generated using the training set of a 70%-30% train-validate split of the dataset. However, in this case the remaining 30% was manually specified as a testing dataset for the random forest algorithm, which used the data to simulate a train-validate-test split and generate RMSE values for both the OOB validation and test sets for a side-by-side comparison. No additional data manipulation or tuning of the algorithm was performed for any of the forests; as a result, each forest consisted of 500 decision trees created using randomly-selected subsets of 22 predictor variables from the full set of 68 predictors.

*Results*

Across all three forests, the OverallQual variable (categorical rating of the overall material and finish of the house) was determined to be the most important predictor for the sale price of a given house. In addition, there were only 11 of the 68 predictor variables from the dataset that were represented in the lists of the top 10 most important predictors of each forest. Those 11 variables were OverallQual, Neighborhood, GrLivArea, ExterQual, GarageCars, TotalBsmtSF, X1stFlrSF, GarageArea, X2ndFlrSF, YearBuilt, and BsmtFinSF1. The YearBuilt variable was the 10th most important predictor in the first forest, and it was replaced by the BsmtFinSF1 predictor in the second and third forests.

The root mean square error from the OOB validation of the first forest was $28,188.17, with 88.12% of variance explained. From the OOB validation of the second forest, the RMSE was $31,868.77 with 84.36% of variance explained, compared with $25,529.17 for the test dataset. The third forest had RMSE values of $32,012.26 for the OOB validation with 84.56% of variance explained and $26,246.80 with 90.04% of variance explained for the test dataset. Given the mean of $180,921.20 and the standard deviation of $79,442.50 for the response variable in the full dataset, the mean of $183,505.80 and the standard deviation of $81,437.67 for the response variable in the test dataset, and the relatively high percentages of variance explained for the three forests, these RMSE values indicate a reasonably high degree of predictive accuracy for each forest, which varies little between forests despite the randomness inherent to the method.

**Future Considerations**

One of the main concerns arising from the results of this analysis is that many of the variables identified as the most important predictors are highly correlated with one another; for example, OverallQual is dependent on ExterQual, GrLivArea is dependent on X1stFlrSF and X2ndFlrSF, and GarageCars is dependent on GarageArea. In future analyses of this and other similar datasets, more work could be done in the exploratory data analysis to identify highly correlated and interdependent variables such as these in the dataset in order to either decorrelate the variables or eliminate them from the dataset entirely.

Future analyses using the random forest algorithm may also benefit from tuning the number of trees and the number of predictor variables chosen at random from the full set of predictors for each tree to consider. Such tuning may result in a significant increase in the predictive accuracy of the resulting forests, which may be worth consideration.

**Appendix**

*Appendix A – Full List of Dataset Features*

* MSSubClass: type of house, categorical
* MSZoning: zone of house, categorical
* LotFrontage: Linear feet of street connected to house, continuous
* LotArea: Lot size in square feet, continuous
* Street: Type of road access to house, categorical
* Alley: Type of alley access to house, categorical
* LotShape: General shape of property, categorical
* LandContour: Flatness of the property, categorical
* Utilities: Type of utilities available, categorical
* LotConfig: Type of lot, categorical
* LandSlope: Slope of property, categorical
* Neighborhood: Physical location, categorical
* Condition1: Proximity to various conditions, categorical
* Condition2: Proximity to various conditions (if more than one is present), categorical
* BldgType: Type of home, categorical
* HouseStyle: Style of home, categorical
* OverallQual: Rating of the overall material and finish of the house, categorical
* OverallCond: Rating of the overall condition of the house, categorical
* YearBuilt: Original construction date, continuous
* YearRemodAdd: Remodel date (same as construction date if no remodeling or additions), continuous
* RoofStyle: Type of roof, categorical
* RoofMatl: Roof material, categorical
* Exterior1st: Exterior covering on house, categorical
* Exterior2nd: Exterior covering on house (if more than one material), categorical
* MasVnrType: Masonry veneer type, categorical
* MasVnrArea: Masonry veneer area in square feet, categorical
* ExterQual: Evaluates the quality of the material on the exterior, categorical
* ExterCond: Evaluates the present condition of the material on the exterior, categorical
* Foundation: Type of foundation, categorical
* BsmtQual: Evaluates the height of the basement, categorical
* BsmtCond: Evaluates the general condition of the basement, categorical
* BsmtExposure: Refers to walkout or garden level walls, categorical
* BsmtFinType1: Rating of basement finished area, categorical
* BsmtFinSF1: Type 1 finished square feet, continuous
* BsmtFinType2: Rating of basement finished area (if multiple types), categorical
* BsmtFinSF2: Type 2 finished square feet, continuous
* BsmtUnfSF: Unfinished square feet of basement area, continuous
* TotalBsmtSF: Total square feet of basement area, continuous
* Heating: Type of heating, categorical
* HeatingQC: Heating quality and condition, categorical
* CentralAir: Central air conditioning, categorical (binary)
* Electrical: Electrical system, categorical
* 1stFlrSF: First Floor square feet, continuous
* 2ndFlrSF: Second floor square feet, continuous
* LowQualFinSF: Low quality finished square feet (all floors), continuous
* GrLivArea: Above ground living area square feet, continuous
* BsmtFullBath: Basement full bathrooms, continuous
* BsmtHalfBath: Basement half bathrooms, continuous
* FullBath: Full bathrooms above ground, continuous
* HalfBath: Half baths above ground, continuous
* Bedroom: Bedrooms above ground (does NOT include basement bedrooms), continuous
* Kitchen: Kitchens above grade, continuous
* KitchenQual: Kitchen quality, categorical
* TotRmsAbvGrd: Total rooms above ground (does not include bathrooms), continuous
* Functional: Home functionality, categorical
* Fireplaces: Number of fireplaces, continuous
* FireplaceQu: Fireplace quality, categorical
* GarageType: Garage location, categorical
* GarageYrBlt: Year garage was built, continuous
* GarageFinish: Interior finish of the garage, categorical
* GarageCars: Size of garage in car capacity, continuous
* GarageArea: Size of garage in square feet, continuous
* GarageQual: Garage quality, categorical
* GarageCond: Garage condition, categorical
* PavedDrive: Paved driveway, categorical
* WoodDeckSF: Wood deck area in square feet, continuous
* OpenPorchSF: Open porch area in square feet, continuous
* EnclosedPorch: Enclosed porch area in square feet, continuous
* 3SsnPorch: Three season porch area in square feet, continuous
* ScreenPorch: Screen porch area in square feet, continuous
* PoolArea: Pool area in square feet, continuous
* PoolQC: Pool quality, categorical
* Fence: Fence quality, categorical
* MiscFeature: Miscellaneous feature not covered in other categories, categorical
* MiscVal: Value of miscellaneous feature, continuous
* MoSold: Month Sold (MM), continuous
* YrSold: Year Sold (YYYY), continuous
* SaleType: Type of sale, categorical
* SaleCondition: Condition of sale, categorical
* SalePrice: Price the house was sold for, continuous

*Appendix B: Reduced Set of Dataset Features*

## 'data.frame': 1460 obs. of 69 variables:  
## $ MSSubClass : Factor w/ 15 levels "20","30","40",..: 6 1 6 7 6 5 1 6 5 15 ...  
## $ MSZoning : Factor w/ 5 levels "C (all)","FV",..: 4 4 4 4 4 4 4 4 5 4 ...  
## $ LotFrontage : num 65 80 68 60 84 85 75 0 51 50 ...  
## $ LotArea : int 8450 9600 11250 9550 14260 14115 10084 10382 6120 7420 ...  
## $ Street : Factor w/ 2 levels "Grvl","Pave": 2 2 2 2 2 2 2 2 2 2 ...  
## $ Alley : Factor w/ 3 levels "Grvl","None",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ LotShape : Factor w/ 4 levels "IR1","IR2","IR3",..: 4 4 1 1 1 1 4 1 4 4 ...  
## $ LandContour : Factor w/ 4 levels "Bnk","HLS","Low",..: 4 4 4 4 4 4 4 4 4 4 ...  
## $ Utilities : Factor w/ 2 levels "AllPub","NoSeWa": 1 1 1 1 1 1 1 1 1 1 ...  
## $ LotConfig : Factor w/ 5 levels "Corner","CulDSac",..: 5 3 5 1 3 5 5 1 5 1 ...  
## $ LandSlope : Factor w/ 3 levels "Gtl","Mod","Sev": 1 1 1 1 1 1 1 1 1 1 ...  
## $ Neighborhood : Factor w/ 25 levels "Blmngtn","Blueste",..: 6 25 6 7 14 12 21 17 18 4 ...  
## $ BldgType : Factor w/ 5 levels "1Fam","2fmCon",..: 1 1 1 1 1 1 1 1 1 2 ...  
## $ HouseStyle : Factor w/ 8 levels "1.5Fin","1.5Unf",..: 6 3 6 6 6 1 3 6 1 2 ...  
## $ OverallQual : Factor w/ 10 levels "1","2","3","4",..: 7 6 7 7 8 5 8 7 7 5 ...  
## $ OverallCond : Factor w/ 9 levels "1","2","3","4",..: 5 8 5 5 5 5 5 6 5 6 ...  
## $ YearBuilt : int 2003 1976 2001 1915 2000 1993 2004 1973 1931 1939 ...  
## $ YearRemodAdd : int 2003 1976 2002 1970 2000 1995 2005 1973 1950 1950 ...  
## $ RoofStyle : Factor w/ 6 levels "Flat","Gable",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ RoofMatl : Factor w/ 8 levels "ClyTile","CompShg",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Exterior1st : Factor w/ 15 levels "AsbShng","AsphShn",..: 13 9 13 14 13 13 13 7 4 9 ...  
## $ ExterQual : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 4 3 4 3 4 3 4 4 4 ...  
## $ ExterCond : Factor w/ 5 levels "Ex","Fa","Gd",..: 5 5 5 5 5 5 5 5 5 5 ...  
## $ Foundation : Factor w/ 6 levels "BrkTil","CBlock",..: 3 2 3 1 3 6 3 2 1 1 ...  
## $ BsmtQual : Factor w/ 5 levels "Ex","Fa","Gd",..: 3 3 3 5 3 3 1 3 5 5 ...  
## $ BsmtCond : Factor w/ 5 levels "Fa","Gd","None",..: 5 5 5 2 5 5 5 5 5 5 ...  
## $ BsmtFinType1 : Factor w/ 7 levels "ALQ","BLQ","GLQ",..: 3 1 3 1 3 3 3 1 7 3 ...  
## $ BsmtFinSF1 : int 706 978 486 216 655 732 1369 859 0 851 ...  
## $ BsmtUnfSF : int 150 284 434 540 490 64 317 216 952 140 ...  
## $ TotalBsmtSF : int 856 1262 920 756 1145 796 1686 1107 952 991 ...  
## $ Heating : Factor w/ 6 levels "Floor","GasA",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ HeatingQC : Factor w/ 5 levels "Ex","Fa","Gd",..: 1 1 1 3 1 1 1 1 3 1 ...  
## $ CentralAir : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...  
## $ Electrical : Factor w/ 6 levels "FuseA","FuseF",..: 5 5 5 5 5 5 5 5 2 5 ...  
## $ X1stFlrSF : int 856 1262 920 961 1145 796 1694 1107 1022 1077 ...  
## $ X2ndFlrSF : int 854 0 866 756 1053 566 0 983 752 0 ...  
## $ LowQualFinSF : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ GrLivArea : int 1710 1262 1786 1717 2198 1362 1694 2090 1774 1077 ...  
## $ BsmtFullBath : int 1 0 1 1 1 1 1 1 0 1 ...  
## $ BsmtHalfBath : int 0 1 0 0 0 0 0 0 0 0 ...  
## $ FullBath : int 2 2 2 1 2 1 2 2 2 1 ...  
## $ HalfBath : int 1 0 1 0 1 1 0 1 0 0 ...  
## $ BedroomAbvGr : int 3 3 3 3 4 1 3 3 2 2 ...  
## $ KitchenAbvGr : int 1 1 1 1 1 1 1 1 2 2 ...  
## $ KitchenQual : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 4 3 3 3 4 3 4 4 4 ...  
## $ TotRmsAbvGrd : int 8 6 6 7 9 5 7 7 8 5 ...  
## $ Functional : Factor w/ 7 levels "Maj1","Maj2",..: 7 7 7 7 7 7 7 7 3 7 ...  
## $ Fireplaces : int 0 1 1 1 1 0 1 2 2 2 ...  
## $ FireplaceQu : Factor w/ 6 levels "Ex","Fa","Gd",..: 4 6 6 3 6 4 3 6 6 6 ...  
## $ GarageType : Factor w/ 7 levels "2Types","Attchd",..: 2 2 2 6 2 2 2 2 6 2 ...  
## $ GarageFinish : Factor w/ 4 levels "Fin","None","RFn",..: 3 3 3 4 3 4 3 3 4 3 ...  
## $ GarageCars : int 2 2 2 3 3 2 2 2 2 1 ...  
## $ GarageArea : int 548 460 608 642 836 480 636 484 468 205 ...  
## $ GarageQual : Factor w/ 6 levels "Ex","Fa","Gd",..: 6 6 6 6 6 6 6 6 2 3 ...  
## $ GarageCond : Factor w/ 6 levels "Ex","Fa","Gd",..: 6 6 6 6 6 6 6 6 6 6 ...  
## $ PavedDrive : Factor w/ 3 levels "N","P","Y": 3 3 3 3 3 3 3 3 3 3 ...  
## $ WoodDeckSF : int 0 298 0 0 192 40 255 235 90 0 ...  
## $ OpenPorchSF : int 61 0 42 35 84 30 57 204 0 4 ...  
## $ EnclosedPorch: int 0 0 0 272 0 0 0 228 205 0 ...  
## $ X3SsnPorch : int 0 0 0 0 0 320 0 0 0 0 ...  
## $ ScreenPorch : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ PoolArea : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ PoolQC : Factor w/ 4 levels "Ex","Fa","Gd",..: 4 4 4 4 4 4 4 4 4 4 ...  
## $ Fence : Factor w/ 5 levels "GdPrv","GdWo",..: 5 5 5 5 5 3 5 5 5 5 ...  
## $ MiscFeature : Factor w/ 5 levels "Gar2","None",..: 2 2 2 2 2 4 2 4 2 2 ...  
## $ MiscVal : int 0 0 0 0 0 700 0 350 0 0 ...  
## $ MoSold : int 2 5 9 2 12 10 8 11 4 1 ...  
## $ YrSold : int 2008 2007 2008 2006 2008 2009 2007 2009 2008 2008 ...  
## $ SalePrice : int 208500 181500 223500 140000 250000 143000 307000 200000 129900 118000 ...

*Appendix C: Null Values per Feature*

## MSSubClass MSZoning LotFrontage LotArea Street   
## 0 0 259 0 0   
## Alley LotShape LandContour Utilities LotConfig   
## 1369 0 0 0 0   
## LandSlope Neighborhood BldgType HouseStyle OverallQual   
## 0 0 0 0 0   
## OverallCond YearBuilt YearRemodAdd RoofStyle RoofMatl   
## 0 0 0 0 0   
## Exterior1st ExterQual ExterCond Foundation BsmtQual   
## 0 0 0 0 37   
## BsmtCond BsmtFinType1 BsmtFinSF1 BsmtUnfSF TotalBsmtSF   
## 37 37 0 0 0   
## Heating HeatingQC CentralAir Electrical X1stFlrSF   
## 0 0 0 1 0   
## X2ndFlrSF LowQualFinSF GrLivArea BsmtFullBath BsmtHalfBath   
## 0 0 0 0 0   
## FullBath HalfBath BedroomAbvGr KitchenAbvGr KitchenQual   
## 0 0 0 0 0   
## TotRmsAbvGrd Functional Fireplaces FireplaceQu GarageType   
## 0 0 0 690 81   
## GarageFinish GarageCars GarageArea GarageQual GarageCond   
## 81 0 0 81 81   
## PavedDrive WoodDeckSF OpenPorchSF EnclosedPorch X3SsnPorch   
## 0 0 0 0 0   
## ScreenPorch PoolArea PoolQC Fence MiscFeature   
## 0 0 1453 1179 1406   
## MiscVal MoSold YrSold SalePrice   
## 0 0 0 0

*Appendix D: Distribution of SalePrice Variable*

