**COMP 1433: Introduction to Data Analytics**

**Final Group Project**

**Option 2 (Data Analysis for House Prices)**

Group Members:

20055588d Pang Ting Pong

20105606d Mak Ho Lun

20041645d Alibek Kuantkhan

Motivation

Living in a comfortable house is the life goal of many people. If somebody asks a random home buyer to describe their dream house, he/she can provide a lot of conditions and demands. However, the most crucial issue concerned is always the house price. Applying advanced data analyzing tools and models, a more precise final result for house prices can be computed considering different key factors. There are several advantages of having a house price prediction model. First of all, buyers will be able to easily navigate through the market of real estate depending on budget, location, and demand. Obviously, this approach can speed up the transaction process, because every customer has the budget and dream house in his/her mind. On the other hand, the real estate agents (sellers) also benefit from the model in terms of business decisions. While tracking the market cycles, the accurate predictions of the liquidity of a project are beneficial and efficient. In other words, based on the house's condition and location, the sellers will be able to determine its profit and investment return plan. Therefore, the model is the means of meeting customers’ requirements, which is essential in any business, while the business can also benefit from the outcomes of the model.

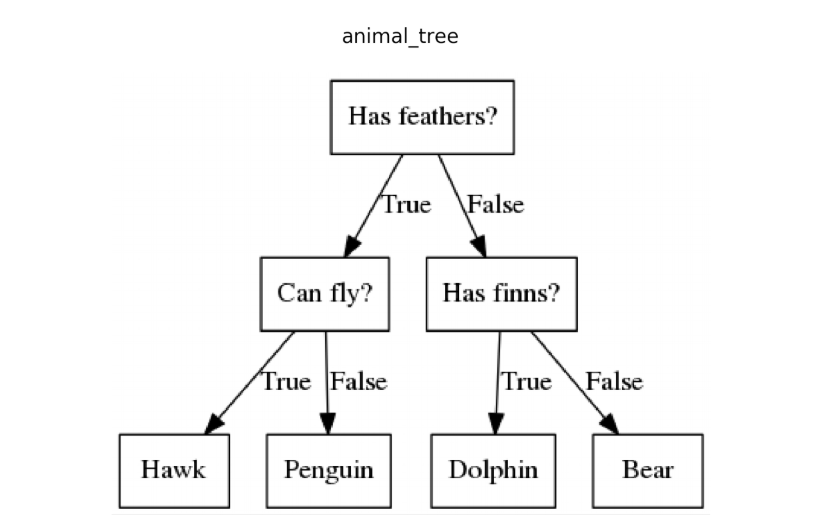
Description

*Project Objective*

Proven that price negotiations have much more influences than other key features of a house, with the usage of given datasets, predicting the final price of each residential home in Ames, Iowa is the main objective of this project. The tools we are using and a brief introduction of the XGBoost model will be provided below.

*Decision Tree*

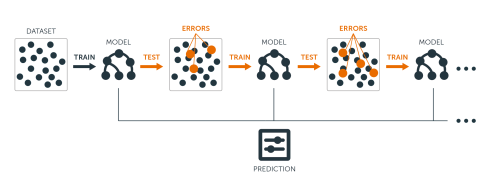
To start with, in order to understand the gradient boosting algorithms, we need to first understand what a decision tree is because it is the single widely used predictive method of gradient boosting. A decision tree is a special flowchart-like structure that allows us to select between the processes. Each tree has leaves and roots. The path from the root to the leaf is called classification (it is where a decision is made). There are many types of decision trees.



The main function of decision trees is to make a decision regarding the problem such as classification, regression or prediction. This tree, for example, is used to differentiate between Hawk, Penguin, Dolphin and Bear. The machine learning model uses and builds many alike decision trees to predict or identify.

*Model (XGBoost)*

XGBoost - Extreme gradient boosting is the regressive machine learning algorithm that builds up on weak predictive models, typically decision trees. It is ideal for large datasets as the provided one, because second order approximation is used. It is also one of the most precise models since it averages simpler predictive ensembles. As we know, averaging decreases variance, so the XGBoost model is especially effective for datasets with high bias and low variance.

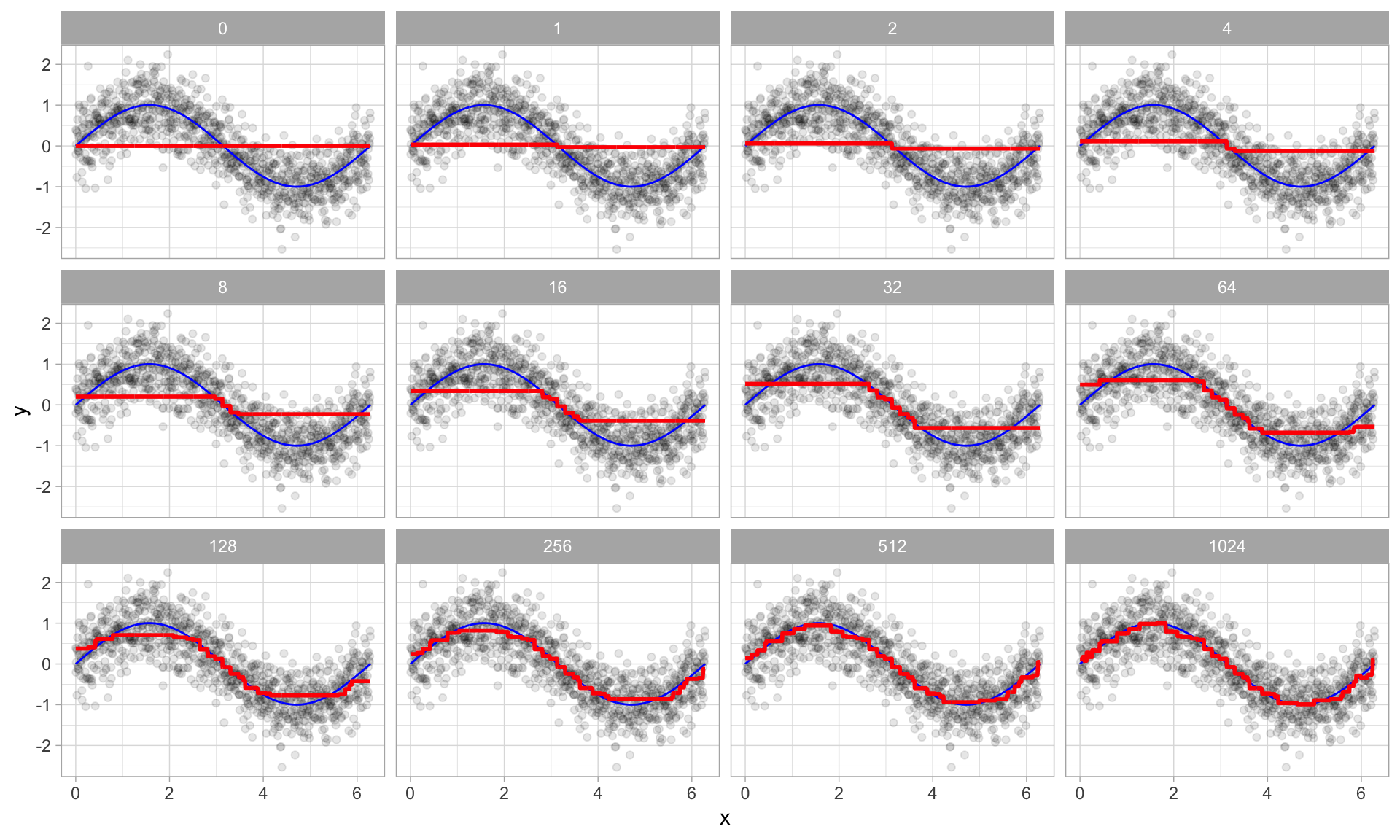


The main idea here is to sequentially add new models to the ensemble. The model starts with a weak tree that has high bias and low variance. Then it runs data through the model to see where it made the biggest mistake. The next model builds up on it, fixing the previous mistake. The process goes on and on, until the needed accuracy is reached.

Sequential training with respect to errors: Boosted trees are grown sequentially; each tree is grown using information from previously grown trees to improve performance. This is illustrated in the following algorithm for boosting regression trees. By fitting each tree in the sequence to the previous tree’s residuals, we’re allowing each new tree in the sequence to focus on the previous tree’s mistakes.

1. Fit a decision tree to the data:
2. We then fit the next decision tree to the residuals of the previous:
3. Add this new tree to our algorithm:
4. Fit the next decision tree to the residuals of
5. Add this new tree to our algorithm:
6. Continue this process until some mechanism (i.e. cross validation) tells us to stop.

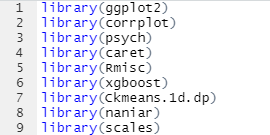
The final model here is a stagewise additive model of *b* individual trees:



This picture illustrates the “sin” relationship between x and y with some errors. The first tree loosely aligned on the graph, almost like a random guess. However, as decision trees build up, the accuracy slightly improves with every iteration.

Implementation

1. **libraries (only important ones are explained)**



library(knitr) - The R package **knitr** is a general-purpose literate programming engine, with lightweight API's designed to give users full control of the output without heavy coding work. It combines many features into one package with slight tweaks motivated from my everyday use of Sweave.

library(ggplot2) - The R package **ggplot2** is a very useful plotting library that contains barplots, charts, scatterplots and a wide range of hyperparameters to tune the visualisation.

library(plyr) - The R package that makes data manipulation, conversion, split and gathering much easier.

library(xgboost) - The R package that allows us to implement the XGBoost model on our data. It has many hyperparameters to tune.

library(Rmisc) - The R package Rmisc is a collection of functions useful for data analysis and utility operations.

1. **Input data**

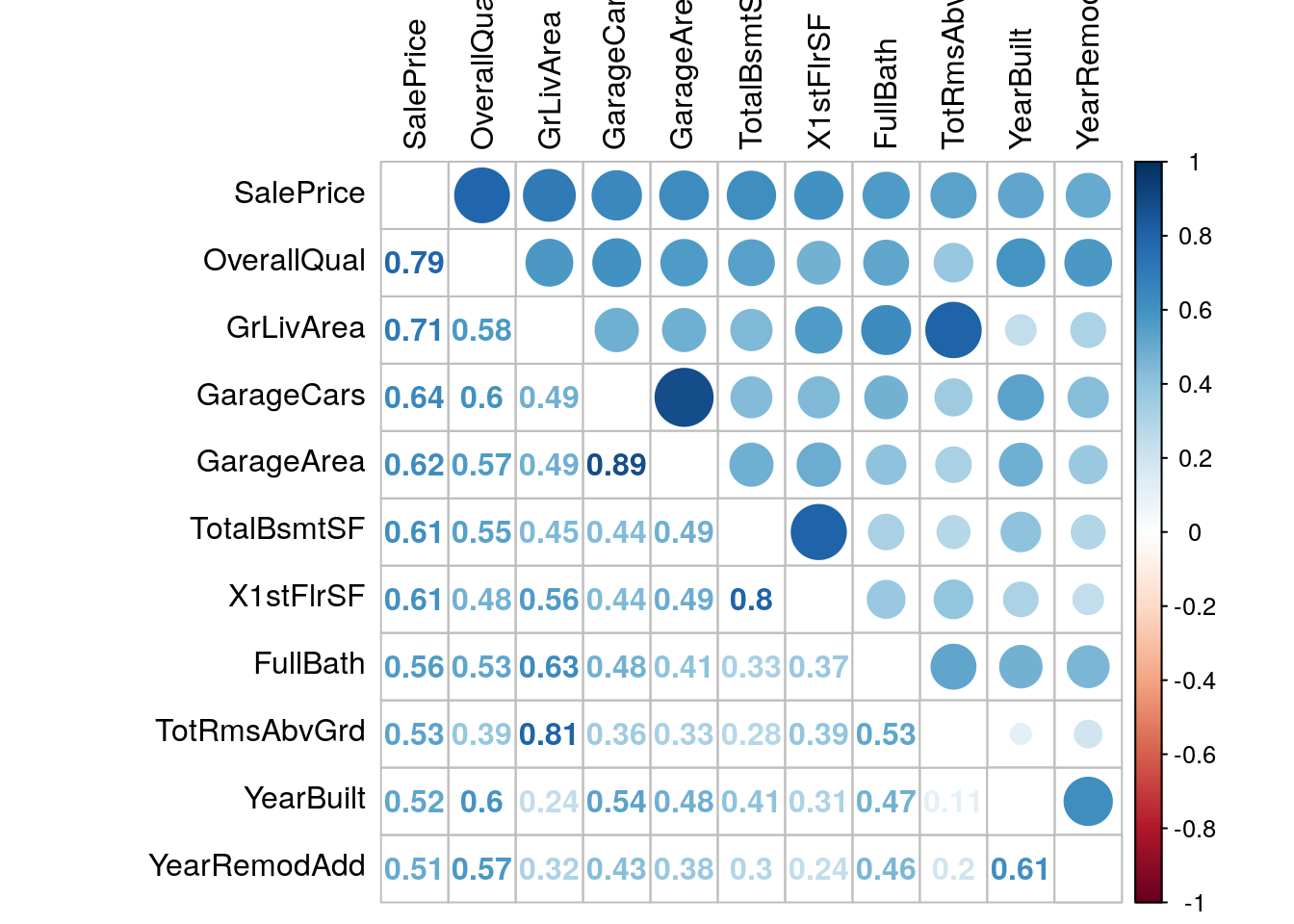
****

* Reads from the “train.csv” and “test.csv” are saved into train and test data frames respectively.

1. **Exploring data**

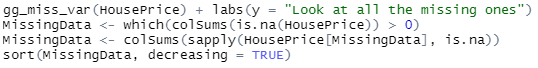
The most important variable, which we need to predict is “SalePrice”. Let’s explore it. As we can notice from the variable distribution, the prices of houses sold are skewed to the left. The reason is that most people can not afford expensive houses. It does, however, form a Guassian distribution, which will aid us in deciding which machine learning algorithm to use later.Normalize distribution by taking log +1 for it.

To build a model we need to first understand which numeric variables have the most influence on the SalePrice variable.



As it turns out, the most important variable for “SalePrice” is “OverallQual” with a correlation as high as 0.79. There are also some variables with high correlation with each other: GarageArea and GarageCar, X1stFlrSF and TotalBsmntSF, ToRmsAbvGrd and GrLivArea. This is understandable, since the variables are mutually influential. For example, an area of a garage depends on a number of cars in a garage etc.

1. **filling missing data with N/A**

****

* To increase the accuracy and make the data “clean” we need to fix or get rid of NA values.



* There are 34 columns that should be fixed. The best decision s to go through each of them, considering that some variables may form a “group”;

1. **Combine some variables in groups**

* Some variables in the dataset derive from another “bigger” variable, so their correlation is high (almost close to 1). For example, there are ground living areas and above ground living areas (1F, 2F, etc.), which means that the bigger variable is equal to the related smaller variables. The aim of this step is to find and group such variables together.

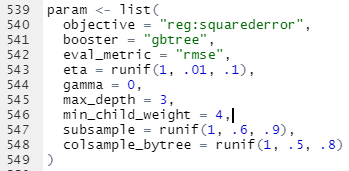
1. **Remove outlier**

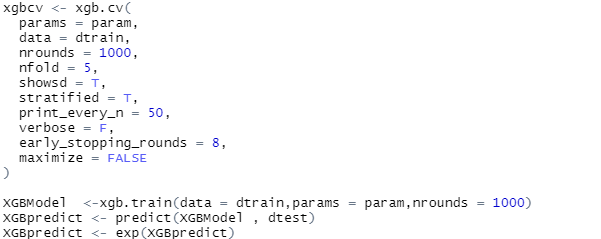
* The outliers are instances of a data that constitute a small fraction of the whole dataset, but their correlation is enough to influence the outcome. In the Data part of the report, the correlation between “SalePrice” and “TotalSqFeet” clearly shows two outliers on the right, and some outliers on the left.

1. **Fix predictors**

* This step is responsible for centering and scaling the true numeric data, and dividing the categorical variables from them; normalizing the skewness of the data, removing data with few observations, fixing the skewness in response variable are crucial steps.

1. **Apply XGBoost**



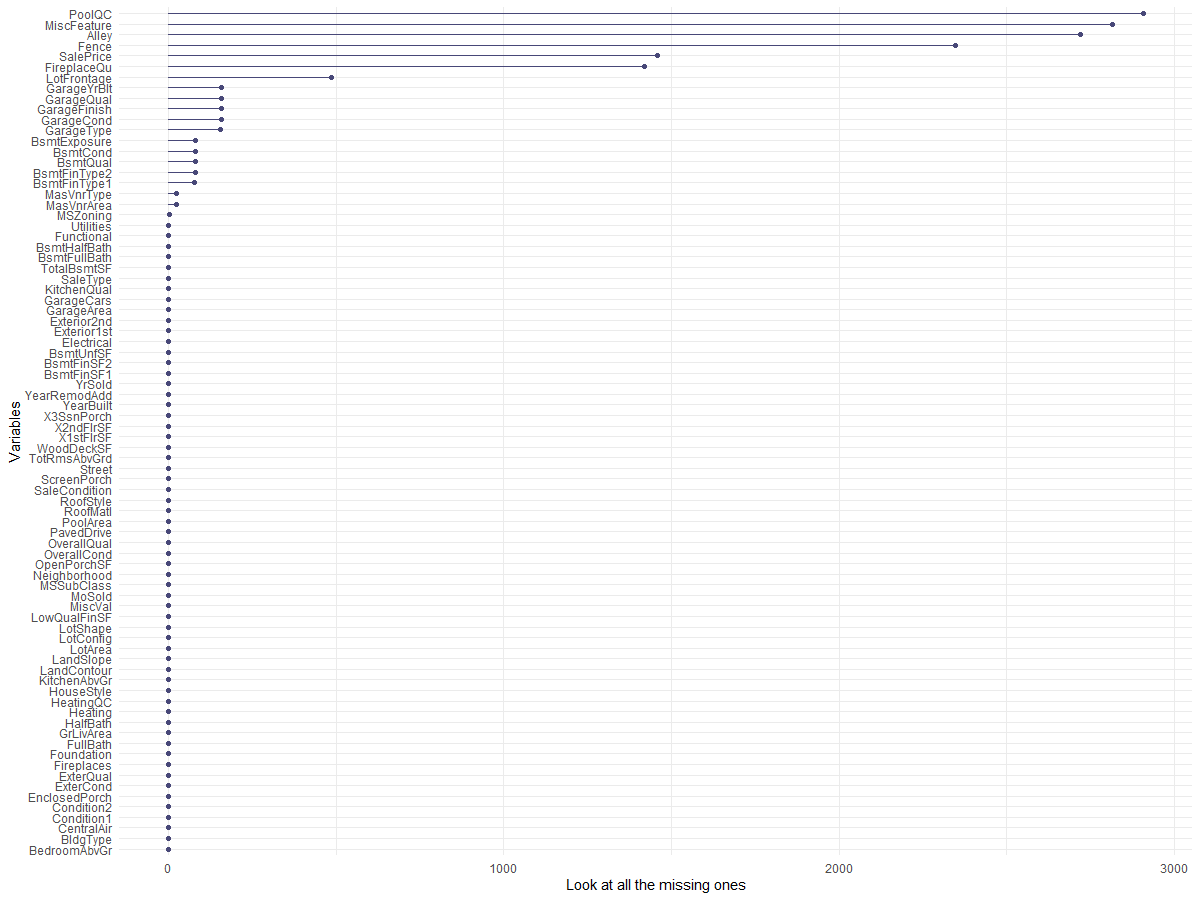
* It's time to put the model to work after the data has been cleaned and corrected. Random search: generate random parameters and replace the old ones with the best. It turns out that scanning for hyperparameters with this method is quite interesting. For regression, rmse is used, and the learning rate, eta, is set to 1,.01,.1. Two parameters that can be used are maximum child weight and minimum child depth. Both of these are essential. Can help to narrow the margin of error. We're taking the best tuned values from the caret cross validation.
* It's time to put the model to work after the data has been cleaned and corrected. Random search: generate random parameters and replace the old ones with the best. It turns out that scanning for hyperparameters with this method is quite interesting. For regression, rmse is used, and the learning rate, eta, is set to 1,.01,.1. Two parameters that can be used are maximum child weight and minimum child depth. Both of these are essential. Can help to narrow the margin of error. We're taking the best tuned values from the caret cross validation.

.

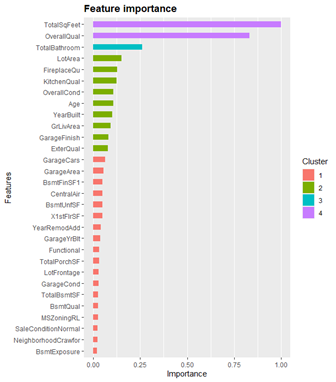
1. **Output the graphs and results**

Data

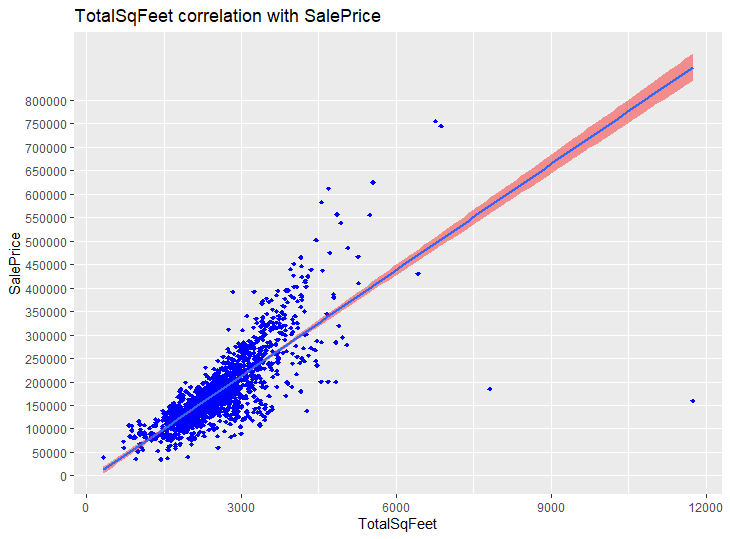
Missing values:



Results and observations



After the computation, the importance of the key features of a house which influences the house prices the most is discovered. A graph as above has been generated, showing that the total square feet of a house is the most important feature, followed by overall quality, total number of bathrooms, etc.



From the graph, a positive relationship between the size and the price of houses in Ames, Iowa.

**Discussions**

When it comes to home buying, using the prediction program to filter the unwanted results is a wiser and faster option compared to traditional methods. Sequentially visiting favourable houses is time-consuming for both buyers and sellers, especially when there are virtual reality tools available to provide 3D views of the interior and exterior of the house. If home buyers consider buying their dream houses, application of the prediction program is recommended because the tool can provide a more accurate house price prediction with different variables considered affecting the house prices.

Here is an example:

*A home buyer would like to buy his dream house with several requirements and he chooses several suitable candidates. He uses the program to predict the house prices of the candidates. After the computation, the predicted house price of each house is computed. From the above analysis, most of the house prices range from $75000usd to $300000usd. If he discovers that some of the candidates are out of his budget, overpriced or out of normal range, the houses will be disqualified from his wishlist. In addition, the houses disqualified may have potential issues which affect the price. For example, it may be a haunted house. Without acknowledgement of the predicted prices, the home buyer may be tricked by advertisement or price tags of real estates.*

From the example, the importance of using a prediction tool is clear. It protects the buyers from losses, provides a clear and accurate house price prediction, and prevents the buyers from making transactions which are over budget.

Nevertheless, buyers are recommended not to fully rely on the results of the prediction program because there are always random accidents and create variables that are out of consideration in the program. Although it is time-consuming, visiting the houses in the wishlists is necessary