Prediction of Home Sales Prices Using Machine Learning

#Project Overview

This project focuses on building predictive models to estimate house sale prices in Ames, Iowa, employing machine learning techniques. The influential factors affecting house prices are categorized into two main aspects: dwelling-related factors and neighborhood-related factors.

Dwelling-Related Factors

Factors such as dwelling size, quality, and condition significantly impact house prices. For instance, attributes like 1stFlrSF (First Floor square feet), OverallQual (overall material and finish quality), and 2ndFlrSF (Second Floor square feet) hold considerable importance in predicting house sale prices.

Neighborhood-Related Factors

Neighborhood attributes, including school quality, nearby amenities, and overall area appeal, also play a crucial role in determining house prices. These factors collectively contribute to the overall market value of a property.

Dataset Details

The project utilizes the Ames Housing dataset, encompassing 1,460 homes sold between 2006 and 2010 in Ames, Iowa, with a comprehensive set of 81 features. These features cover various aspects of the properties, including numeric attributes detailing sizes and categorical attributes describing house features, quality, and environmental conditions.

Project Objectives

The main objectives of this project include:

Model Development: Creating predictive models using machine learning techniques like linear regression and random forest to estimate house sale prices.

Identifying Influential Factors: Determining the factors that significantly influence house prices, emphasizing both dwelling-specific and neighborhood-related attributes.

Exploratory Analysis: Analyzing relationships between average sale prices and specific attributes like Neighborhoods, Garage Quality, Paved Drive, Functional, and Sale Condition.

Temporal Trends: Studying changes in house prices over time to observe any temporal patterns or fluctuations.

Neighborhood Insights: Identifying the top 10 neighborhoods in Ames, Iowa, with the highest predicted median house prices.

This repository hosts code, data, and insights from the predictive modeling project, showcasing machine learning techniques applied to real estate data to predict house prices effectively. Feel free to explore the code, analysis, and findings derived from this project.

Data Preprocessing

Load libraries

```
#import libraries
library(knitr)
library(ggplot2)
library(corrplot)
#for vif
library(car)
library(plyr)
library(dplyr)
library(cowplot)
library(jtools)
library(MASS)
library(Metrics)
library(randomForest)
```

Load data

```
data <- read.csv("train.csv")

test <- read.csv("test.csv")
#drop Id variable

data$Id <- NULL
```

Data preprocessing is often described as the set of methods that are used in the process of enhancing the quality of the raw data and includes steps such as outlier removal and imputation of missing values (Cheng, et al., 2021). (Cheng, et al., 2021) further argues that data preprocessing methods are categorized into broad groups depending on their functionality in the treatment of missing observations, detecting outlier values, reducing the dimensions of the data, data normalization and transformation, as well as data sampling (partitioning).

In practice, data preprocessing is essential in that, it is needed to facilitate the validity and reliability of data analysis findings. For instance, sales data such as that of properties will generally have some missing observations due to differences in the characteristics of the properties like whether or not a house has a garage. While some properties might have these kinds of facilities, others might not have and would lead to missing entries in the data. Problems such as missing values and outliers will generally reduce the statistical power of the findings and introduce bias respectively making it necessary to implement appropriate data preprocessing techniques.

Since one of the problems involves developing a linear regression model, imputing of the missing observations in variables that have less than 30% of the missing observations and excluding the variables with a large proportion of missing observations is important. This way, it will be possible to introduce bias when imputing the variables and retaining variables that provide enough information regarding the homes.

Missing Observations Given the large number of variables included in the data, Table 1 below only provides an overview of the variables with at least one missing observation in the data including the percentage of missing observations in each attribute.

```
miss <- data.frame(colSums(is.na(data)))
miss <- cbind(Variable = rownames(miss), miss)
rownames(miss) <- 1:nrow(miss)
names(miss) <- c("Variable","Number of missing observations")
miss <- miss[miss$`Number of missing observations`>0,]
missin <- data.frame((colMeans(is.na(data)))*100)
names(missin) <- c("Missing")
missin <- missin[missin$Missing > 0,]
miss$`Proportion (%)` <- missin
kable(miss)
```

	Variable	Number of missing observations	Proportion (%)
3	LotFrontage	259	17.7397260
6	Alley	1369	93.7671233
25	MasVnrType	8	0.5479452
26	MasVnrArea	8	0.5479452
30	BsmtQual	37	2.5342466
31	BsmtCond	37	2.5342466
32	BsmtExposure	38	2.6027397
33	BsmtFinType1	37	2.5342466
35	BsmtFinType2	38	2.6027397
42	Electrical	1	0.0684932
57	FireplaceQu	690	47.2602740
58	GarageType	81	5.5479452
59	GarageYrBlt	81	5.5479452
60	GarageFinish	81	5.5479452
63	GarageQual	81	5.5479452
64	GarageCond	81	5.5479452
72	PoolQC	1453	99.5205479
73	Fence	1179	80.7534247
74	MiscFeature	1406	96.3013699

Ouput after treating the missing observations in the data.

```
#Drop variables with many missing observations

house_data1 <- data[colSums(is.na(data))/nrow(data) < .3]

#Count missing

# replacing NA with mean value if the variable is numeric and mode for factor variables

Mode <- function(x) {
    ux <- unique(x)
    ux[which.max(tabulate(match(x, ux)))]
}


house_data2 <- data.frame(lapply(house_data1, function(x) {
    if(is.character(x)) replace(x, is.na(x), Mode(na.omit(x)))
    else if(is.numeric(x)) replace(x, is.na(x), median(x, na.rm=TRUE))
```

```
    else x
}))


miss <- data.frame(colSums(is.na(house_data2)))
miss <- cbind(Variable = rownames(miss), miss)
rownames(miss) <- 1:nrow(miss)
names(miss) <- c("Variable","Number of missing observations")
miss <- miss[miss$`Number of missing observations`>0,]
missin <- data.frame((colMeans(is.na(house_data2)))*100)
names(missin) <- c("Missing")
missin <- missin[missin$Missing > 0,]
miss$`Proportion (%)` <- missin
kable(miss)
```

Variable	Number of missing observations	Proportion (%)

The table above shows that all the missing observations in the data have been treated.

Exploratory Data Analysis

Exploratory data analysis was conducted to determine:

i. Whether there were any outliers in the target variable (Sales Price)

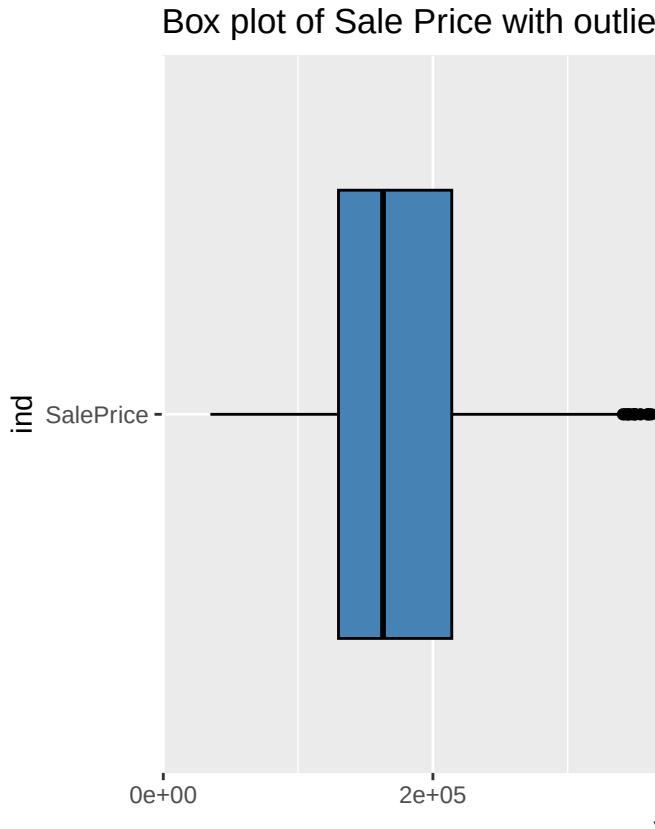
ii. Which variables that exhibit multicollinearity hence are suitable for the linear regression model and which variables do not have a quantifiable relationship with Sales price i.e., which variables are significant?

iii. Examine the distributional relationship between the average sales price and, Neighborhoods, Garage Quality, Paved Drive, Functional, and Sale Condition.

iv. Whether the price of houses in Ames, Iowa change with time?

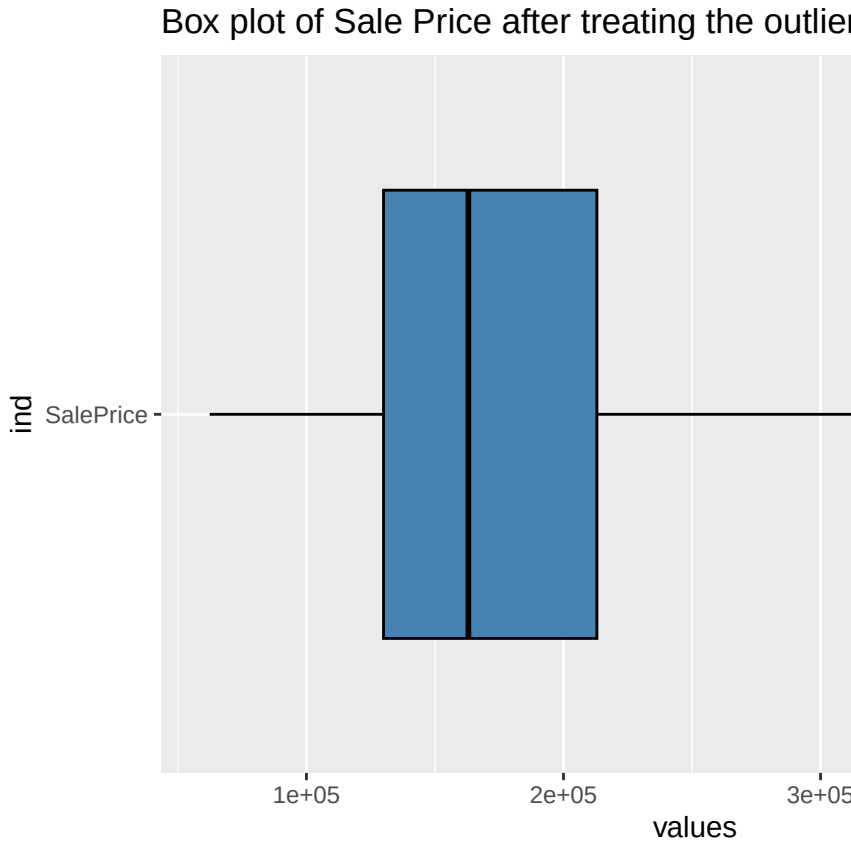
v. Which sale type and condition results are associated with higher sale prices?

```
dfx <- house_data2[75]

ggplot(stack(dfx), aes(x = ind, y = values)) +
  geom_boxplot(fill='steelblue', color="black") +
  coord_flip() + ggtitle("Box plot of Sale Price with outliers")
```



Are there any outliers in the target variable (Sales Price)?

```
x<-quantile(house_data2$SalePrice,c(0.01,0.99))
data <- house_data2[house_data2$SalePrice >=x[1] & house_data2$SalePrice<=x[2],]
dfx <- data[75]
ggplot(stack(dfx), aes(x = ind, y = values)) +
  geom_boxplot(fill='steelblue', color="black") +
  coord_flip() + ggtitle("Box plot of Sale Price after treating the outliers")
```



Removing outliers from the target variable

Which variables that exhibit multicollinearity hence are suitable for the linear regression model and which variables do not have a quantifiable relationship with Sales price i.e., which variables are significant? The following table shows the variables with a correlation score of greater than 0.3 with the target attribute indicating that they are the variables that have the highest association with the sales price of a home.

```
data <- house_data2
nums <- unlist(lapply(data, is.numeric))

numdf <- data[ , nums]
corl <- data.frame(cor(numdf[-37], numdf$SalePrice))
corl <- cbind(Variable = rownames(corl), corl)
rownames(corl) <- 1:nrow(corl)
names(corl) <- c("Variable","Correlation")
corl <- corl[abs(corl$Correlation) > 0.3,]

kable(corl)
```

	Variable	Correlation
2	LotFrontage	0.3347709
4	OverallQual	0.7909816
6	YearBuilt	0.5228973
7	YearRemodAdd	0.5071010
8	MasVnrArea	0.4726145

	Variable	Correlation
9	BsmtFinSF1	0.3864198
12	TotalBsmtSF	0.6135806
13	X1stFlrSF	0.6058522
14	X2ndFlrSF	0.3193338
16	GrLivArea	0.7086245
19	FullBath	0.5606638
23	TotRmsAbvGrd	0.5337232
24	Fireplaces	0.4669288
25	GarageYrBlt	0.4667537
26	GarageCars	0.6404092
27	GarageArea	0.6234314
28	WoodDeckSF	0.3244134
29	OpenPorchSF	0.3158562

There are up to 18 attributes with significantly high association with the sale price of a home as shown in the preceeding table. However, it is possible that some attributes have high correlation i.e., multicollinearity which happens when predictor variables in the regression model are highly correlated to each other. This will generally make it hard to interpret the model outcome besides also leading to the overfitting problem. To deal with multicollinearity, the variance inflation factor of each attribute will be computed and the variables with the highest VIF will be excluded from the final dataset.

The table below shows the VIF scores of the predictor variables.

```
dfc <- data[corl$Variable]

dfc$SalePrice <- data$SalePrice

init_model <- lm(SalePrice ~., data = dfc)
scrs <- data.frame(vif(init_model))
scrs <- cbind(Variable = rownames(scrs), scrs)
rownames(scrs) <- 1:nrow(scrs)
names(scrs) <- c("Variable","VIF")
kable(scrs, caption = "Variance Inflation Factor per attribute")
```

Check for multicollinerity in the selected variables

Table 4: Variance Inflation Factor per attribute

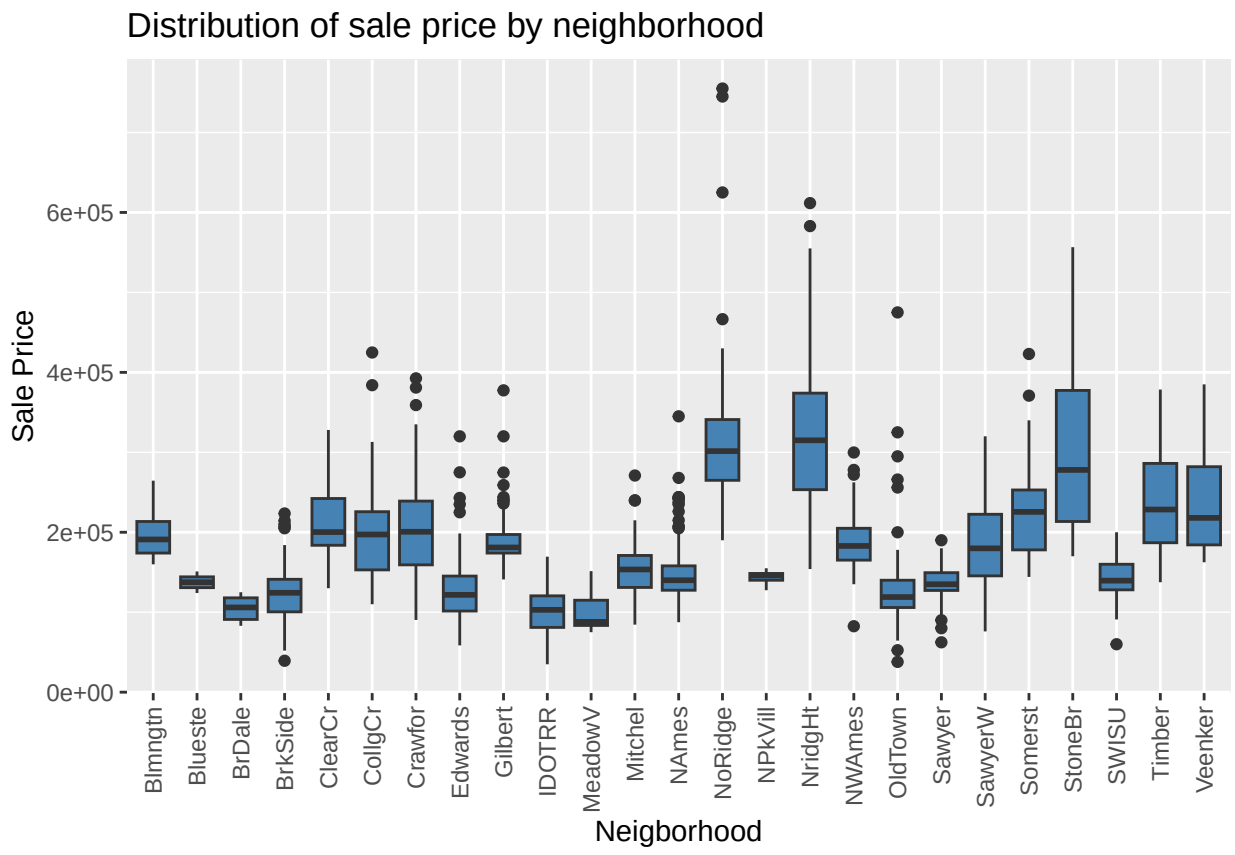
Variable	VIF
LotFrontage	1.311882
OverallQual	2.890824
YearBuilt	3.644942
YearRemodAdd	1.931439
MasVnrArea	1.356099
BsmtFinSF1	1.539294
TotalBsmtSF	3.923885
X1stFlrSF	73.625735
X2ndFlrSF	90.839600

Variable	VIF
GrLivArea	129.745506
FullBath	2.339937
TotRmsAbvGrd	3.474327
Fireplaces	1.471686
GarageYrBlt	3.179289
GarageCars	5.467861
GarageArea	5.345455
WoodDeckSF	1.146747
OpenPorchSF	1.189558

Examine the distributional relationship between the average sales price and, Neighborhoods, Garage Quality, Paved Drive, Functional, and Sale Condition.

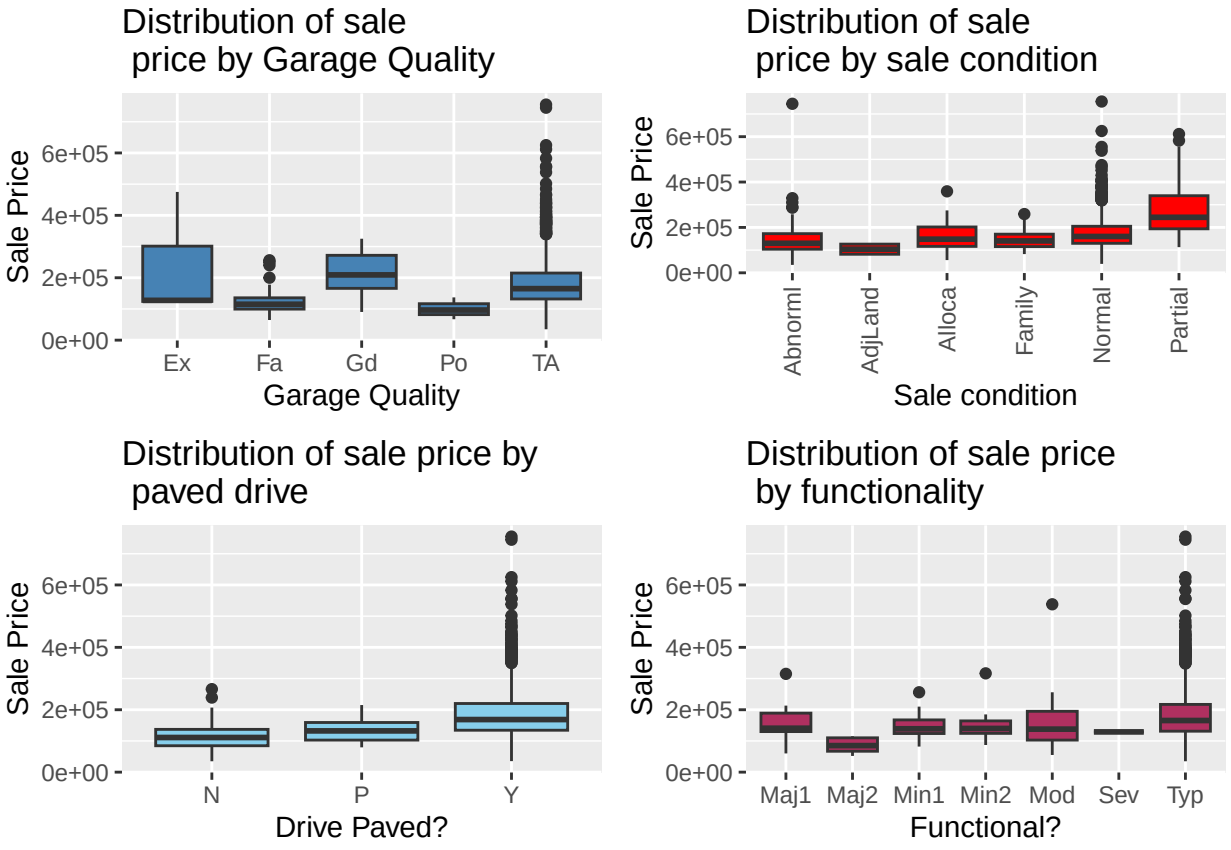
Distribution of Sale price by neighborhood From the figure below, it is noted that the NridgHt, NoRidge, and StoneBr neighborhoods had the highest median sale price while Meadow,V, BrDale, and IDOTRR neighborhoods had the lowest median sale price.

```
g1 <- data %>% ggplot(aes(x =Neighborhood, y = SalePrice)) +
                geom_boxplot(fill = 'steelblue')+
        labs(x="Neigborhood", y="Sale Price")+
        ggtitle("Distribution of sale price by neighborhood")+ theme(axis.text.x = element_text(angle =

g1
```

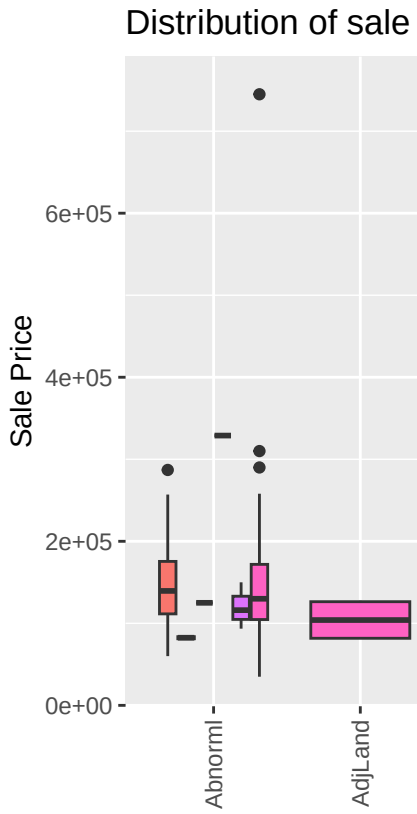


Similarly, as observed from the plot below, homes with good garage quality, Partial condition (Home was not completed when last assessed (associated with New Homes)), paved drives, and Typical Functionality had the highest median sale price.

```
g2 <- data %>% ggplot(aes(x =GarageQual , y = SalePrice)) +
                geom_boxplot(fill = 'steelblue')+
        labs(x="Garage Quality", y="Sale Price")+
        ggtitle("Distribution of sale\n price by Garage Quality")

g3 <- data %>% ggplot(aes(x =SaleCondition , y = SalePrice)) +
                geom_boxplot(fill = 'red')+
        labs(x="Sale condition", y="Sale Price")+
        ggtitle("Distribution of sale\n price by sale condition")+ theme(axis.text.x = element_text(angl

g4 <- data %>% ggplot(aes(x =PavedDrive , y = SalePrice)) +
                geom_boxplot(fill = 'skyblue')+
        labs(x="Drive Paved?", y="Sale Price")+
        ggtitle("Distribution of sale price by\n paved drive")

g5 <- data %>% ggplot(aes(x = Functional , y = SalePrice)) +
                geom_boxplot(fill = 'maroon')+
        labs(x="Functional?", y="Sale Price")+
        ggtitle("Distribution of sale price\n by functionality")

plot_grid(g2,g3, g4, g5)
```



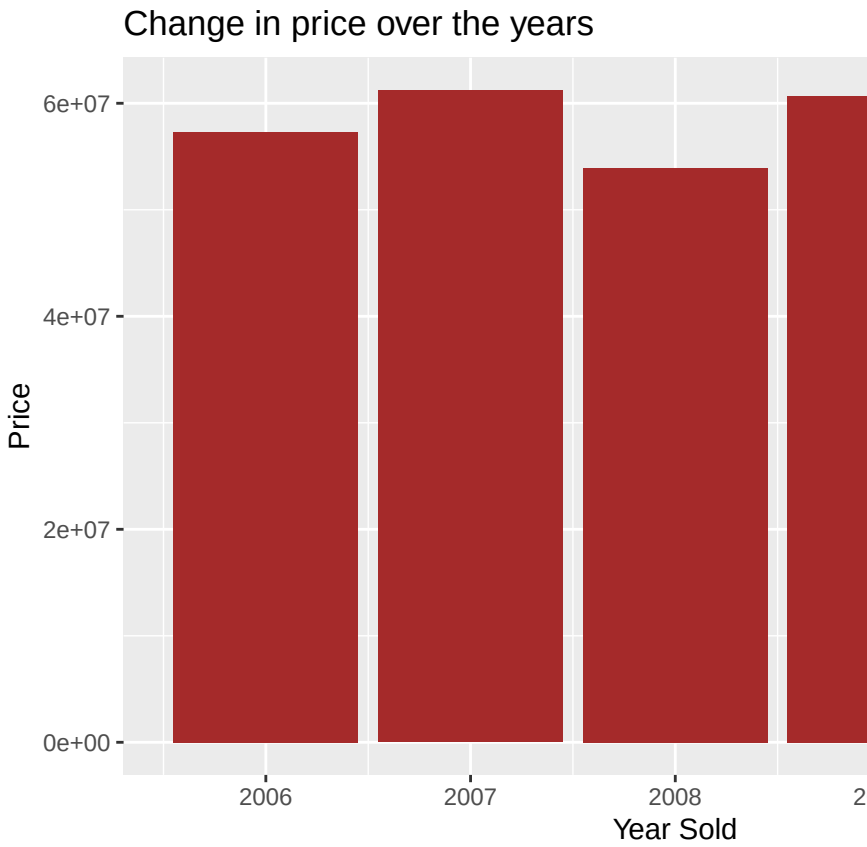
```
g3 <- data %>% ggplot(aes(x =SaleCondition , y = SalePrice, fill= SaleType)) +
                geom_boxplot()+
        labs(x="Sale condition", y="Sale Price")+
        ggtitle("Distribution of sale price by sale condition and sale type")+ theme(axis.text.x = eleme

g3
```



Which sale type and condition results are associated with higher sale prices?

Homes with Normal sale condition and sale with Contract 15% Down payment regular terms have the highest median sale price.

```
ggplot(data, aes(x= YrSold, y= SalePrice)) +
  geom_bar(stat = "identity", fill = "brown")   +
  scale_color_gradient(name = "|y - ybar|")+
  xlab("Year Sold") +
  ylab("Price") +
  ggtitle("Change in price over the years")
```



Does the price of homes change with time

While 2010 has the lowest prices, there is no apparent trend in the prices of houses. As such, the prices of houses cannot be assumed to increase over time

Further Processing

Using a threshold of 10 which is proposed by studies such as O'Brien (2007) and Vittinghoff, et al. (2012), it is observed that the variables that exhibit the highest VIF are X1stFlrSF, X2ndFlrSF, and GrLivArea. As shown in the table below, removing the GrLivArea attribute reduced the VIF of both X1stFlrSF and X2ndFlrSF to below the threshold.

The table below shows the VIF of each variable after removing highly collinear attributes

```
threshold=10

# Sequentially exclude attributes with largest VIF until
# all variables have VIF less than threshold
drop=TRUE

handling_vif=data.frame()
while(drop==TRUE) {
  vinit_model=vif(init_model)
  handling_vif=rbind.fill(handling_vif,as.data.frame(t(vinit_model)))
  if(max(vinit_model)>threshold) { init_model=
  update(init_model,as.formula(paste(".","~",".","-",names(which.max(vinit_model))))) }
  else { drop=FALSE }}
```

```
# Show the Model after removing highly correlated Variables
#print(init_model)

# How variables removed sequentially
t_handling_vif = as.data.frame(t(handling_vif))

# Final (uncorrelated) variables with their VIFs
vinit_model_d= as.data.frame(vinit_model)
scrs <- data.frame(vif(init_model))
scrs <- cbind(Variable = rownames(scrs), scrs)
rownames(scrs) <- 1:nrow(scrs)
names(scrs) <- c("Variable","VIF")
kable(scrs, caption = "Variance Inflation Factor per attribute after cleaning")
```

Table 5: Variance Inflation Factor per attribute after cleaning

Variable	VIF
LotFrontage	1.309348
OverallQual	2.881655
YearBuilt	3.474858
YearRemodAdd	1.930446
MasVnrArea	1.354100
BsmtFinSF1	1.539283
TotalBsmtSF	3.922018
X1stFlrSF	5.855265
X2ndFlrSF	3.932646
FullBath	2.339533
TotRmsAbvGrd	3.411467
Fireplaces	1.471683
GarageYrBlt	3.092239
GarageCars	5.459145
GarageArea	5.345452
WoodDeckSF	1.146746
OpenPorchSF	1.189375

As noted earlier, the excluded variables had little association with sale price (see the correlation table above). Therefore, the final dataset contains 17 predictor attributes and 1 target variable (Sale Price). The following table shows the correlation scores of the excluded attributes.

```
corl1 <- data.frame(cor(numdf[-37], numdf$SalePrice))
corl1 <- cbind(Variable = rownames(corl1), corl1)
rownames(corl1) <- 1:nrow(corl1)
names(corl1) <- c("Variable","Correlation")
corl1 <- corl1[abs(corl1$Correlation) < 0.3,]

kable(corl1, caption = "Variables excluded due to low association with Sale Price")
```

Table 6: Variables excluded due to low association with Sale Price

	Variable	Correlation
1	MSSubClass	-0.0842841
3	LotArea	0.2638434
5	OverallCond	-0.0778559
10	BsmtFinSF2	-0.0113781
11	BsmtUnfSF	0.2144791
15	LowQualFinSF	-0.0256061
17	BsmtFullBath	0.2271222
18	BsmtHalfBath	-0.0168442
20	HalfBath	0.2841077
21	BedroomAbvGr	0.1682132
22	KitchenAbvGr	-0.1359074
30	EnclosedPorch	-0.1285780
31	X3SsnPorch	0.0445837
32	ScreenPorch	0.1114466
33	PoolArea	0.0924035
34	MiscVal	-0.0211896
35	MoSold	0.0464322
36	YrSold	-0.0289226

Modelling

This section includes the results obtained following an implementation of both a linear regression model using all the predictor variables, stepwise linear regression model that selects the most relevant variables, and a random forest model. Each of the models are then evaluated using the root mean squared error obtained when predicting the sale price in the test data.

Full Model

```
#get the predictors

dfl <- data[scrs$Variable]

dfl$SalePrice <- data$SalePrice
#fit the model
model <- lm(SalePrice ~., data = dfl)


summ(model)
```

```
## Warning in !is.null(rmarkdown::metadata$output) && rmarkdown::metadata$output
## %in% : 'length(x) = 3 > 1' in coercion to 'logical(1)'
```

Observations	1460
Dependent variable	SalePrice
Type	OLS linear regression

Assumption checking

1. The regression model is linear in parameters

F(17,1442)	322.83
R^2	0.79
Adj. R^2	0.79

	Est.	S.E.	t val.	p
(Intercept)	-1113152.81	133122.54	-8.36	0.00
LotFrontage	63.01	49.57	1.27	0.20
OverallQual	18968.58	1171.34	16.19	0.00
YearBuilt	124.27	58.90	2.11	0.04
YearRemodAdd	342.84	64.22	5.34	0.00
MasVnrArea	28.27	6.14	4.60	0.00
BsmtFinSF1	17.39	2.60	6.70	0.00
TotalBsmtSF	11.31	4.31	2.63	0.01
X1stFlrSF	46.28	5.97	7.75	0.00
X2ndFlrSF	38.19	4.34	8.81	0.00
FullBath	-1811.81	2649.50	-0.68	0.49
TotRmsAbvGrd	1687.10	1084.42	1.56	0.12
Fireplaces	7747.20	1795.79	4.31	0.00
GarageYrBlt	60.82	69.93	0.87	0.38
GarageCars	10381.41	2983.61	3.48	0.00
GarageArea	8.06	10.32	0.78	0.43
WoodDeckSF	31.86	8.15	3.91	0.00
OpenPorchSF	9.70	15.71	0.62	0.54

Standard errors: OLS

Observations as obtained after removing multicollinearity, are independent of each other.

2. The mean of residuals is zero

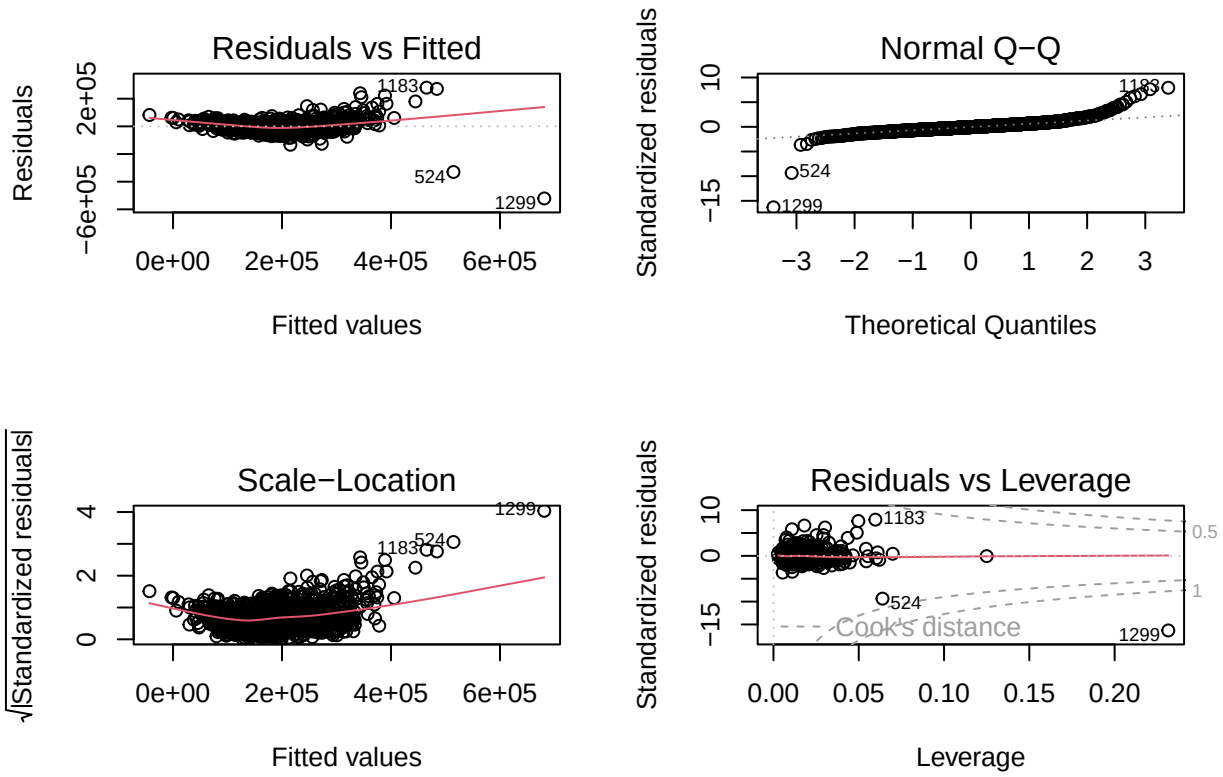
In this test, a check on the mean of the residuals is conducted and if the mean is zero (or very close), then this assumption is held true for that model. With a mean of approximately 0, the model satisfies this assumption.

```
mean(model$residuals)
```

```
## [1] 1.517867e-13
```

3. Homoscedasticity of residuals or equal variance

```
par(mfrow=c(2,2))
plot(model)
```



From the first plot (top-left), there is pattern no pattern indicated by the red line hence it can be argued that the disturbances are homoscedastic hence the assumption is met.

4. No autocorrelation of residuals

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
dwtest(model)
```

```
##
##  Durbin-Watson test
##
## data:  model
## DW = 1.9687, p-value = 0.2745
## alternative hypothesis: true autocorrelation is greater than 0
```

With a high p value of 0.2745, the null hypothesis that true autocorrelation is zero is not rejected. So the assumption that residuals should not be autocorrelated is satisfied by this model.

Stepwise model

```
# Stepwise regression model
stepModel <- stepAIC(model, direction = "both",
                     trace = FALSE)
summ(stepModel)
```

Observations	1460
Dependent variable	SalePrice
Type	OLS linear regression

F(12,1447)	457.07
R^2	0.79
Adj. R^2	0.79

	Est.	S.E.	t val.	p
(Intercept)	-1045760.60	118793.70	-8.80	0.00
OverallQual	18955.63	1166.47	16.25	0.00
YearBuilt	142.37	46.64	3.05	0.00
YearRemodAdd	351.75	61.73	5.70	0.00
MasVnrArea	28.41	6.12	4.64	0.00
BsmtFinSF1	17.93	2.55	7.03	0.00
TotalBsmtSF	12.26	4.27	2.87	0.00
X1stFlrSF	46.92	5.64	8.32	0.00
X2ndFlrSF	38.28	4.07	9.41	0.00
TotRmsAbvGrd	1663.24	1068.77	1.56	0.12
Fireplaces	7493.98	1761.00	4.26	0.00
GarageCars	12527.31	1755.53	7.14	0.00
WoodDeckSF	31.61	8.10	3.90	0.00

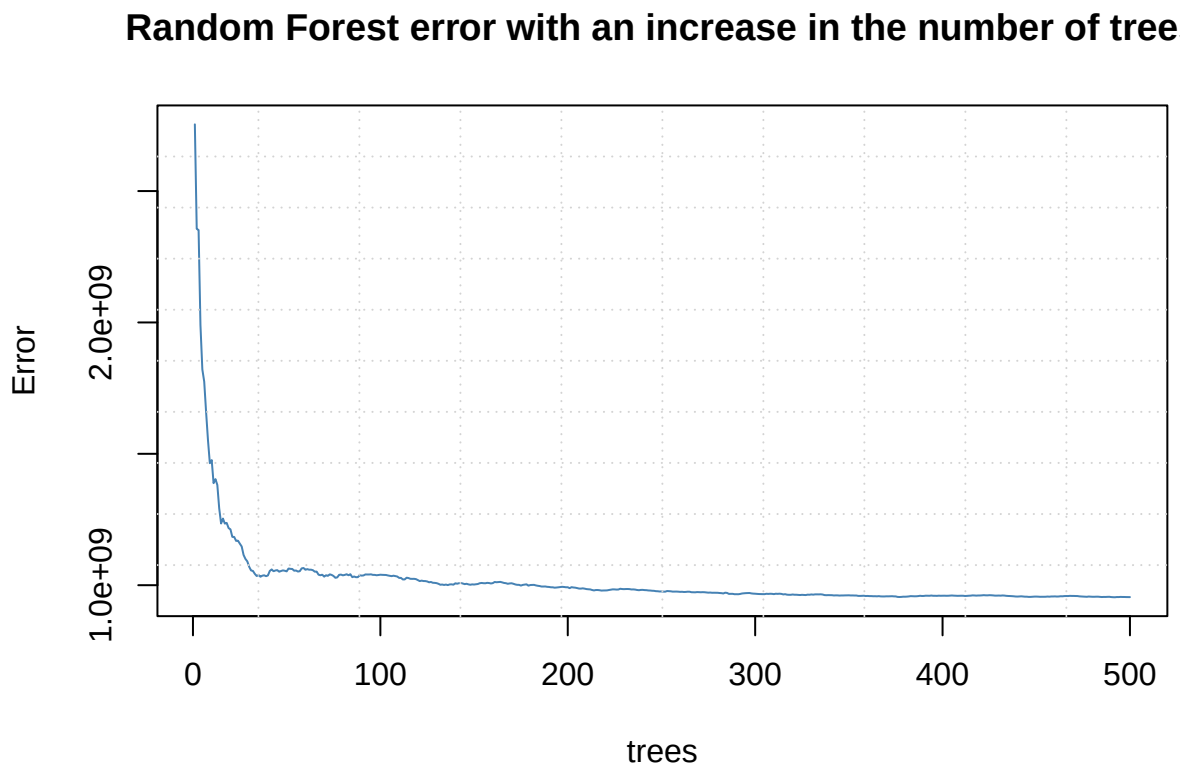
Standard errors: OLS

Random Forest

```
set.seed(42)
rf <- randomForest(SalePrice ~ ., data = dfl, mtry = 3,
                           importance = TRUE, na.action = na.omit)
```

The following plot shows the error rate of the random forest with increase in the number of trees. From the plot, it is noted that the model's performance tends to improve with an increase in the number of trees.

```
plot(rf, main = "Random Forest error with an increase in the number of trees", col = "steelblue")
grid(10)
```



Evaluation

Evaluate the Performance of the models Full model

```
#treat missing in test set

test <- data.frame(lapply(test, function(x) {
    if(is.character(x)) replace(x, is.na(x), Mode(na.omit(x)))
    else if(is.numeric(x)) replace(x, is.na(x), median(x, na.rm=TRUE))
    else x
}))


rmse(predict(model, test), test$SalePrice)
```

```
## [1] 32622.58
```

Stepwise model

```
rmse(predict(stepModel, test), test$SalePrice)
```

```
## [1] 32895.01
```

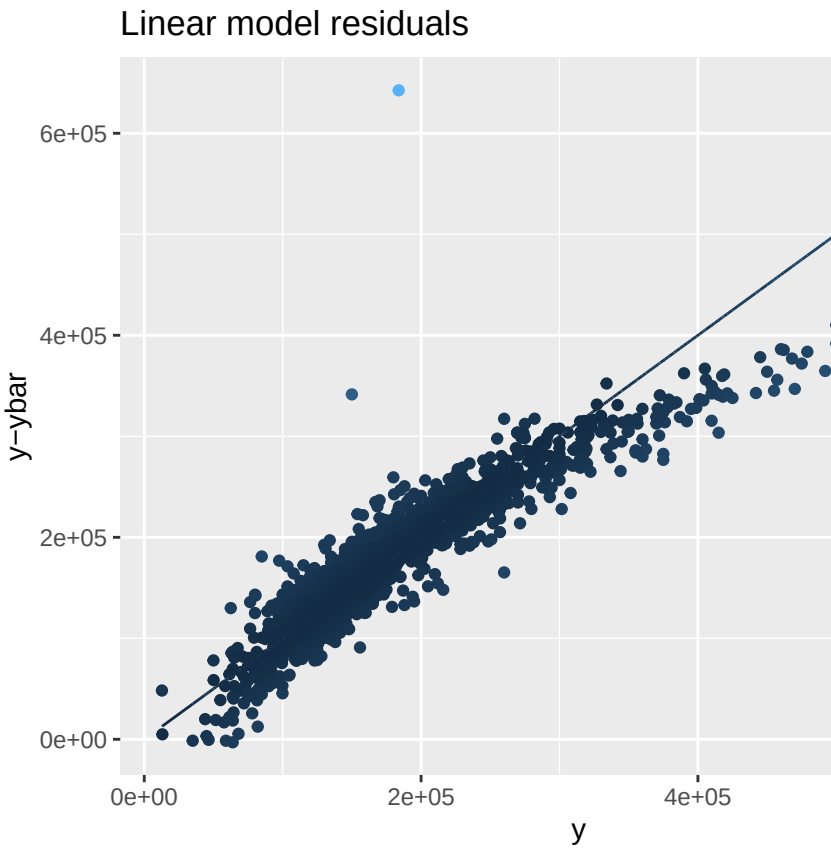
Random Forest

```r
rmse(predict(rf, test), test$SalePrice)
```

```
## [1] 26315.01
```

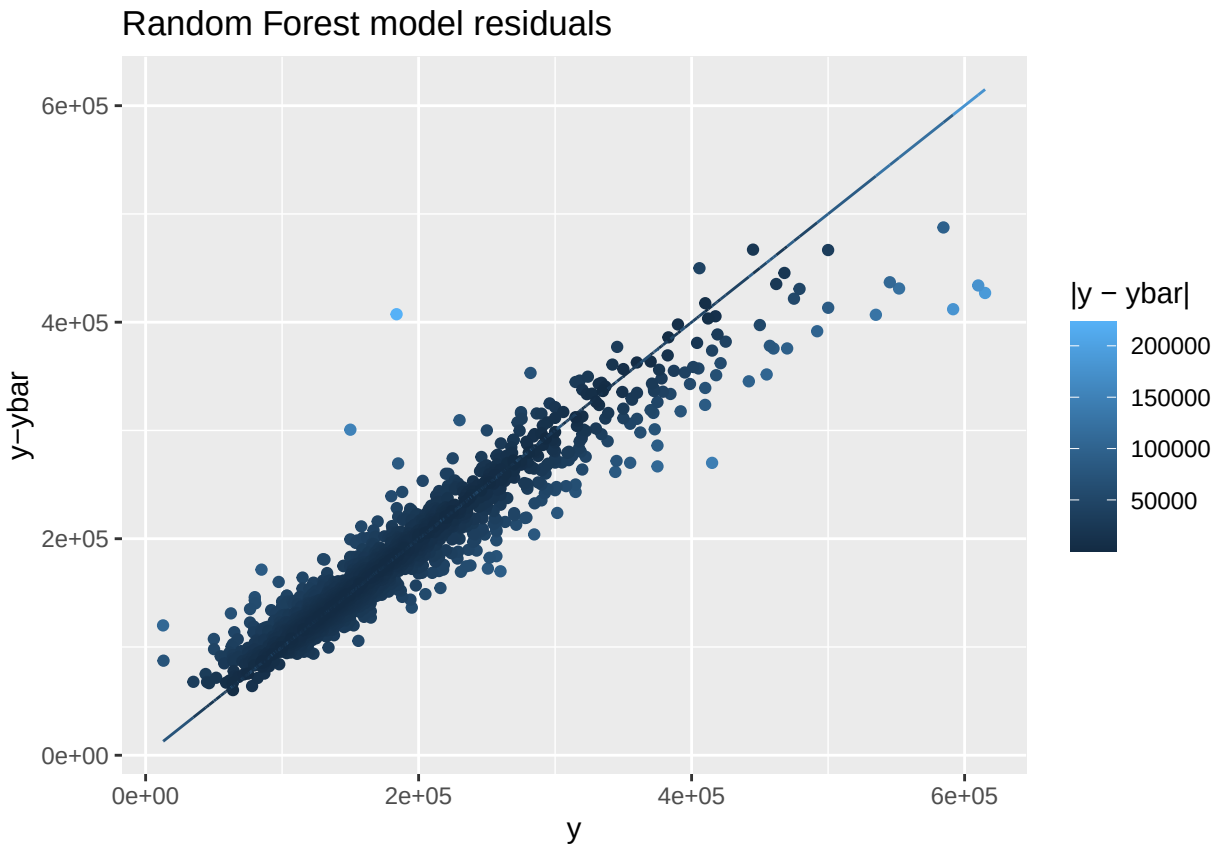
From the performance scores above, it is noted that the random forest model has the lowest RMSE score of $26315.01 compared to the RMSE score of the full regression model of 32622.58 and that of the stepwise regression model which has the highest RMSE indicating the lowest performance of 32895.01.

```r
modelpred <- predict(model, newdata = test)
lmdata <- test %>% mutate(y = SalePrice) %>%
  mutate(ybar =modelpred) %>% mutate(diff = abs(y - ybar))

prediction_data <- lmdata %>% filter(diff > 1.5) %>% arrange(desc(diff))


resid_plot<- ggplot(prediction_data, aes(x= SalePrice, y= ybar, col = diff)) +
  geom_point() +geom_line(aes(y= y))  +
  scale_color_gradient(name = "|y - ybar|")+
  xlab("y") +
  ylab("y-ybar") +
  ggtitle("Linear model residuals")

resid_plot
```



Residual plot for the best performing model

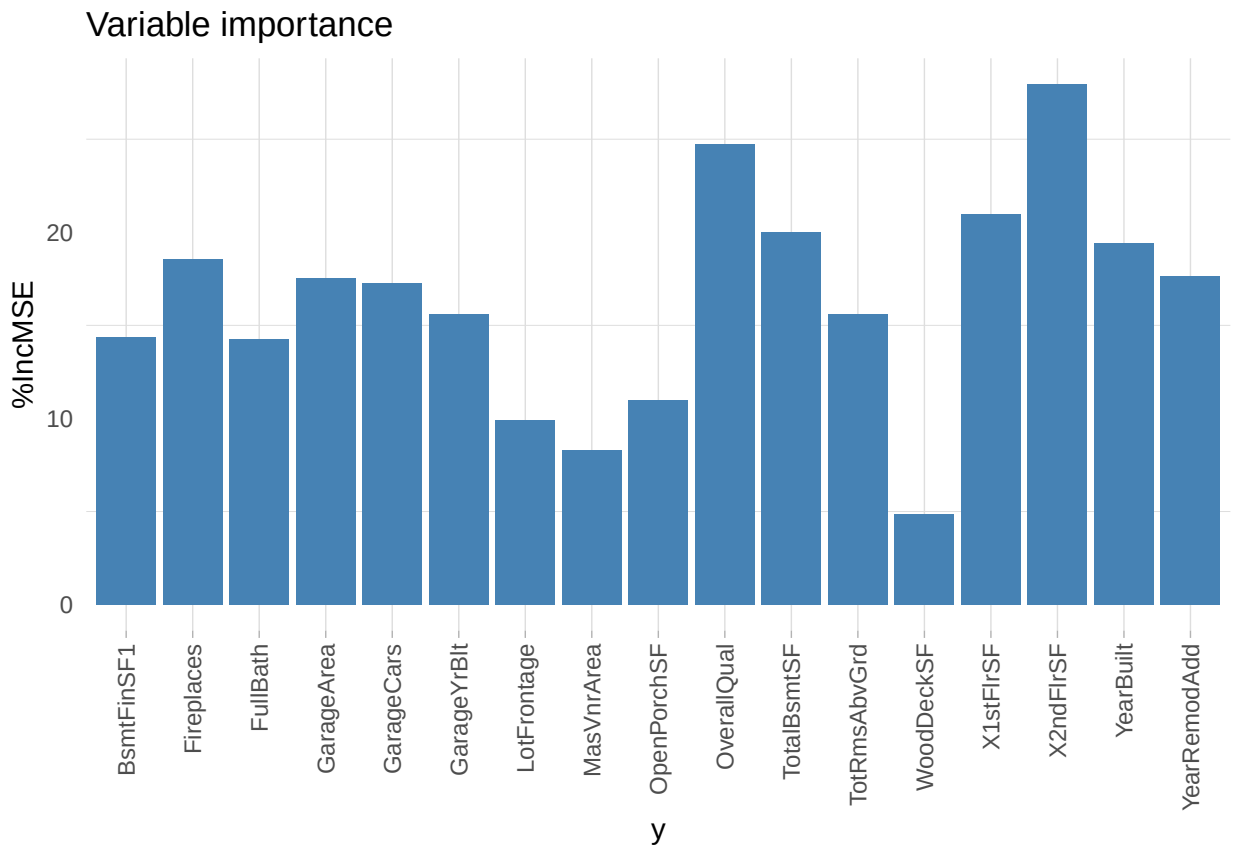
```
modelpred <- predict(rf, newdata = test)
lmdata <- test %>% mutate(y = SalePrice) %>%
  mutate(ybar =modelpred) %>% mutate(diff = abs(y - ybar))

prediction_data <- lmdata %>% filter(diff > 1.5) %>% arrange(desc(diff))


resid_plot<- ggplot(prediction_data, aes(x= SalePrice, y= ybar, col = diff)) +
  geom_point() +geom_line(aes(y= y))  +
  scale_color_gradient(name = "|y - ybar|")+
  xlab("y") +
  ylab("y-ybar") +
  ggtitle("Random Forest model residuals")

resid_plot
```



Compared to the best performing linear regression model, the distribution of the predicted and actual varibales tend to lie along the best line of fit indicating that the random forest has a better fit and will therefore be used during the generation of recommendations regarding the various aspects of pricing in Ames, Ohio.

From the variable importance plot below,First Floor square feet of a home, OverallQual (overall material and finish quality used in a home), Second Floor square feet of a home attributes are the most important in predicting the sale price of a home.

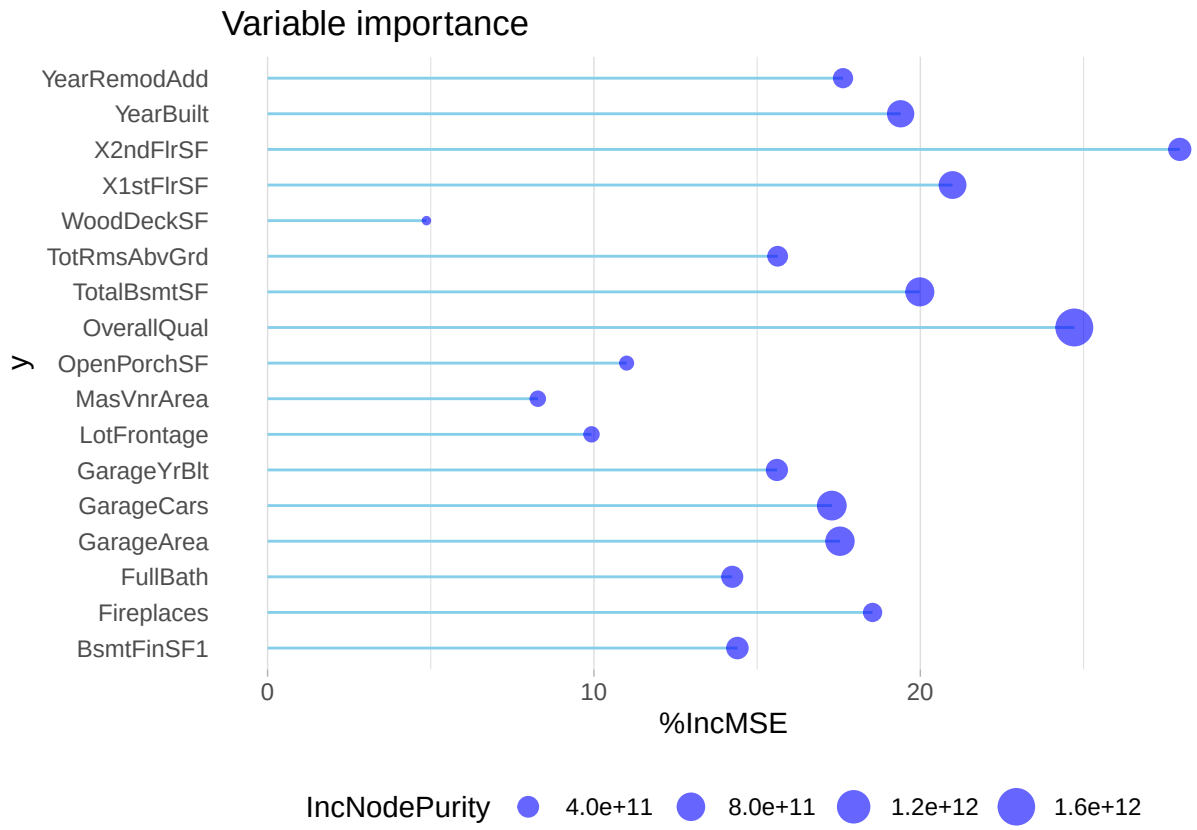
```r
# Get variable importance from the model fit
ImpData <- as.data.frame(importance(rf))
ImpData$Var.Names <- row.names(ImpData)


ggplot(ImpData, aes(x=Var.Names, y=`%IncMSE`)) +
  geom_bar(stat = "identity", fill="steelblue")+
  theme_light() +
  theme(
    legend.position="bottom",
    panel.grid.major.y = element_blank(),
    panel.border = element_blank(),
    axis.ticks.y = element_blank()
  )+
  xlab("y") + ggtitle("Variable importance")+ theme(axis.text.x = element_text(angle = 90, vjust = 0.5,
```



Recommendations and Final Conclusions

```r
ggplot(ImpData, aes(x=Var.Names, y=`%IncMSE`)) +
  geom_segment( aes(x=Var.Names, xend=Var.Names, y=0, yend=`%IncMSE`), color="skyblue") +
  geom_point(aes(size = IncNodePurity), color="blue", alpha=0.6) +
  theme_light() +
  coord_flip() +
```

```
  theme(
    legend.position="bottom",
    panel.grid.major.y = element_blank(),
    panel.border = element_blank(),
    axis.ticks.y = element_blank()
  )+
  xlab("y") + ggtitle("Variable importance")
```



The price of properties is influenced by many factors ranging from economic to the various characteristics of the dwelling space. Inng the price of a home. Based on the correlation table given below, it is observed that the overall overall material and finish quality used in a home has a strong positive correlation with the sale price of the house indicating that, an increase in the overall quality corresponds to an increase in the sale price of a home. Similarly, with a strong positive and moderate positive correlation, an increase in the First and second Floor square feet of a home respectively, corresponds with an increase in the sale price of a home.

```
kable(corl)
```

	Variable	Correlation
2	LotFrontage	0.3347709
4	OverallQual	0.7909816
6	YearBuilt	0.5228973
7	YearRemodAdd	0.5071010
8	MasVnrArea	0.4726145

	Variable	Correlation
9	BsmtFinSF1	0.3864198
12	TotalBsmtSF	0.6135806
13	X1stFlrSF	0.6058522
14	X2ndFlrSF	0.3193338
16	GrLivArea	0.7086245
19	FullBath	0.5606638
23	TotRmsAbvGrd	0.5337232
24	Fireplaces	0.4669288
25	GarageYrBlt	0.4667537
26	GarageCars	0.6404092
27	GarageArea	0.6234314
28	WoodDeckSF	0.3244134
29	OpenPorchSF	0.3158562

To this end, the following recommendations are made:

1. Before listing a home, owners and realtors need to conduct a thorough renovation to ensure that the overall overall material and finish quality used in a home have the best quality. This as noted in this study will increase the value of the property.

2. Developers of homes in Iowa need to take into consideration the aspect of expansion and utilization of development spaces. This will allow owners to expand various parts of the home including installing baths, swimming pool, and the floor area among other amenities.

3. Newer houses attract higher prices. Therefore, sellers sshould consider taking into consideration the range of age of houses listed in Iowa at the time of the sale so as to offer competitive prices.

References

Cheng, F. et al., 2021. A Review on Data Preprocessing Techniques Toward Efficient and Reliable Knowledge Discovery From Building Operational Data. Frontiers in Energy Research, 9(2021).

Cock, D. D., 2011. Ames, Iowa: Alternative to the Boston Housing Data as an End of Semester Regression Project. Journal of Statistics Education, 19(3), pp. 1-15.

Grum, B. & Govekarb, D. K., 2016. Influence of Macroeconomic Factors on Prices of Real Estate in Various Cultural Environments: Case of Slovenia. Procedia Economics and Finance, 39(2016), pp. 597-604.

Johansson, A., 2017. 6 factors that influence a home's value, s.l.: inman.

O'Brien, R. M., 2007. A Caution Regarding Rules of Thumb for Variance Inflation Factors. Quality & Quantity, 41(2007), pp. 673-690.

Tupenaite, L., Kanapeckiene, L. & Naimaviciene, J., 2017. Determinants of Housing Market Fluctuations: Case Study of Lithuania. Procedia Engineering, 172(2017), pp. 1169-1175.

Vittinghoff, E., Glidden, D., Shiboski, S. & McCulloch, C., 2012. Regression Methods in Biostatistics: Linear, Logistic, Survival, and Repeated Measures Models. 2nd ed. s.l.:Springer.

Wittowsky, D., Hoekveld, J., Welsch, J. & Steier, M., 2019. Residential housing prices: impact of housing characteristics, accessibility and neighbouring apartments – a case study of Dortmund, Germany. Urban, Planning and Transport Research , 8(1), pp. 44-70.