FINAL
2016FA_PREDICT_413-DL_SEC59
NITIN GAONKAR

The purpose of our project was to analyze prices of homes with the Ames housing data based on several variables. When people consider buying homes, usually the location has been constrained to a certain area such as not too far from the work place. With location factor pretty much fixed, the property characteristics information weights more in the home prices. There are many factors describing the condition of a house, and they do not weigh equally in determining the home value. In this paper, I will present a modeling process for estimating home values using multivariate linear regression model based on the condition information of the dwellings in order to examine the key factors affecting their values [4]

Studies on home prices have been going on for many years using various models. The traditional and standard model is the hedonic pricing model that says the prices of goods are directly influenced by external or environmental factors in addition to the characteristics of the goods. For housing market analysis, the hedonic price model [9] infers that the price of dwellings are determined by the internal factors (characteristics of the property) as well as external attributes. The method used in this model is multi regression that considers various combinations of internal and external predictors [1, 4, 13]. The predictors may be first-order or higher order (such as Area2) so that the hedonic regression may be a polynomial function of the predictors [2, 7].

Housing price valuation is one of most important trading decisions.

An accurate prediction on the house price is important to prospective homeowners, developers, investors, appraisers, tax assessors and other real estate market participants, such as, mortgage lenders and insurers (Frew and Jud, 2003). Traditional house price prediction is based on cost and sale price comparison lacking of an accepted standard and a certification process. Therefore, the availability of a house price prediction model helps fill up an important information gap and improve the efficiency of the real estate market (Calhoun, 2003).

The data we received is from the Ames housing dataset compiled by Dean de Cock for use in data science education, the data has around 79 explanatory variables The data set contains 2930 observations and a large number of explanatory variables (23 nominal, 23 ordinal, 14 discrete, and 20 continuous) involved in assessing home values. The complete documentation of the data can be found in the below link.

https://ww2.amstat.org/publications/jse/v19n3/decock/DataDocumentation.txt

Below are the few variables from the data set:

Data fields

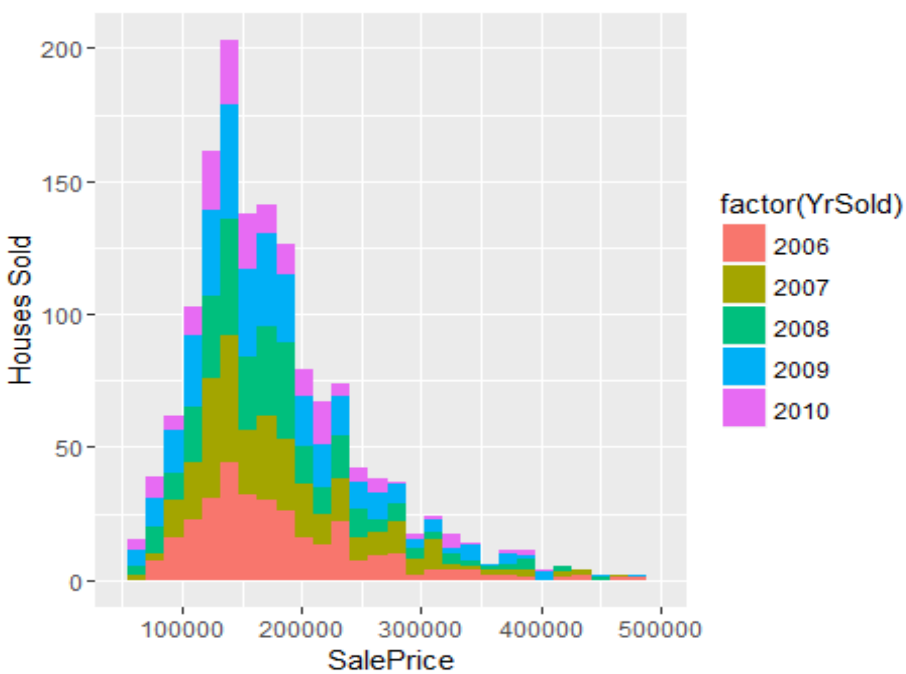
Here's a brief version of what you'll find in the data description file.

- **SalePrice** - the property's sale price in dollars. This is the target variable that you're trying to predict.
- **MSSubClass**: The building class
- **MSZoning**: The general zoning classification
- **LotFrontage**: Linear feet of street connected to property
- **LotArea**: Lot size in square feet
- **Street**: Type of road access
- **Alley**: Type of alley access
- **LotShape**: General shape of property
- **LandContour**: Flatness of the property
- **Utilities**: Type of utilities available
- **LotConfig**: Lot configuration
- **LandSlope**: Slope of property
- **Neighborhood**: Physical locations within Ames city limits
- **Condition1**: Proximity to main road or railroad
- **Condition2**: Proximity to main road or railroad (if a second is present)
- **BldgType**: Type of dwelling
- **HouseStyle**: Style of dwelling
- **OverallQual**: Overall material and finish quality
- **OverallCond**: Overall condition rating

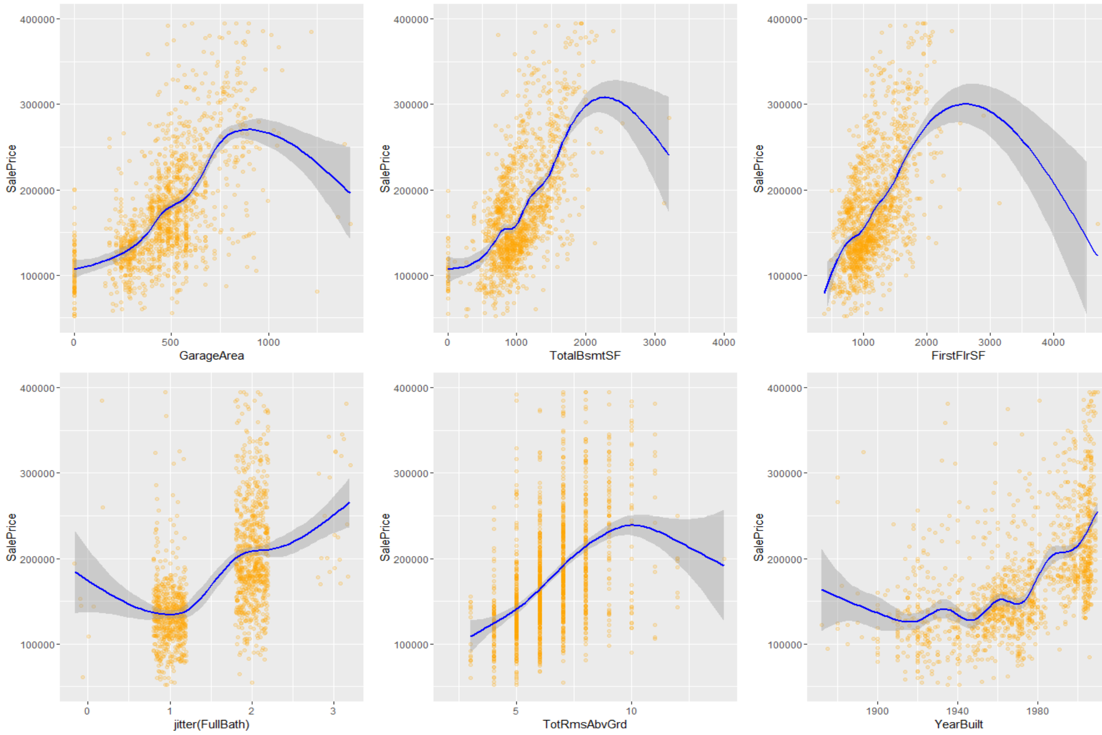
- **YearBuilt**: Original construction date
- **YearRemodAdd**: Remodel date
- **RoofStyle**: Type of roof
- **RoofMatl**: Roof material
- **Exterior1st**: Exterior covering on house
- **Exterior2nd**: Exterior covering on house (if more than one material)
- **MasVnrType**: Masonry veneer type
- **MasVnrArea**: Masonry veneer area in square feet
- **ExterQual**: Exterior material quality
- **ExterCond**: Present condition of the material on the exterior
- **Foundation**: Type of foundation
- **BsmtQual**: Height of the basement
- **BsmtCond**: General condition of the basement
- **BsmtExposure**: Walkout or garden level basement walls
- **BsmtFinType1**: Quality of basement finished area
- **BsmtFinSF1**: Type 1 finished square feet
- **BsmtFinType2**: Quality of second finished area (if present)
- **BsmtFinSF2**: Type 2 finished square feet
- **BsmtUnfSF**: Unfinished square feet of basement area
- **TotalBsmtSF**: Total square feet of basement area
- **Heating**: Type of heating
- **HeatingQC**: Heating quality and condition
- **CentralAir**: Central air conditioning
- **Electrical**: Electrical system

1. Exploratory data analysis:

Below is the distribution of the sale price in the data:

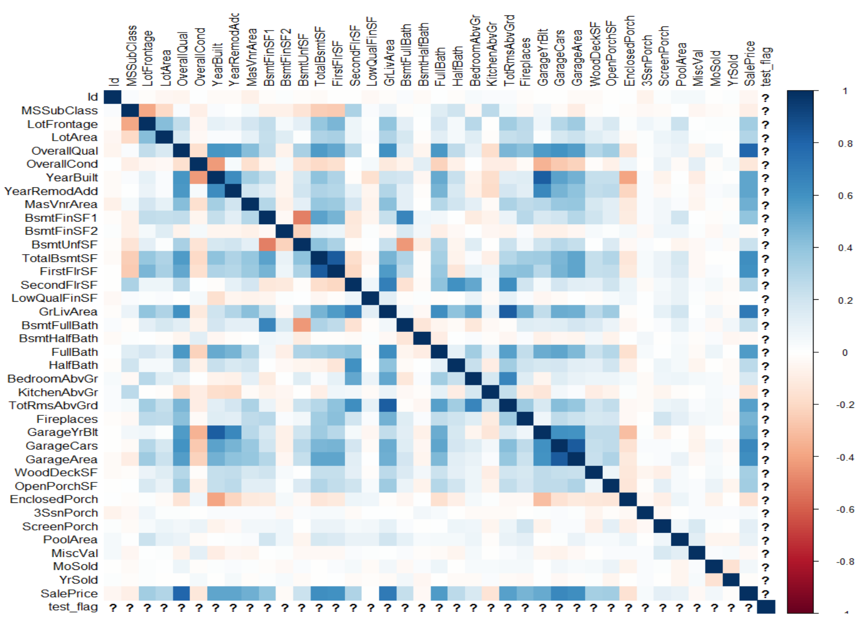


Below is the distribution of the few of the other continuous variables:



Correlation:

Below is the correlation of the sale price with the other variables:



2. MODELS:

I started off by building regression models with different features, built few of the models and few of them are documented here:

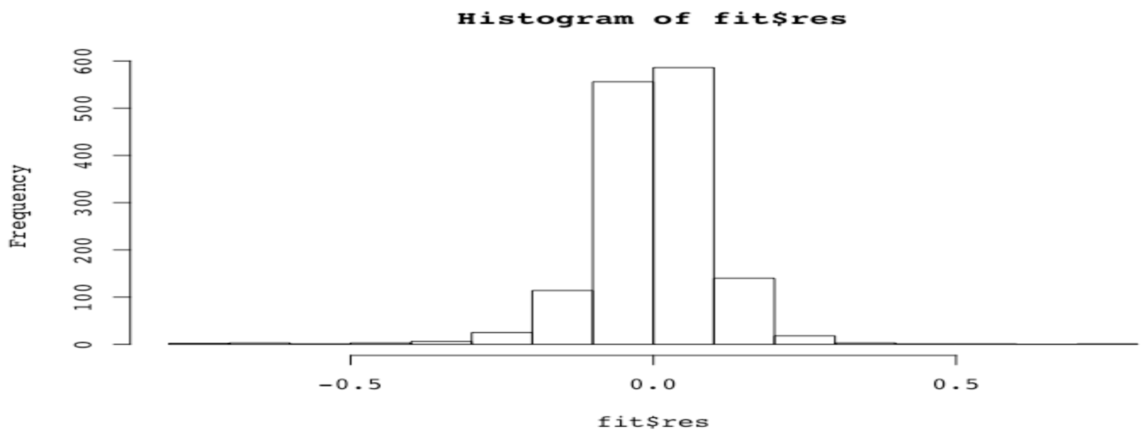
MULTI-REGRESSION MODEL:

```
Call:
lm(formula = log(finalset$SalePrice[1:1460]) ~ MSSubClass + MSZoning +
    LotArea + Street + LandContour + Utilities + LotConfig +
    LandSlope + Neighborhood + Condition1 + Condition2 + BldgType +
    OverallQual + OverallCond + YearBuilt + YearRemodAdd + RoofStyle +
    RoofMatl + Exterior1st + MasVnrType + MasVnrArea + ExterQual +
    BsmtQual + BsmtCond + BsmtExposure + BsmtFinType1 + BsmtFinSF1 +
    BsmtFinSF2 + BsmtUnfSF + X1stFlrSF + X2ndFlrSF + FullBath +
    BedroomAbvGr + KitchenAbvGr + KitchenQual + TotRmsAbvGrd +
    Functional + Fireplaces + GarageCars + GarageArea + GarageQual +
    ScreenPorch + PoolArea + SaleCondition, data = finalset[1:1460,
    ])

Residuals:
     Min       1Q   Median       3Q      Max
-0.73267 -0.04687  0.00205  0.05793  0.73267
```

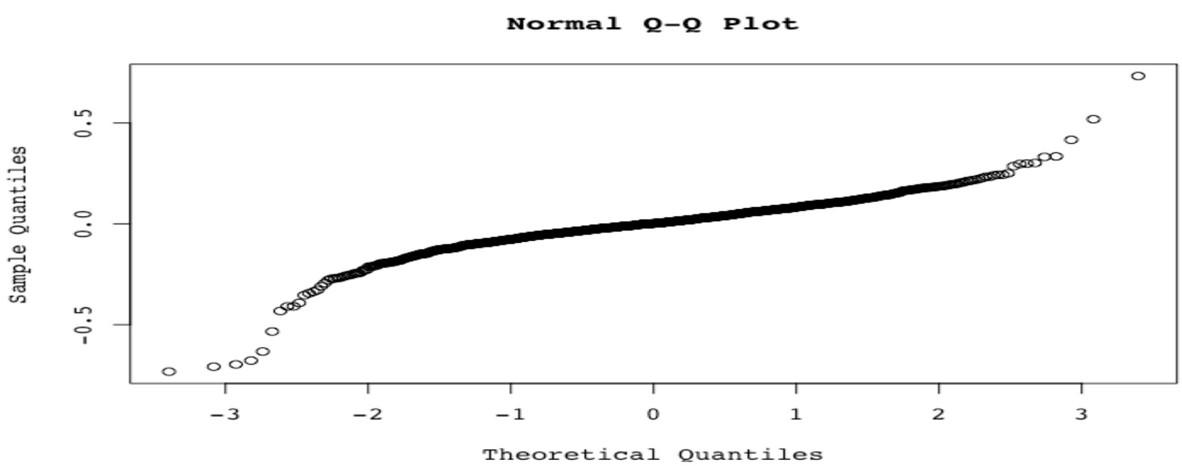
```
Residual standard error: 0.1088 on 1316 degrees of freedom
Multiple R-squared:  0.9331,	Adjusted R-squared:  0.9258
F-statistic: 128.3 on 143 and 1316 DF,  p-value: < 2.2e-16
```

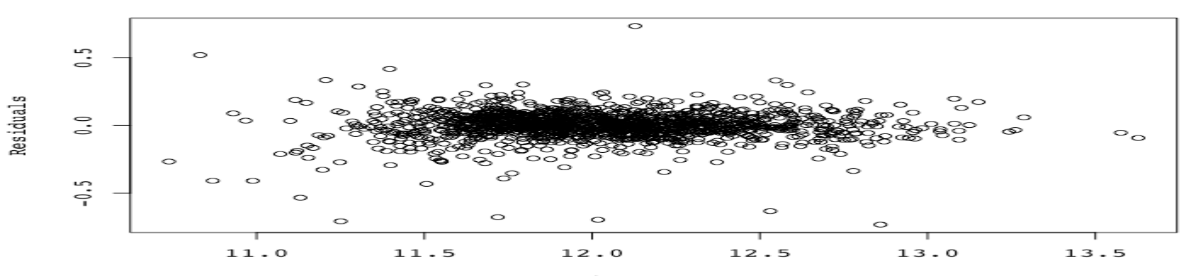
Residual plots:

The histogram of the residual looks normal.



The qq plot looks linear, could see few outliers but looks pretty ok.





The above graph shows a plot of the residuals against the fitted values for the saleprice model.

Predicted sample output:

	A	B
1	Id	Saleprice
2	1461	122900.16
3	1462	153223.966
4	1463	182132.142
5	1464	193401.075
6	1465	196422.678
7	1466	171562.942
8	1467	177822.008
9	1468	167168.228
10	1469	189569.917
11	1470	119853.58
12	1471	182437.022
13	1472	96320.146
14	1473	97280.9483
15	1474	143779.147
16	1475	113884.377
17	1476	366969.202
18	1477	255226.581
19	1478	293691.491

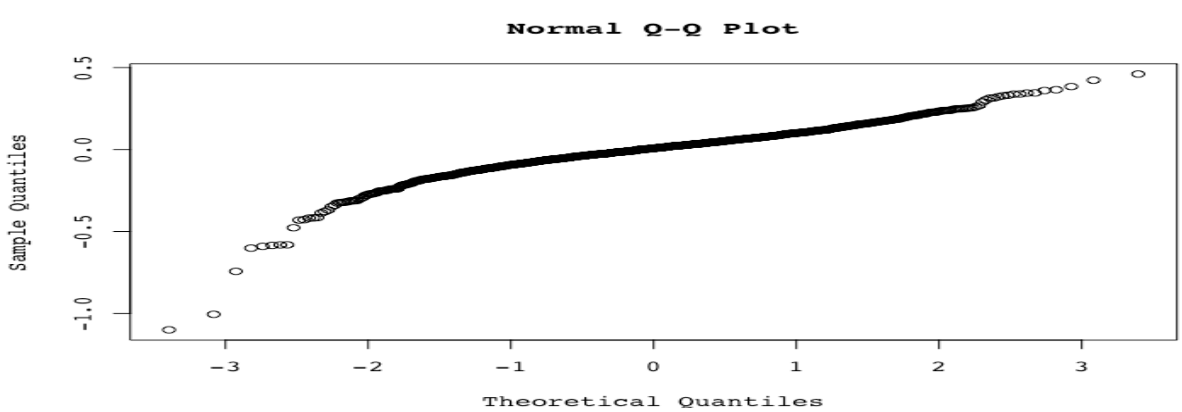
2.1 Model 2- Regression:

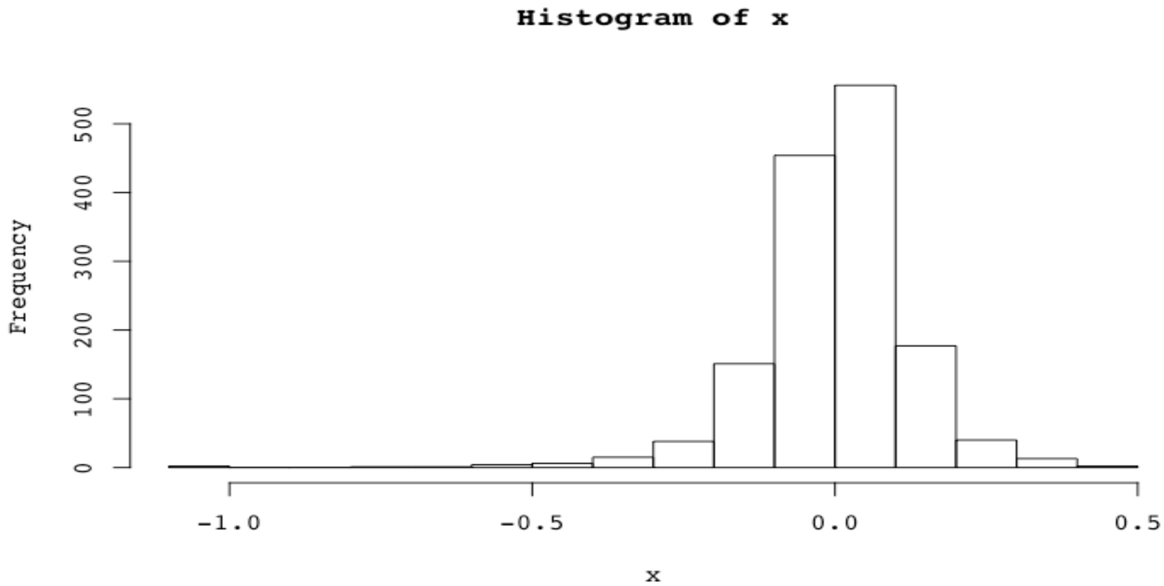
Linear model 2 with few transformed data:

```
> fit <- lm(log(SalePrice) ~ sqrt(MasVnrArea) + log(LotArea) + sqrt(TotalBsmtSF) + sqrt(X1stFlrS
F) + log(GrLivArea) + sqrt(GarageArea) + YrSold + MoSold + sqrt(OverallQual) + YearBuilt + YearR
emodAdd + GarageCars + TotRmsAbvGrd + Neighborhood + Fireplaces + MSZoning + Street + X2ndFlrSF
+ FullBath + BedroomAbvGr + KitchenAbvGr + GarageQual + ScreenPorch + PoolArea + SaleCondition +
BsmtExposure + BsmtFinType1 + BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF , data=totalimp[1:1460,])
> summary(fit)

Call:
lm(formula = log(SalePrice) ~ sqrt(MasVnrArea) + log(LotArea) +
    sqrt(TotalBsmtSF) + sqrt(X1stFlrSF) + log(GrLivArea) + sqrt(GarageArea) +
    YrSold + MoSold + sqrt(OverallQual) + YearBuilt + YearRemodAdd +
    GarageCars + TotRmsAbvGrd + Neighborhood + Fireplaces + MSZoning +
    Street + X2ndFlrSF + FullBath + BedroomAbvGr + KitchenAbvGr +
    GarageQual + ScreenPorch + PoolArea + SaleCondition + BsmtExposure +
    BsmtFinType1 + BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF, data = totalimp[1:1460,
    ])

Residuals:
     Min       1Q   Median       3Q      Max
-1.09965 -0.05712  0.00723  0.06772  0.46034
```

```
Residual standard error: 0.1286 on 1377 degrees of freedom
Multiple R-squared:  0.9022,    Adjusted R-squared:  0.8964
F-statistic: 154.9 on 82 and 1377 DF,  p-value: < 2.2e-16
```

Residual plots:



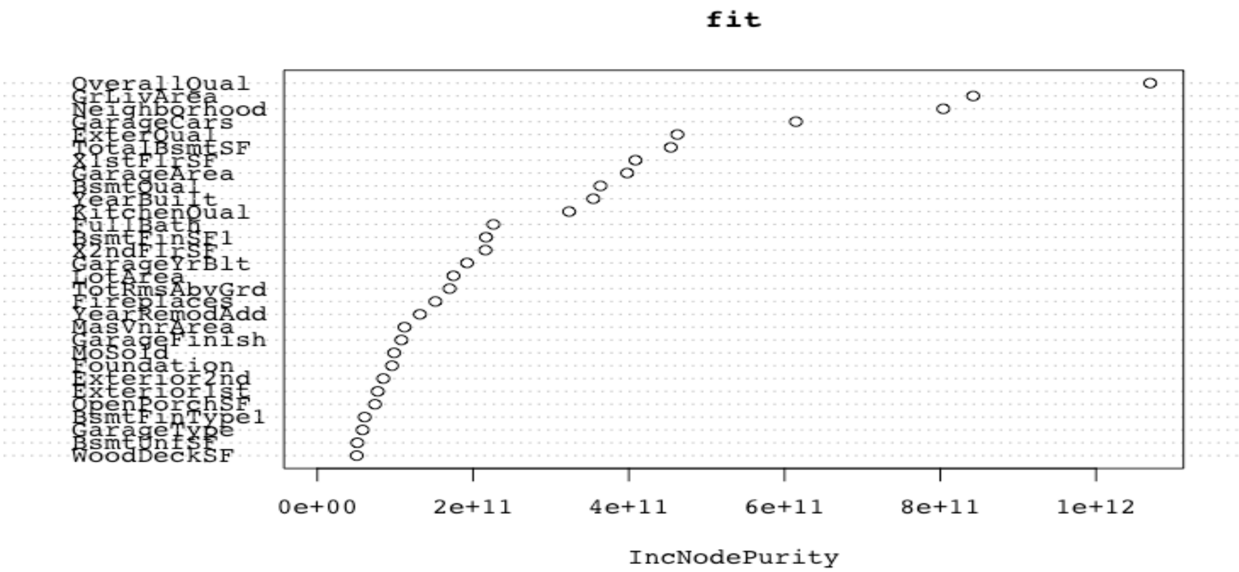


By looking at the adjusted r square values and the residual plots the model looks good.

2.2 MODEL3:
RANDOM FOREST:

```
> foresubmit<-predict(fit,finalset[1461:2919,])
> fit<-randomForest(finalset$SalePrice[1:1460]~., data=finalset[1:1460,], mtry=9, ntree=500)
> foresubmit<-predict(fit,finalset[1461:2919,])
> write.csv(foresubmit,"forestmodel.csv")
```

Important variables as per the model



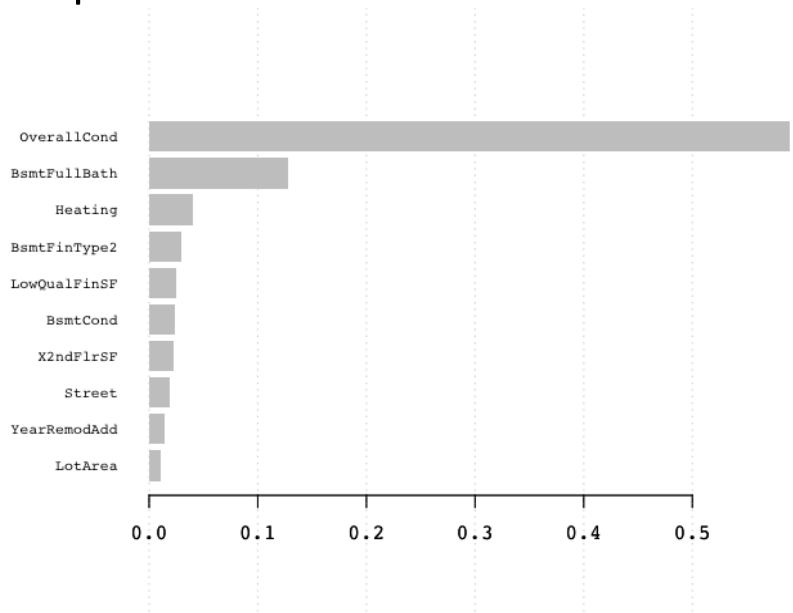
predicted output:

A	B
Id	Saleprice
1461	129298.631
1462	156058.263
1463	180918.44
1464	188348.151
1465	197674.197
1466	184155.992
1467	175108.809
1468	177327.064
1469	181084.432
1470	129420.757
1471	193004.674
1472	100207.362
1473	103074.144
1474	152466.674
1475	129905.911
1476	366619.501
1477	263797.629

2.3 MODEL4:
GRADIENT BOOST:

```
#Cross validate the model

cv.sparse <- xgb.cv(data = train, nrounds = 600,
                    min_child_weight = 0, max_depth = 10,
                    eta = 0.02, subsample = .7, colsample_bytree = .7,
                    booster = "gbtree", eval_metric = "rmse",
                    verbose = TRUE, print_every_n = 50,
                    nfold = 4, nthread = 2, objective="reg:linear")

#Train the model #Choose the parameters for the model
                    param <- list(colsample_bytree = .7,
                    subsample = .7, booster = "gbtree",
                    max_depth = 10, eta = 0.02,
                    eval_metric = "rmse",
                    objective="reg:linear")

#Train the model using those parameters
bstSparse <- xgb.train(params = param, data = train,
                    nrounds = 600, watchlist = list(train = train),
                    verbose = TRUE, print_every_n = 50, nthread = 2)
```

Variable importance in the model



3. PIER REVIEW

Case 1:[4]

I came across this paper where they built multivariate regression models of home prices using a dataset composed of 81 homes. They applied the maximum information coefficient (MIC) statistics to the observed home values (Y) and the predicted values (X) as an evaluation of the regression models. The results showed very high strength of the relationship between the two variables X and Y.

Table 1 Attributes of a House

Variable			Description
Dependent		*Value*	Assessed home value
Predictor	first-order	*Acreage*	Area of lot in acres
		Stories	Number of stories
		Area	Area in square footage
		Exterior	Exterior condition, 1 = good / excellent, 0 = average / below
		NatGes	1 = natural gas heating system, 0 = other heating system
		Rooms	Total number of rooms
		Bedrooms	Number of bedrooms
		FullBath	Number of full bathrooms
		HalfBath	Number of half bathrooms
		Fireplace	1 = with, 0 = without
		Garage	1 = with, 0 = without
	second-order	*Area**2*	House area squared
		*Acreage**2*	Lot size squared
		*Stories**2*	Number of stories squared
		*Rooms**2*	Number of rooms squared

Below is the regression model built:

Based on the Best Subsets analysis results, three regression models were built for the three selected cases:

$$\hat{V}_1 = -104582 + 45216\,Acreage + 36542\,Stories + 67.4\,Area + 12242\,FullBath + 16428\,HalfBath + 30480\,Garage - 4397\,Acreage^2 \qquad (M\text{-}1)$$

$$\hat{V}_2 = -101097 + 21512\,Acreage + 38141\,Stories + 71.2\,Area + 18580\,Exterior + 12218\,FullBath + 14569\,HalfBath + 23999\,Garage \qquad (M\text{-}2)$$

$$\hat{V}_3 = -111721 + 42939\,Acreage + 38965\,Stories + 72.3\,Area + 18901\,Exterior - 6781\,Rooms + 12139\,Bedrooms + 9721\,FullBath + 21047\,HalfBath + 24095\,Garage - 3919\,Acreage^2 \qquad (M\text{-}3)$$

Notice that the third model (M-3) has fewer variables than as indicated in the row `Vars=14` of Table 2. This is because several non-significant indicators were removed (in the order of first removing least significant and second-order indicators).

CASE2: [5]

This article describes a complete multiple linear regression analysis of home price data for a city in Oregon, USA in 2005.The article discusses statistical ideas ranging from those suitable for the regression component of a second college statistics course to those typically found in more advanced linear regression courses. The analysis includes many elements covered in typical regression components of second statistics courses such as indicator variables for coding qualitative information, model building, hypothesis testing, diagnostics, and model interpretation. The analysis also provides a compelling application of more challenging topics including predictor interactions, predictor transformations, and understanding model results through the use of graphics.

- *Size* = floor size (thousands of square feet)
- *Lot* = lot size category (from 1 to 11—explained below)
- *Bath* = number of bathrooms (with half-bathrooms counting as 0.1—explained below)
- *Bed* = number of bedrooms (between 2 and 6)
- *Age* = age (standardized: (year built - 1970)=10—explained below)
- *Garage* = garage size (0, 1, 2, or 3 cars)
- *Active* = indicator for "active listing" (reference: pending or sold)
- *Edison* = indicator for Edison Elementary (reference: Edgewood Elementary)
- *Harris* = indicator for Harris Elementary (reference: Edgewood Elementary)
- *Adams* = indicator for Adams Elementary (reference: Edgewood Elementary)
- *Crest* = indicator for Crest Elementary (reference: Edgewood Elementary)
- *Parker* = indicator for Parker Elementary (reference: Edgewood Elementary)

Below were the models built in this article:

E(Price) = b0 + b1Size + b2Lot + b3Bath + b4Bed + b5Age + b6Garage + b7Active + b8Edison + b9Harris + b10Adams + b11Crest + b12Parker.

E(Price) = b0 + b1Size + b2Lot + b3Bath + b4Bed + b34BedBath + b5Age + b52Age2 + b6Garage + b7Active + b8Edison + b9Harris + b10Adams + b11Crest + b12Parker.

MODEL interpretation:

A potential use for the final model might be to narrow the range of possible values for the asking price of a home about to be put on the market. For example, consider a home with the following features: 1879 square feet, lot size category 4, two and a half bathrooms, three bedrooms, built in 1975, two-car garage, and near Parker Elementary School (this was my home at the time). A 95% prediction interval ignoring the model comes to ($164,800, $406,800); this is based on the formula: sample mean ± t-percentile × sample standard deviation × By contrast, a 95% prediction interval using the model results comes to ($197,100, $369,000), which is about 70% the width of the interval ignoring the model. A realtor could advise the vendors to price their home somewhere within this range depending on other factors not included in the model (e.g., toward the upper end of this range if the home is on a nice street, the property is in good condition, and landscaping has been done to the yard). As is often the case, the regression analysis results are more effective when applied in the context of expert opinion and experience.

CASE3:

The algorithm The random forests algorithm (for both classification and regression) is as follows: 1. Draw ntree bootstrap samples from the original data. 2. For each of the bootstrap samples, grow an unpruned classification or regression tree, with the following modification: at each node, rather than choosing the best split among all predictors, randomly sample mtry of the predictors and choose the best split from among those variables. (Bagging can be thought of as the special case of random forests obtained when mtry = p, the number of predictors.) 3. Predict new data by aggregating the predictions of the ntree trees (i.e., majority votes for classification, average for regression). An estimate of the error rate can be obtained, based on the training data, by the following: 1. At each bootstrap iteration, predict the data not in the bootstrap sample (what Breiman calls "out-of-bag", or OOB, data) using the tree grown with the bootstrap sample. 2. Aggregate the OOB predictions. (On the average, each data point would be out-of-bag around 36% of the times, so aggregate these predictions.) Calcuate the error rate, and call it the OOB estimate of error rate.

A regression example:

We use the Boston Housing data (available in the MASS package) as an example for regression by random forest. Note a few differences between classifi- cation and regression random forests:

• The default mtry is p/3, as opposed to p 1/2 for classification, where p is the number of predictors. •

The default nodesize is 5, as opposed to 1 for classification. (In the tree building algorithm, nodes with fewer than nodesize observations are not splitted.) • There is only one measure of variable importance, instead of four.

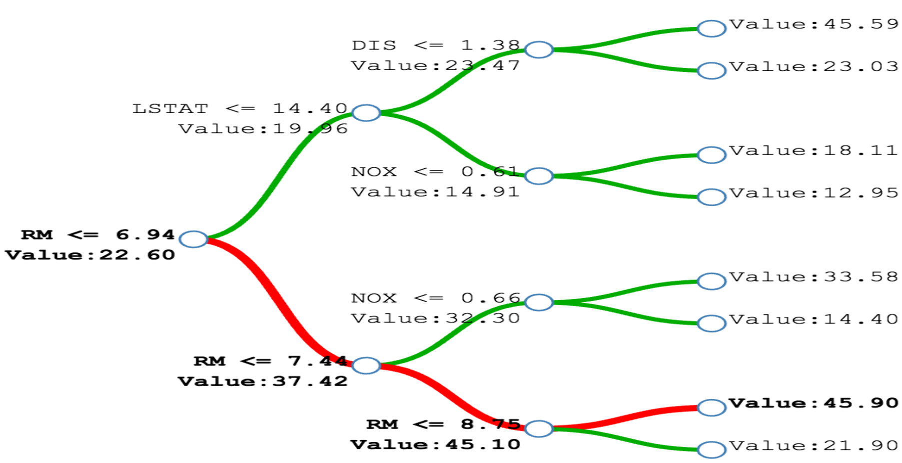
```
> data(Boston)
> set.seed(1341)
> BH.rf <- randomForest(medv ~ ., Boston)
> print(BH.rf)
Call:
 randomForest.formula(formula = medv ~ .,
     data = Boston)
               Type of random forest: regression
                     Number of trees: 500
```

No. of variables tried at each split: 4 Mean of squared residuals: 10.64615 % Var explained: 87.39

CASE4:[6]

Boston housing data

In this case the author has taken the Boston housing price data, which includes housing prices in suburbs of Boston together with a number of key attributes such as air quality (NOX variable below), distance from the city center (DIST) and a number of others. Author has built a regression decision tree (of depth 3 to keep things readable) to predict housing prices. As usual, the tree has conditions on each internal node and a value associated with each leaf (i.e. the value to be predicted). But additionally we've plotted out the value at each internal node i.e. the mean of the response variables in that region.

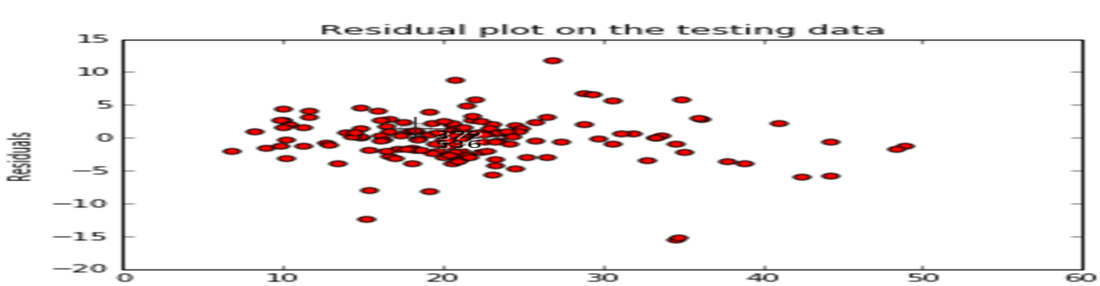


CASE 5:[7]

In this case the author used the xgboost for the Boston housing dataset.

Below is the code used for the training the model and implementing the same

```
from sklearn import cross_validation
from sklearn.datasets import load_boston
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.metrics import mean_squared_error,mean_absolute_err
import matplotlib.pyplot as plt

#Load boston housing dataset as an example
boston = load_boston()
bos_dat = boston["data"]
bos_tgt = boston["target"]
names = boston["feature_names"]

#Split data between training(70%) and test data(30%).
X_train, X_test, y_train, y_test = cross_validation.train_test_s

# initialize model with default parameters
xgb_model = GradientBoostingRegressor()

# train model using training data
xgb_model.fit(X_train, y_train)

# using model predict test data with features.
y_test_pred = xgb_model.predict(X_test)

# calculate error based on expected value y value(y_test) from p
print("explained variance score is", explained_variance_score(y_
print("mean square error is",mean_squared_error(y_test,y_test_pr
plt.scatter(y_test_pred, (y_test_pred - y_test), c='r', s=30)
plt.title("Residual plot on the testing data") plt.ylabel("Resid
```



MODEL IMPLEMENTATION:

Model1:

```
# regression model

Model1 <- lm (log(finalset$SalePrice[1:1460]) ~ MSSubClass + MSZoning + LotArea + Street +
                        LandContour + Utilities + LotConfig + LandSlope + Neighborhood + Condition1 + Condition2 +
                        BldgType + OverallQual + OverallCond + YearBuilt + YearRemodAdd + RoofStyle + RoofMatl +
                        Exterior1st + MasVnrType + MasVnrArea + ExterQual + BsmtQual + BsmtCond + BsmtExposure +
                        BsmtFinType1 + BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF + X1stFlrSF + X2ndFlrSF + FullBath +
                        BedroomAbvGr + KitchenAbvGr + KitchenQual + TotRmsAbvGrd + Functional + Fireplaces +
                        GarageCars + GarageArea + GarageQual + ScreenPorch + PoolArea + SaleCondition, data = finalset[1:1460, ])

#prediction
mypred=predict(fit,total[1461:2919,])
mypred<-exp(mypred)

#writing the results in csv:
write.csv(mypred,"linearmodel.csv")
```

Model2:

```
#Random forest:

Model2 <- randomForest(totalimp$SalePrice[1:1460]~., data=totalimp[1:1460,], mtry=9, ntree=500)

#predict

forestsubmit <- predict(model2,finalset[1461:2919,])

write.csv(forestsubmit,"randomforest.csv")
```

Model3:

```
#xgboost

sparse<- xgb.cv(data = trainD, nrounds = 600, min_child_weight = 0, max_depth = 10, eta = 0.02, subsample = .7,
             colsample_bytree = .7, booster = "gbtree", eval_metric = "rmse", verbose = TRUE,
             print_every_n = 50, nfold = 4,
             nthread = 2, objective="reg:linear")

#choose parameters:

parameters <- list(colsample_bytree = .7, subsample = .7, booster = "gbtree", max_depth = 10, eta = 0.02,
                eval_metric = "rmse", objective="reg:linear")

#model
Model3 <- xgb.train(params = parameters, data = trainData, nrounds = 600,
          watchlist = list(train = trainData), verbose = TRUE, print_every_n = 50, nthread = 2)

#predict:

testData <- xgb.DMatrix(data = test2[,vars])
prediction <- predict(Model3, testData)
test <- as.data.frame(as.matrix(test2))
prediction <- as.data.frame(as.matrix(prediction))
```

After implementation of the models the best results were obtained from the linear model

After submission of the model I got the kaggle rmse value of 0.13 for the multi regression Model.

Below are the kaggle RMSE for each model:

Model	**Kaggle rmse**
Multi regression	0.13
Random Forest	0.15
Xgboost	0.14

4. Limitations and future work and Learning:

After verifying and validating the data with various test and residual plots for the first model I could see that there is scope of improvement in this model, as far as the other two models goes, I have lot of work to do on random forest and xgboost model to fine tune these models and choosing the correct pamaters for the models Also in the current models I did not concentrate much on the outliers, which would definitely be a part of my future work.

As part of this project I have learned a lot on feature engineering and using those features in Modelling process. Exploratory data analysis was a major part of learning in this project where EDA was the key to understand the data and to build the features, also I learnt various Algorithms while working on this project which would definitively help me going forward.

As part of the future work I would like to do some research on the housing data And try to understand the various aspects which affects the housing market. Definitely work on

The outliers in the data and try out various new algorithms which I have not used up till now.

I have built around 20 odd models for this competition and since this competition would go for

Another 3 months I would definitely like to be a part of it and keep submitting my improved

And hopefully be on the top 1 % of the leader board.

References:

1. Anselin, L., Lozano-Gracia, N.: Errors in variables and spatial effects in hedonic house price models of ambient air quality. Tech. Rep. Working Paper 2007–1, University of Illinois, Urbana-Champaign (2007)
2. Bitter, C., Mulligan, G.F., Dall'erba, S.: Incorporating spatial variation in housing attribute prices: A comparison of geographically weighted reg
9. Lentz, G.H., Wang, K.: Residential appraisal and the lending process: A survey of issues. Journal of Real Estate Research 15(1-2), 11–40 (1998)
4.http://www.acisinternational.org/Springer/SamplePaper.pdf
5.Journal of Statistics Education Volume 16, Number 2 (2008), www.amstat.org/publications/jse/v16n2/pardoe.html
6. http://www.bios.unc.edu/~dzeng/BIOS740/randomforest.pdf
L. Breiman. Bagging predictors. Machine Learning, 24 (2):123–140, 1996. 18 L. Breiman. Random forests. Machine Learning, 45(1): 5–32, 2001. 18
7. https://flymont.io/tag/xgboost/
8. https://edumine.wordpress.com/2016/09/06/sold-how-do-home-features-add-up-to-its-price-tag/
9. https://www.google.com/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#q=exploratory+analysis+on+housing+data+in+kaggle+ames