Abstract

The analysis of the house data set provides valuable insights into the real estate market, allowing individuals to make the good decisions about buying or selling properties. By performing data analysis, it is possible to identify trends and patterns within the data. This report includes the descriptive statistics and visual summaries of the attributes

Introduction

As a Data analyst at an estate agency , as a team we have performed the data analysis on the given house data set using R Programming and Machine Learning algorithms. By using this we can predict the house prices in future and can observe how the current housing market is doing. Before performing some machine learning algorithms, it is mandatory to perform EDA (exploratory data analysis), Data visualization and Data cleaning process.

The given “house_data.csv” data set has a total of 1460 observations and 51 variables. After Exploring the whole data using data analysis techniques there are some missing values and outliers in the given data. Firstly we need to handle all these missing values with a suitable imputation method based on the context of the data otherwise these missing values in the data will degrade the accuracy of the machine learning algorithms.

Data Exploration

It is the very first step in every data analysis process, It involves loading the data set and viewing its dimensions and structure of the data. We can explore top and bottom rows of the dataset to see the values. As R is popular open-source programming language for statistical or data analysis, it has many built in powerful libraries like dplyr, tidyverse, ggplot2 etc. Some of the data exploration we carried out are:

a. Loading the dataset and viewing its dimensions using **dim()**. This gives dimensions of the dataset i.e., No: of observations and variables.
b. Getting the Structure of the data using **str ()**. This shows the data with the datatype.
c. Finding the count of numeric and character variables will give the overview of the data we are using.
d. Filtering out numeric and character variables and storing them in separate data frame to find the summaries of both using **summary ()**.

e. We also explored the data by visualizing it based on some important variables like Sale Price.

f. Checking for missing values and visualizing the data using boxplot for checking for outliers in the data.

Overall, Data Exploration gives a clear and comprehensive overview of the dataset we want to analyse and we also get to know the summary statistics of each important variable . This also includes Data visualization where we can visualize the data in different perspectives to get more insights of the data and how its attributes are related to each other. For example, see in the **Figure 1** below.

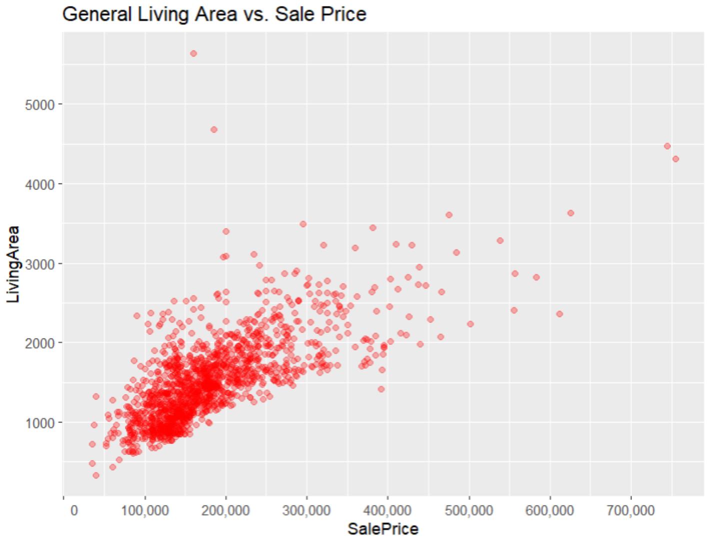


Figure 1: Scatter plot (Living Area Vs Sale Price)

This shows that how the sale price and Living area of the house are related to each other. By this simply we can say that as Living Area increases sale price also increases, but there are some variables which are not following this trend we can consider them as outliers.

In the house dataset, we have 23-numerical and 28-character observations. We need to convert these character observations to categorical observations because they are memory efficient and gives better performance rather than character variables and also we can get better visualization from the categorical variables. This can improve the accuracy of the model we implemented. In those we observed there are missing values in these observations which affects the accuracy of machine learning model and leads to in accurate and incorrect predictions. So, we need to handle these missing data by imputing them based on the context of the data. The following **Table 1** shows the percentage of missing values in the dataset.

Variable	Missing percentage
PoolQC	0.995205479
MiscFeature	0.963013699
Alley	0.937671233
Fence	0.807534247
LotFrontage	0.17739726
GarageType	0.055479452
GarageCond	0.055479452
BsmtQual	0.025342466
BsmtCond	0.025342466
MasVnrArea	0.005479452

Table 1: Percentage of missing values

In the above table the first four columns in red colour represents that they have percentage of missing values above 80 . So, we need to handle these missing values in our data to get good statistical summary of data and accuracy of the model we implemented on the data.

Handling Missing values

Missing data is a data quality problem. In order to get good results we need to handle them efficiently by some imputation methods such as Mean, Median ,Mode or assigning a new group value for those missing categorical variables.

Data cleaning is the most important step in any of the data analysis or machine learning project. Based on the given data set there are a total of 5910 missing values. We have carried out some steps to handle these missing values.

i. Finding the percentage of missing values of each attribute
ii. If the percentage exceeds 80%, dropping those columns won't affect the accuracy.
iii. Applying the Imputation methods for those columns having fewer missing values.

In the dataset, there are four columns **PoolQC, Alley, MiscFeature, Fence** that exceeds 80% of missing values as shown in **Figure 1**.So dropping those columns will be good option.

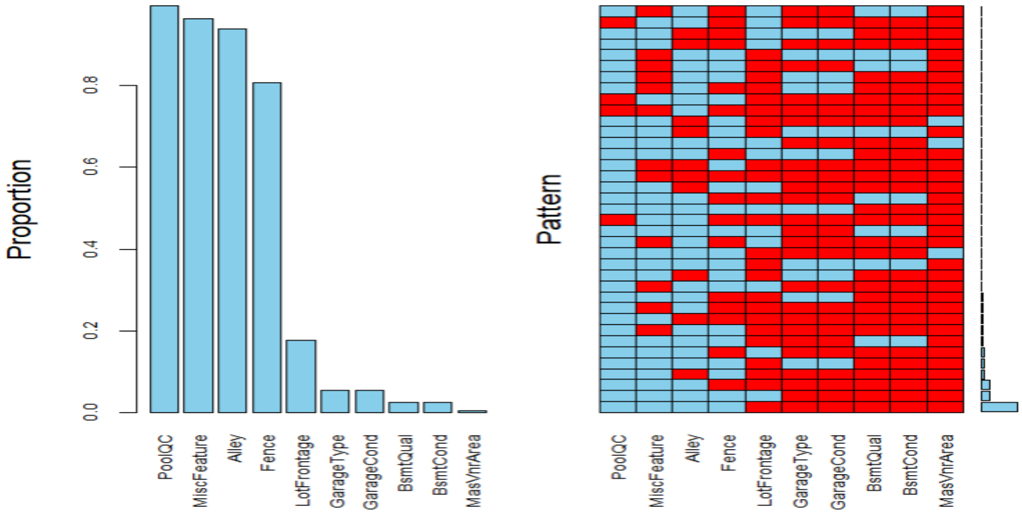


Figure 2: percentage of Missing values

The above figure shows the distribution of missing values. It is done by **VIM** and **MICE** package in R which gives clear visualization of missing values. There are missing values in LotFrontage, GarageType, GarageCond, BsmtQual, BsmtCond, MasVnrArea, but they can be considerable. We can use some imputation methods to handle them if it is numerical variable, we can impute the values based on mean and median, if it is character (Categorical) variable we can impute the most frequently used one or assign a new group it depends on the context of the data. In this, we imputed the LotFrontage using the Median value by observing its as in **Figure 3.** we can see that is a continuous and skewed distribution towards right, this indicates it has outliers so in this median will be a robust measure to deal with the outliers so by using **median ()**, we have imputed the value for the LotFrontage.

For MasVarArea we have imputed it with mean value as the data is slightly skewed and at the same time it is not exactly as symmetric as it starts from a point of Y-axis and smoothly down towards the X-axis as shown in **Figure 3**. So, in this case we can use either median or mean to impute the values.

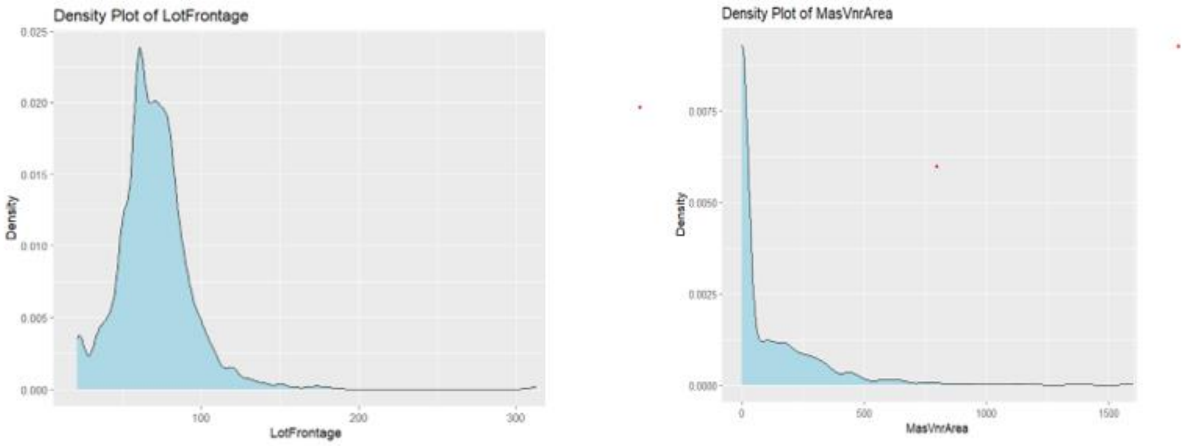


Figure 3: showing the density plots for LotFrontage and MasVnrArea

As mean is the more informative measure of central tendency,
we used mean to impute the MasVnrArea by using **mean (). For** GarageType, GarageCond, BsmtCond, BsmtType as they are categorical variables, we have imputed them as No garage and No Basement respectively as it is given in description of data set.

Checking for Outliers

Outliers are the data observations that are significantly different from other observations in the dataset**.** When working on any data set it is very important to check for outliers because they can have a big impact on the results of statistical analysis. We can detect them by using the combinations of statistical measures such as IQR(Inter Quartile range) and the data visualizations mainly based on boxplots. The data points that fall outside of the IQR values are considered as outliers.

In the house data set we have used box plot to detect the outliers and by using IQR we can remove the outliers. For an Instance, we have a found an outlier in the SalePrice by using boxplot. We can get this by using **ggplot2** library in R of it as shown in **Figure 4.**

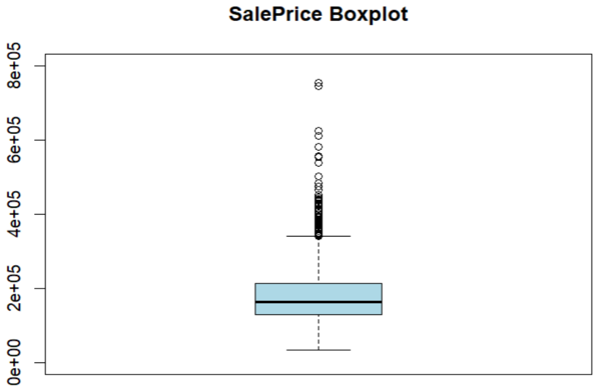


Figure 4: Boxplot of Sale Price

This plot shows that the line inside the box is median of the data. The whiskers extend from the box to show the range of the data. By default, the whiskers extend to 1.5 times the interquartile range (IQR), which is the distance between Q1 and Q3. Any data points that lie outside the whiskers are considered outliers. These outliers may indicate extreme values or errors in the data.

By getting the summary of the sale price using **Summary()** in R , we can get to know the Descriptive statistics of the saleprice such as median, mean, 1st quartile etc. as shown in the **Table 2** below.

Min	1st Quartile	Median	Mean	3rd Quartile	Max
34900	129975	163000	180921	214000	755000

Table 2: Descriptive statistics of sale price

Now by using values we can find IQR to remove the outliers by below calculations:

IQR_SalePrice = 214000 - 129975

Lower = 214000 -1.5*IQR_SalePrice

Upper = 129975 + 1.5*IQR_SalePrice

By filtering the data using sale price by **SalePrice >Lower & SalePrice < Upper .**we can remove the outliers using this . To visualize this, we can plot the box plot again after this Statistical calculation as shown in **Figure 5**.

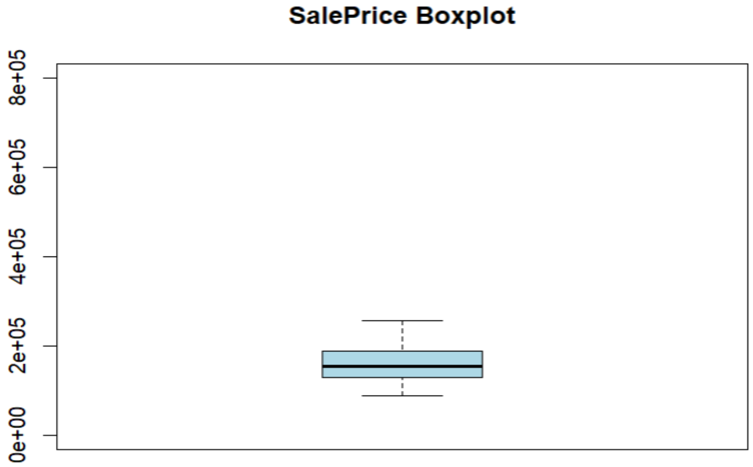


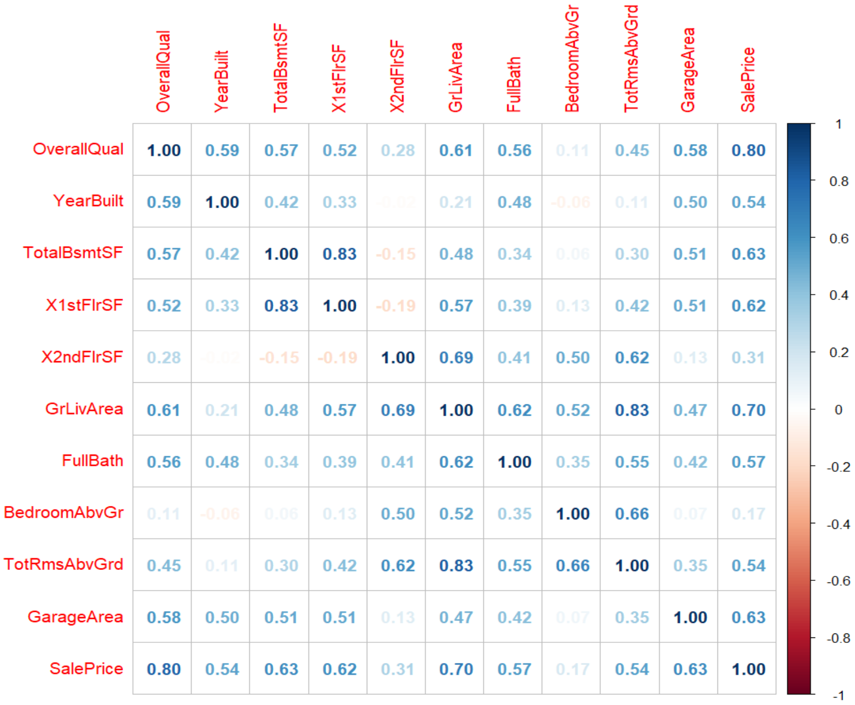
Figure 5: Boxplot of saleprice (without outliers)

In this dataset, we have identified the outliers but we didn't remove the outliers in the data, because there is a small amount of data (1460*51). So, we removed only the missing values that exceed 80%. Now we have the new dimensions of the data i.e.,(1460*47) as we removed four columns that have missing values.

Correlation Analysis

The main goal of correlation analysis is to identify any relationships or associations between different variables in the dataset. we can calculate the correlation coefficients of each variable all these coefficients lies between 0 and 1 as shown in the **Figure 6.** We can plot that correlation plot using **corrplot ()** library in R**.** This correlation analysis can be done on numerical variables in the house dataset. As in the house data we already saw that we have 23 numerical variables.

In this correlation plot, we have removed Id column as it doesn't give any insights into the data and we also taken only the variables in which their correlation coefficients lie between 0.5 to -0.5.



	OverallQual	YearBuilt	TotalBsmtSF	X1stFlrSF	X2ndFlrSF	GrLivArea	FullBath	BedroomAbvGr	TotRmsAbvGrd	GarageArea	SalePrice
OverallQual	1.00	0.59	0.57	0.52	0.28	0.61	0.56	0.11	0.45	0.58	0.80
YearBuilt	0.59	1.00	0.42	0.33	-0.02	0.21	0.48	-0.06	0.11	0.50	0.54
TotalBsmtSF	0.57	0.42	1.00	0.83	-0.15	0.48	0.34	0.06	0.30	0.51	0.63
X1stFlrSF	0.52	0.33	0.83	1.00	-0.19	0.57	0.39	0.13	0.42	0.51	0.62
X2ndFlrSF	0.28	-0.02	-0.15	-0.19	1.00	0.69	0.41	0.50	0.62	0.13	0.31
GrLivArea	0.61	0.21	0.48	0.57	0.69	1.00	0.62	0.52	0.83	0.47	0.70
FullBath	0.56	0.48	0.34	0.39	0.41	0.62	1.00	0.35	0.55	0.42	0.57
BedroomAbvGr	0.11	-0.06	0.06	0.13	0.50	0.52	0.35	1.00	0.66	0.07	0.17
TotRmsAbvGrd	0.45	0.11	0.30	0.42	0.62	0.83	0.55	0.66	1.00	0.35	0.54
GarageArea	0.58	0.50	0.51	0.51	0.13	0.47	0.42	0.07	0.35	1.00	0.63
SalePrice	0.80	0.54	0.63	0.62	0.31	0.70	0.57	0.17	0.54	0.63	1.00

Figure 6: Correlation analysis of house dataset

From this correlation plot, we can say that SalePrice and OverallQual has a highest of 0.80 Correlation coefficient which means these two variables are highly related to each other and also we can see that GrLivArea (Living Area) and sale price has a correlation coefficient of 0.70 which is high. Similarly, we can find which variables are highly correlated to each other. In case of Positive correlation, if one variable is increasing and the other also will be in same trend. For negative correlation it will be vice versa. By this plot we can easily visualize how two variables are correlated to each other. There is a statistical term called multi collinearity means high degree of correlation between two or more predictor variables. Having this will be a problem in finding the results.

Overall, after performing Data exploration, Data cleaning and Data visualization techniques on the house data set , we came to know that what are variables present in data set and what they mean and how they related to each other. Now the dataset was relatively clean with few missing values that are also imputed with the appropriate methods like Mean and median as discussed in data cleaning section. Now our data is ready to do further analysis like predicting some target variables based on some dependent variables. We can also perform the techniques like Regression and Classification in order to get more insights of the data. On this house data set we are using some classification techniques like Logistic regression and Naïve bayes for predicting the overall Condition of the house and using Regression algorithms such as Linear regression and Random Forest to predict the sale price of the house based on some features. Before performing these, we have divided the houses into three categories namely “Poor”, “Average”, “Good” based on the Overall Condition of the house which defines the condition in form of rating i.e., numeric variable. Convert the newly created column to factor variable and converted all character to factor variables. Before applying any of the machine learning models, we need to split the cleaned dataset into train and test parts. Mostly 80% for training and 20% for testing the model. In R this can by using **Caret** library.

Logistic Regression

In the house data we have created a new column based on the Overall condition of house and named it is as New_cond it has only 3 values poor, good, average. This variable has three levels. Logistic regression is used when the dependent variable has only two factors. But here we have 3 levels in the dependent variable i.e., New_cond. So, we used extension of the logistic regression i.e., Multinomial Logistic regression to predict the New_cond of the house. This can be used for classification problems used to make predictions on new data by estimating the probabilities of each category based on the predictor variables. It can be often used in the field of marketing where there are multiple categories of interest. In R this can be implemented by using **multinom()** method which is in the **nnet** library. We need to train the data on this model by specifying the dependent variable i.e., New_cond and measured the performance metrics like accuracy, F1-score , AUC_ROC as shown in **Table 3** using test data we got an accuracy of 0.959.

Category	Accuracy	F1_score	AUC_ROC
Average	0.959	0.973	0.974
Good	0.959	0.959	1
poor	0.959	0.364	0.868

Table 3: performance metrics of the logistic regression

Based on these metrics for each category , we can say that the model is good at predicting the different categories of the New_Cond. F1 score considers both the precision and recall of the model. A higher F1 score indicates the better balance between Precision and Recall. AUC-ROC is a metric that checks how well the model can differentiate between the classes. A big score of it indicates the model has better capacity to distinguish the classes. So, this model is good at the accuracy and ability to differentiate between Poor, Average, Good classes based on the overall condition of the house.

Naive Bayes Classifier

It is commonly used in machine learning algorithm for classification tasks because it is both simple and computationally efficient. It works by Bayes' theorem to calculate the probability of each class given the input features. This classification algorithm can be well suited for predicting the overall condition of houses because this problem involves data with both categorical and continuous variables, such as the age of the house, the number of bedrooms and bathrooms, and information about heating and cooling systems. This algorithm can handle this kind of mixed data, and provide accurate predictions. We can implement this in R by loading **e1071** library and using **naiveBayes()** function we got an accuracy of 0.835. Apart from this we also found some metrics such as Recall, Precision, F1 score and auc_roc value shown in the **Table 4** below. These metrics help us to compare with other models performance on the house data set.

Metric	Value
Accuracy	0.8350515
Precision	0.8053097
Recall	0.9784946
F1 Score	0.8834951
AUC_ROC	0.9287796

Table 4: Performance of Naive bayes

By this metrics we can compare both the model's performance. By considering the accuracy of the models we can say that Naive bayes outperforms Logistic regression but not much difference in the accuracy.

Linear Regression

This statistical technique is used to model the relationship between the dependent variables and independent variables. As this is a part of the base package in R. It uses the ‘**lm()**’ function to fit the model. In this analysis, we predicted the SalePrice of the house using linear regression algorithm. Before applying the model, we need to split the data into train and test data using **caret** package in R and pre-processed the test and training data using **center** and **scale** method which helps the model to evaluate the better results. We also dropped some factor (Categorical) columns which has lesser levels ,they are not necessary to produce better results.

In the house data as we are predicting the saleprice of the house Here, Sale price is target variable and there are independent variables which are used to predicted the sale price of the house. By considering the saleprice in test data i.e., actual saleprice of the house and after training the data, we can predict the sale price of the house.

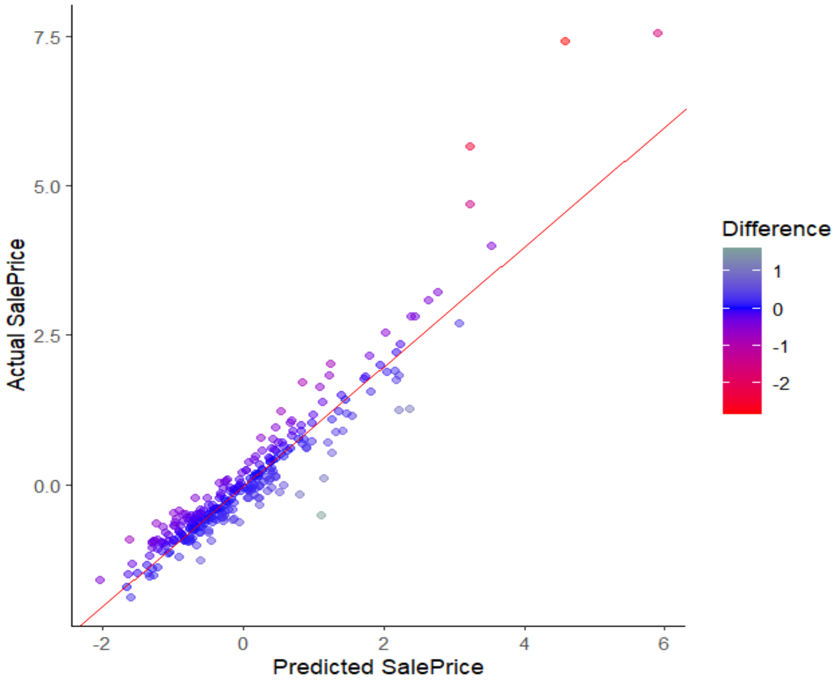


Figure 7: Difference between the actual and predicted saleprice

To see the differences between the actual and predicted SalePrice we can use **ggplot2** library in R to visualize it as shown in **Figure 7.** From the above figure we can observe that most of them are in thick blue colour that indicates the actual and predicted sale price are same and also we can see that some are in red colour and light grey colour means that they are differ in the predicted saleprice of house. We can say that most of the saleprice of the houses are predicted correctly using the linear regression model.

Random Forest:

It is a machine learning technique that is used for regression tasks that involves the building d ecision trees. This works by creating many decision trees, each tree trained on the randomly selected subsets of training data and predictor variables to get better accuracy of the model. In R, this can be done by using '**randomForest**' library. As the house data set has both continuous and categorical data, the random forest algorithm works well on this mixed type of data sets. Once the model is trained by this algorithm, we can predict the saleprice using this algorithm, to know the accuracy of the model we find some metrics like R-Square d value and Root mean square error (RMSE). Among these R-Squared (R2) value defines how much variance in SalePrice can be explained by the remaining features of the dataset. The higher R2 value indicates that model is a good fit for the data. By using these metrics, we can evaluate the performance of the above two models as shown in the **Table 5**.

	RMSE	**R2**
Linear Regression	0.3947491	0.896292
Random Forest	0.13947	0.832769

Table 5: performance metrics of both models

By viewing these performance metrics, R2 value is higher for the Linear regression. So, in Predicting the sale Price of the house random forest out performs the Linear regression.

Re-Sampling Methods

These methods involve statistical techniques that estimate the performance of the model by randomly sampling subsets of the data multiple times. The main aim of using this resampling methods is to evaluate how accurately a model will perform on new data. These methods are especially helpful when the dataset is limited in size or when the objective is to determine the generalization error of a model. We have used 2 methods cross-validation and bootstrapping on the two models that are used to predict the sale price of the house. To use these methods we can use **"ipred"** library and **errorest()** function in R.

We have used cross validation to estimate the misclassification error of the random forest mo del .This involves repeatedly splitting the data into training and testing sets to evaluate the per formance of the model on new data. In order to apply this method set the estimator argument to **"cv"** by default it used 10-fold cross validation. The RMSE value obtained from this estim ation is 29053.34, from this we can say that on average the sale price will differ by 29053.34 compared to actual sale price. Similarly, we have done the same process using the Boot strapping technique to get misclassification error. we got RMSE value is 29890.71. So by using these resampling methods, we have validated the model to ensure that the model can work on new or unseen data.

Further Analysis

In this, we tried find a relation between the Living area and saleprice of house. As Sale Price of the house increases, the living area also increases, we predicted the size of living area with the sale price of the house. As we discussed about the correlation analysis earlier, there is a strong correlation between living area and price of house. We also seen that not only with price some other features like Lot Area, Total bathrooms, Total Bedrooms , Kitchen and Total Rooms the living area is positively correlated. So, it’s good to use all these as independent variables to predict the living area of house using linear regression. We got the R2 value for this model as 0.7808 which shows that this model is good fit for this data. In this whole process correlation analysis helped a lot in determining the independent variables for predicting the living area of the house.

Summary

By using this house dataset, we analysed the information for predicting the sale price and condition of houses based on their features, such as location, living area, and number of bedrooms by using machine learning techniques such as linear regression and random forest can be used to make accurate predictions. This analysis has practical applications in real life scenarios such as buying or selling a house, property assessment, and real estate investment. Making informed decisions based on accurate predictions can help individuals avoid financial risks and make profitable investments.