REPORT

NAME: Abhishak Varshney

EMAIL: abhishakvarshey@gmail.com

COLLEGE: NIT Jaipur (MNIT)

#MOBILE: +91-8433489919

PREDICT HOUSE PRICES FOR TEST-SET

Introduction

1…..Load Package

2…..Read Train and Test data

3…..Combine Train and Test dataset

4…..Data Visualization

5…..Missing Value

6…..Removing Skewed Variables

7…..Build the model

8…..Variable importance

9…..Final Prediction

10….Calculate RMSE (Root Mean Squared Error)

Introduction

This dataset contains house sale prices. I'll use Random Forest to create a model predicting House prices using attributes given in dataset. The housing train data set has 1460 rows and 81 features. To start, I will hypothesize the following subset of the variables as potential predicators.

- salePrice - the property's sale price in dollars. This is the target variable that I am trying to predict.
- OverallCond - Overall condition rating
- YearBuilt - Original construction date
- YearRemodAdd - Remodel data
- BedroomAbvGr - Number of bedrooms above basement level
- GrLivArea - Above grade (ground) living area square feet

- KitchenAbvGr - Number of kitchens above grade
- TotRmsAbvGrd - Total rooms above grade (does not include bathrooms)
- GarageCars - Size of garage in car capacity
- PoolArea - Pool area in square feet
- LotArea - Lot size in square feet

Load Package

Packages include: ggplot2, ggthemes, scales, dplyr, randomForest, data.table, gridExtra, corrplot, GGally, e1071.

```
> ##Load Packages:
> library('ggplot2')
> library('ggthemes')
> library('scales')
> library('dplyr')

Attaching package: ‘dplyr’

The following objects are masked from ‘package:stats’:

    filter, lag

The following objects are masked from ‘package:base’:

    intersect, setdiff, setequal, union

> library('randomForest')
randomForest 4.6-14
Type rfNews() to see new features/changes/bug fixes.

Attaching package: ‘randomForest’

The following object is masked from ‘package:dplyr’:

    combine

The following object is masked from ‘package:ggplot2’:

    margin

> library('data.table')
data.table 1.11.2
  The fastest way to learn (by data.table authors): https://www.datacamp.com/courses/data-
analysis-the-data-table-way
  Documentation: ?data.table, example(data.table) and browseVignettes("data.table")
  Release notes, videos and slides: http://r-datatable.com

Attaching package: ‘data.table’

The following objects are masked from ‘package:dplyr’:

    between, first, last

> library('gridExtra')

Attaching package: ‘gridExtra’

The following object is masked from ‘package:randomForest’:

    combine

The following object is masked from ‘package:dplyr’:
```

```
    combine

> library('corrplot')
corrplot 0.84 loaded
> library('GGally')

Attaching package: ‘GGally’

The following object is masked from ‘package:dplyr’:

    nasa

> library('e1071')
```

Read Train and Test data

Now we read both the files train.csv and test.csv.

```
> #Read Train & Test Data
> train <-read.csv('C:\\Users\\abhis\\Desktop\\A\\train.csv', stringsAsFactors = F)
> test  <-read.csv('C:\\Users\\abhis\\Desktop\\A\\test.csv', stringsAsFactors = F)
```

The housing train data set has 1460 rows and 81 features with the target feature Sale Price. The housing test data set has 1459 rows and 80 features with the target feature Sale Price.

```
#Structure of the data
> dim(train)
[1] 1460   81
> str(train)
'data.frame':  1460 obs. of  81 variables:
 $ Id           : int  1 2 3 4 5 6 7 8 9 10 ...
 $ MSSubClass   : int  60 20 60 70 60 50 20 60 50 190 ...
 $ ScreenPorch  : int  0 0 0 0 0 0 0 0 0 0 ...
 $ PoolArea     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ PoolQC       : chr  NA NA NA NA ...
 $ Fence        : chr  NA NA NA NA ...
 $ MiscFeature  : chr  NA NA NA NA ...
 $ MiscVal      : int  0 0 0 0 0 700 0 350 0 0 ...
 $ GarageType   : chr  "Attchd" "Attchd" "Attchd" "Detchd" ...
 $ GarageYrBlt  : int  2003 1976 2001 1998 2000 1993 2004 1973 1931 1939 ...
 $ GarageFinish : chr  "RFn" "RFn" "RFn" "Unf" ...
 $ GarageCars   : int  2 2 2 3 3 2 2 2 2 1 ...
 $ GarageArea   : int  548 460 608 642 836 480 636 484 468 205 ...
 $ GarageQual   : chr  "TA" "TA" "TA" "TA" ...
 $ GarageCond   : chr  "TA" "TA" "TA" "TA" ...
 $ PavedDrive   : chr  "Y" "Y" "Y" "Y" ...
 $ WoodDeckSF   : int  0 298 0 0 192 40 255 235 90 0 ...
 $ OpenPorchSF  : int  61 0 42 35 84 30 57 204 0 4 ...
 $ EnclosedPorch: int  0 0 0 272 0 0 0 228 205 0 ...
 $ X3SsnPorch   : int  0 0 0 0 0 320 0 0 0 0 ...
 $ BsmtFullBath : int  1 0 1 1 1 1 1 1 0 1 ...
 $ BsmtHalfBath : int  0 1 0 0 0 0 0 0 0 0 ...
 $ FullBath     : int  2 2 2 1 2 1 2 2 2 1 ...
 $ HalfBath     : int  1 0 1 0 1 1 0 1 0 0 ...
 $ BedroomAbvGr : int  3 3 3 3 4 1 3 3 2 2 ...
 $ KitchenAbvGr : int  1 1 1 1 1 1 1 1 2 2 ...
 $ KitchenQual  : chr  "Gd" "TA" "Gd" "Gd" ...
 $ TotRmsAbvGrd : int  8 6 6 7 9 5 7 7 8 5 ...
 $ Functional   : chr  "Typ" "Typ" "Typ" "Typ" ...
 $ Fireplaces   : int  0 1 1 1 1 0 1 2 2 2 ...
 $ FireplaceQu  : chr  NA "TA" "TA" "Gd" ...
 $ Heating      : chr  "GasA" "GasA" "GasA" "GasA" ...
 $ HeatingQC    : chr  "Ex" "Ex" "Ex" "Gd" ...
 $ CentralAir   : chr  "Y" "Y" "Y" "Y" ...
 $ Electrical   : chr  "SBrkr" "SBrkr" "SBrkr" "SBrkr" ...
 $ X1stFlrSF    : int  856 1262 920 961 1145 796 1694 1107 1022 1077 ...
 $ X2ndFlrSF    : int  854 0 866 756 1053 566 0 983 752 0 ...
 $ LowQualFinSF : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
 $ GrLivArea    : int  1710 1262 1786 1717 2198 1362 1694 2090 1774 1077 ...
 $ BsmtQual     : chr  "Gd" "Gd" "Gd" "TA" ...
 $ BsmtCond     : chr  "TA" "TA" "TA" "Gd" ...
 $ BsmtExposure : chr  "No" "Gd" "Mn" "No" ...
 $ BsmtFinType1 : chr  "GLQ" "ALQ" "GLQ" "ALQ" ...
 $ BsmtFinSF1   : int  706 978 486 216 655 732 1369 859 0 851 ...
 $ BsmtFinType2 : chr  "Unf" "Unf" "Unf" "Unf" ...
 $ BsmtFinSF2   : int  0 0 0 0 0 0 0 32 0 0 ...
 $ BsmtUnfSF    : int  150 284 434 540 490 64 317 216 952 140 ...
 $ TotalBsmtSF  : int  856 1262 920 756 1145 796 1686 1107 952 991 ...
 $ Exterior2nd  : chr  "VinylSd" "MetalSd" "VinylSd" "Wd Shng" ...
 $ MasVnrType   : chr  "BrkFace" "None" "BrkFace" "None" ...
 $ MasVnrArea   : int  196 0 162 0 350 0 186 240 0 0 ...
 $ ExterQual    : chr  "Gd" "TA" "Gd" "TA" ...
 $ ExterCond    : chr  "TA" "TA" "TA" "TA" ...
 $ Foundation   : chr  "PConc" "CBlock" "PConc" "BrkTil" ...
 $ YearBuilt    : int  2003 1976 2001 1915 2000 1993 2004 1973 1931 1939 ...
 $ YearRemodAdd : int  2003 1976 2002 1970 2000 1995 2005 1973 1950 1950 ...
 $ RoofStyle    : chr  "Gable" "Gable" "Gable" "Gable" ...
 $ RoofMatl     : chr  "CompShg" "CompShg" "CompShg" "CompShg" ...
 $ Exterior1st  : chr  "VinylSd" "MetalSd" "VinylSd" "Wd Sdng" ...
 $ BldgType     : chr  "1Fam" "1Fam" "1Fam" "1Fam" ...
 $ HouseStyle   : chr  "2Story" "1Story" "2Story" "2Story" ...
 $ OverallQual  : int  7 6 7 7 8 5 8 7 7 5 ...
 $ OverallCond  : int  5 8 5 5 5 5 5 6 5 6 ...
 $ Neighborhood : chr  "CollgCr" "Veenker" "CollgCr" "Crawfor" ...
 $ Condition1   : chr  "Norm" "Feedr" "Norm" "Norm" ...
 $ Condition2   : chr  "Norm" "Norm" "Norm" "Norm" ...
 $ LandContour  : chr  "Lvl" "Lvl" "Lvl" "Lvl" ...
 $ Utilities    : chr  "AllPub" "AllPub" "AllPub" "AllPub" ...
 $ LotConfig    : chr  "Inside" "FR2" "Inside" "Corner" ...
 $ LandSlope    : chr  "Gtl" "Gtl" "Gtl" "Gtl" ...
 $ LotArea      : int  8450 9600 11250 9550 14260 14115 10084 10382 6120 7420 ...
 $ Street       : chr  "Pave" "Pave" "Pave" "Pave" ...
 $ Alley        : chr  NA NA NA NA ...
 $ LotShape     : chr  "Reg" "Reg" "IR1" "IR1" ...
 $ MSZoning     : chr  "RL" "RL" "RL" "RL" ...
 $ LotFrontage  : int  65 80 68 60 84 85 75 NA 51 50 ...
 $ MoSold       : int  2 5 9 2 12 10 8 11 4 1 ...
 $ YrSold       : int  2008 2007 2008 2006 2008 2009 2007 2009 2008 2008 ...
 $ SaleType     : chr  "WD" "WD" "WD" "WD" ...
 $ SaleCondition: chr  "Normal" "Normal" "Normal" "Abnorml" ...
 $ SalePrice    : int  208500 181500 223500 140000 250000 143000 307000 200000 129
900 118000 ...
>
> dim(test)
[1] 1459   80
> str(test)
'data.frame':  1459 obs. of  80 variables:
 $ Id           : int  1461 1462 1463 1464 1465 1466 1467 1468 1469 1470 ...
 $ MSSubClass   : int  20 20 60 60 120 60 20 60 20 20 ...
 $ ScreenPorch  : int  120 0 0 0 144 0 0 0 0 0 ...
 $ PoolArea     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ PoolQC       : chr  NA NA NA NA ...
 $ Fence        : chr  "MnPrv" NA "MnPrv" NA ...
 $ MiscFeature  : chr  NA "Gar2" NA NA ...
 $ MiscVal      : int  0 12500 0 0 0 0 500 0 0 0 ...
 $ GarageType   : chr  "Attchd" "Attchd" "Attchd" "Attchd" ...
 $ GarageYrBlt  : int  1961 1958 1997 1998 1992 1993 1992 1998 1990 1970 ...
 $ GarageFinish : chr  "Unf" "Unf" "Fin" "Fin" ...
 $ GarageCars   : int  1 1 2 2 2 2 2 2 2 2 ...
 $ GarageArea   : int  730 312 482 470 506 440 420 393 506 525 ...
 $ GarageQual   : chr  "TA" "TA" "TA" "TA" ...
 $ GarageCond   : chr  "TA" "TA" "TA" "TA" ...
 $ PavedDrive   : chr  "Y" "Y" "Y" "Y" ...
 $ WoodDeckSF   : int  140 393 212 360 0 157 483 0 192 240 ...
 $ OpenPorchSF  : int  0 36 34 36 82 84 21 75 0 0 ...
 $ EnclosedPorch: int  0 0 0 0 0 0 0 0 0 0 ...
 $ X3SsnPorch   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ BsmtFullBath : int  0 0 0 0 0 0 1 0 1 1 ...
 $ BsmtHalfBath : int  0 0 0 0 0 0 0 0 0 0 ...
 $ FullBath     : int  1 1 2 2 2 2 2 2 1 1 ...
 $ HalfBath     : int  0 1 1 1 0 1 0 1 1 0 ...
 $ BedroomAbvGr : int  2 3 3 3 2 3 3 3 2 2 ...
 $ KitchenAbvGr : int  1 1 1 1 1 1 1 1 1 1 ...
 $ KitchenQual  : chr  "TA" "Gd" "TA" "Gd" ...
 $ TotRmsAbvGrd : int  5 6 6 7 5 7 6 7 5 4 ...
```

```
 $ Functional   : chr  "Typ" "Typ" "Typ" "Typ" ...
 $ Fireplaces   : int  0 0 1 1 0 1 0 1 1 0 ...
 $ FireplaceQu  : chr  NA NA "TA" "Gd" ...
 $ Heating      : chr  "GasA" "GasA" "GasA" "GasA" ...
 $ HeatingQC    : chr  "TA" "TA" "Gd" "Ex" ...
 $ CentralAir   : chr  "Y" "Y" "Y" "Y" ...
 $ Electrical   : chr  "SBrkr" "SBrkr" "SBrkr" "SBrkr" ...
 $ X1stFlrSF    : int  896 1329 928 926 1280 763 1187 789 1341 882 ...
 $ X2ndFlrSF    : int  0 0 701 678 0 892 0 676 0 0 ...
 $ LowQualFinSF : int  0 0 0 0 0 0 0 0 0 0 ...
 $ GrLivArea    : int  896 1329 1629 1604 1280 1655 1187 1465 1341 882 ...
 $ BsmtQual     : chr  "TA" "TA" "Gd" "TA" ...
 $ BsmtCond     : chr  "TA" "TA" "TA" "TA" ...
 $ BsmtExposure : chr  "No" "No" "No" "No" ...
 $ BsmtFinType1 : chr  "Rec" "ALQ" "GLQ" "GLQ" ...
 $ BsmtFinSF1   : int  468 923 791 602 263 0 935 0 637 804 ...
 $ BsmtFinType2 : chr  "LwQ" "Unf" "Unf" "Unf" ...
 $ BsmtFinSF2   : int  144 0 0 0 0 0 0 0 0 78 ...
 $ BsmtUnfSF    : int  270 406 137 324 1017 763 233 789 663 0 ...
 $ TotalBsmtSF  : int  882 1329 928 926 1280 763 1168 789 1300 882 ...
 $ Exterior2nd  : chr  "VinylSd" "Wd Sdng" "VinylSd" "VinylSd" ...
 $ MasVnrType   : chr  "None" "BrkFace" "None" "BrkFace" ...
 $ MasVnrArea   : int  0 108 0 20 0 0 0 0 0 0 ...
 $ ExterQual    : chr  "TA" "TA" "TA" "TA" ...
 $ ExterCond    : chr  "TA" "TA" "TA" "TA" ...
 $ Foundation   : chr  "CBlock" "CBlock" "PConc" "PConc" ...
 $ YearBuilt    : int  1961 1958 1997 1998 1992 1993 1992 1998 1990 1970 ...
 $ YearRemodAdd : int  1961 1958 1998 1998 1992 1994 2007 1998 1990 1970 ...
 $ RoofStyle    : chr  "Gable" "Hip" "Gable" "Gable" ...
 $ RoofMatl     : chr  "CompShg" "CompShg" "CompShg" "CompShg" ...
 $ Exterior1st  : chr  "VinylSd" "Wd Sdng" "VinylSd" "VinylSd" ...
 $ BldgType     : chr  "1Fam" "1Fam" "1Fam" "1Fam" ...
 $ HouseStyle   : chr  "1Story" "1Story" "2Story" "2Story" ...
 $ OverallQual  : int  5 6 5 6 8 6 6 6 7 4 ...
 $ OverallCond  : int  6 6 5 6 5 5 7 5 5 5 ...
 $ Neighborhood : chr  "NAmes" "NAmes" "Gilbert" "Gilbert" ...
 $ Condition1   : chr  "Feedr" "Norm" "Norm" "Norm" ...
 $ Condition2   : chr  "Norm" "Norm" "Norm" "Norm" ...
 $ LandContour  : chr  "Lvl" "Lvl" "Lvl" "Lvl" ...
 $ Utilities    : chr  "AllPub" "AllPub" "AllPub" "AllPub" ...
 $ LotConfig    : chr  "Inside" "Corner" "Inside" "Inside" ...
 $ LandSlope    : chr  "Gtl" "Gtl" "Gtl" "Gtl" ...
 $ LotArea      : int  11622 14267 13830 9978 5005 10000 7980 8402 10176 8400 ...
 $ Street       : chr  "Pave" "Pave" "Pave" "Pave" ...
 $ Alley        : chr  NA NA NA NA ...
 $ LotShape     : chr  "Reg" "IR1" "IR1" "IR1" ...
 $ MSZoning     : chr  "RH" "RL" "RL" "RL" ...
 $ LotFrontage  : int  80 81 74 78 43 75 NA 63 85 70 ...
 $ MoSold       : int  6 6 3 6 1 4 3 5 2 4 ...
 $ YrSold       : int  2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...
 $ SaleType     : chr  "WD" "WD" "WD" "WD" ...
 $ SaleCondition: chr  "Normal" "Normal" "Normal" "Normal" ...
```

Count the number of columns that consist of text data and number of columns that c
onsist of numerical data seperately by using ‘sapply’ family of function.

```
> #Count the number of columns that consists of text data
> sum(sapply(train[,1:81], typeof) == "character")
[1] 43
>
> #Count the number of columns that consists of numerical data
> sum(sapply(train[,1:81], typeof) == "integer")
[1] 38
```

There are total 43 columns that consist of text data and 38 columns that consist
numerical data respectively.

```
> # Obtain summary statistics
> summary(train[,sapply(train[,1:81], typeof) == "integer"])

       Id           MSSubClass     ScreenPorch        PoolArea
 Min.   :   1.0   Min.   : 20.0   Min.   :  0.00   Min.   :  0.000
 1st Qu.: 365.8   1st Qu.: 20.0   1st Qu.:  0.00   1st Qu.:  0.000
 Median : 730.5   Median : 50.0   Median :  0.00   Median :  0.000
 Mean   : 730.5   Mean   : 56.9   Mean   : 15.06   Mean   :  2.759
```

```
3rd Qu.:1095.2   3rd Qu.: 70.0   3rd Qu.:  0.00   3rd Qu.:  0.000
Max.   :1460.0   Max.   :190.0   Max.   :480.00   Max.   :738.000

   MiscVal          GarageYrBlt     GarageCars      GarageArea
Min.   :    0.00   Min.   :1900   Min.   :0.000   Min.   :   0.0
1st Qu.:    0.00   1st Qu.:1961   1st Qu.:1.000   1st Qu.: 334.5
Median :    0.00   Median :1980   Median :2.000   Median : 480.0
Mean   :   43.49   Mean   :1979   Mean   :1.767   Mean   : 473.0
3rd Qu.:    0.00   3rd Qu.:2002   3rd Qu.:2.000   3rd Qu.: 576.0
Max.   :15500.00   Max.   :2010   Max.   :4.000   Max.   :1418.0
                   NA's   :81
  WoodDeckSF      OpenPorchSF     EnclosedPorch      X3SsnPorch
Min.   :  0.00   Min.   :  0.00   Min.   :  0.00   Min.   :  0.00
1st Qu.:  0.00   1st Qu.:  0.00   1st Qu.:  0.00   1st Qu.:  0.00
Median :  0.00   Median : 25.00   Median :  0.00   Median :  0.00
Mean   : 94.24   Mean   : 46.66   Mean   : 21.95   Mean   :  3.41
3rd Qu.:168.00   3rd Qu.: 68.00   3rd Qu.:  0.00   3rd Qu.:  0.00
Max.   :857.00   Max.   :547.00   Max.   :552.00   Max.   :508.00

 BsmtFullBath     BsmtHalfBath        FullBath        HalfBath
Min.   :0.0000   Min.   :0.00000   Min.   :0.000   Min.   :0.0000
1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:1.000   1st Qu.:0.0000
Median :0.0000   Median :0.00000   Median :2.000   Median :0.0000
Mean   :0.4253   Mean   :0.05753   Mean   :1.565   Mean   :0.3829
3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:2.000   3rd Qu.:1.0000
Max.   :3.0000   Max.   :2.00000   Max.   :3.000   Max.   :2.0000

 BedroomAbvGr    KitchenAbvGr    TotRmsAbvGrd      Fireplaces      X1stFlrSF
Min.   :0.000   Min.   :0.000   Min.   : 2.000   Min.   :0.000   Min.   : 334
1st Qu.:2.000   1st Qu.:1.000   1st Qu.: 5.000   1st Qu.:0.000   1st Qu.: 882
Median :3.000   Median :1.000   Median : 6.000   Median :1.000   Median :1087
Mean   :2.866   Mean   :1.047   Mean   : 6.518   Mean   :0.613   Mean   :1163
3rd Qu.:3.000   3rd Qu.:1.000   3rd Qu.: 7.000   3rd Qu.:1.000   3rd Qu.:1391
Max.   :8.000   Max.   :3.000   Max.   :14.000   Max.   :3.000   Max.   :4692

  X2ndFlrSF     LowQualFinSF       GrLivArea      BsmtFinSF1
Min.   :   0   Min.   :  0.000   Min.   : 334   Min.   :   0.0
1st Qu.:   0   1st Qu.:  0.000   1st Qu.:1130   1st Qu.:   0.0
Median :   0   Median :  0.000   Median :1464   Median : 383.5
Mean   : 347   Mean   :  5.845   Mean   :1515   Mean   : 443.6
3rd Qu.: 728   3rd Qu.:  0.000   3rd Qu.:1777   3rd Qu.: 712.2
Max.   :2065   Max.   :572.000   Max.   :5642   Max.   :5644.0

  BsmtFinSF2        BsmtUnfSF       TotalBsmtSF       MasVnrArea
Min.   :   0.00   Min.   :   0.0   Min.   :   0.0   Min.   :   0.0
1st Qu.:   0.00   1st Qu.: 223.0   1st Qu.: 795.8   1st Qu.:   0.0
Median :   0.00   Median : 477.5   Median : 991.5   Median :   0.0
Mean   :  46.55   Mean   : 567.2   Mean   :1057.4   Mean   : 103.7
3rd Qu.:   0.00   3rd Qu.: 808.0   3rd Qu.:1298.2   3rd Qu.: 166.0
Max.   :1474.00   Max.   :2336.0   Max.   :6110.0   Max.   :1600.0
                                                    NA's   :8
  YearBuilt     YearRemodAdd   OverallQual      OverallCond       LotArea
Min.   :1872   Min.   :1950   Min.   : 1.000   Min.   :1.000   Min.   :  1300
1st Qu.:1954   1st Qu.:1967   1st Qu.: 5.000   1st Qu.:5.000   1st Qu.:  7554
Median :1973   Median :1994   Median : 6.000   Median :5.000   Median :  9478
Mean   :1971   Mean   :1985   Mean   : 6.099   Mean   :5.575   Mean   : 10517
3rd Qu.:2000   3rd Qu.:2004   3rd Qu.: 7.000   3rd Qu.:6.000   3rd Qu.: 11602
Max.   :2010   Max.   :2010   Max.   :10.000   Max.   :9.000   Max.   :215245

 LotFrontage         MoSold           YrSold       SalePrice
Min.   : 21.00   Min.   : 1.000   Min.   :2006   Min.   : 34900
1st Qu.: 59.00   1st Qu.: 5.000   1st Qu.:2007   1st Qu.:129975
Median : 69.00   Median : 6.000   Median :2008   Median :163000
Mean   : 70.05   Mean   : 6.322   Mean   :2008   Mean   :180921
3rd Qu.: 80.00   3rd Qu.: 8.000   3rd Qu.:2009   3rd Qu.:214000
Max.   :313.00   Max.   :12.000   Max.   :2010   Max.   :755000
NA's   :259
```

The values of the variables can be printed using **print()** or **cat()**function. The **cat()** function combines multiple items into a continuous print output. Print the number of rows and columns in train.csv as well as in test.csv file.

```
>
> cat('Train has', dim(train)[1], 'rows and', dim(train)[2], 'columns.')
Train has 1460 rows and 81 columns.
> cat('Test has', dim(test)[1], 'rows and', dim(test)[2], ' columns.')
Test has 1459 rows and 80  columns.
```

Calculate the percentage of missing data in train.csv file.

```
>
> # The percentage of data missing in train
> sum(is.na(train)) / (nrow(train) *ncol(train))
[1] 0.05889565
```

Calculate the percentage of missing data in test.csv file.

```
>
> # The percentage of data missing in test
> sum(is.na(test)) / (nrow(test) * ncol(test))
[1] 0.05997258
```

Now check the number of duplicates rows in train.csv file.

```
>
> # Check for duplicated rows
> cat("The number of duplicated rows are", nrow(train) - nrow(unique(train)))
The number of duplicated rows are 0
```

We have 43 columns that consist of text and 38 columns are numerical. The text data could be challenging to work with. For those that are numerical, we looked at some descriptive statistics. There is no duplicate rows inside the data.

Combine data

```
> ##Combine data
> test$SalePrice<-rep(NA,1459)
> house<-bind_rows(train,test)
```

Since test dataset has no “Saleprice” variable. We will create it and then combine both the data-sets train as well as test in house data-set.

```
> ## Data Exploration
> str(house)
'data.frame':  2919 obs. of  81 variables:
 $ Id           : int  1 2 3 4 5 6 7 8 9 10 ...
 $ MSSubClass   : int  60 20 60 70 60 50 20 60 50 190 ...
 $ ScreenPorch  : int  0 0 0 0 0 0 0 0 0 0 ...
 $ PoolArea     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ PoolQC       : chr  NA NA NA NA ...
 $ Fence        : chr  NA NA NA NA ...
 $ MiscFeature  : chr  NA NA NA NA ...
 $ MiscVal      : int  0 0 0 0 0 700 0 350 0 0 ...
 $ GarageType   : chr  "Attchd" "Attchd" "Attchd" "Detchd" ...
 $ GarageYrBlt  : int  2003 1976 2001 1998 2000 1993 2004 1973 1931 1939 ...
 $ GarageFinish : chr  "RFn" "RFn" "RFn" "Unf" ...
 $ GarageCars   : int  2 2 2 3 3 2 2 2 2 1 ...
 $ GarageArea   : int  548 460 608 642 836 480 636 484 468 205 ...
 $ GarageQual   : chr  "TA" "TA" "TA" "TA" ...
 $ GarageCond   : chr  "TA" "TA" "TA" "TA" ...
 $ PavedDrive   : chr  "Y" "Y" "Y" "Y" ...
 $ WoodDeckSF   : int  0 298 0 0 192 40 255 235 90 0 ...
```

```
 $ OpenPorchSF  : int  61 0 42 35 84 30 57 204 0 4 ...
 $ EnclosedPorch: int  0 0 0 272 0 0 0 228 205 0 ...
 $ X3SsnPorch   : int  0 0 0 0 0 320 0 0 0 0 ...
 $ BsmtFullBath : int  1 0 1 1 1 1 1 1 0 1 ...
 $ BsmtHalfBath : int  0 1 0 0 0 0 0 0 0 0 ...
 $ FullBath     : int  2 2 2 1 2 1 2 2 2 1 ...
 $ HalfBath     : int  1 0 1 0 1 1 0 1 0 0 ...
 $ BedroomAbvGr : int  3 3 3 3 4 1 3 3 2 2 ...
 $ KitchenAbvGr : int  1 1 1 1 1 1 1 1 2 2 ...
 $ KitchenQual  : chr  "Gd" "TA" "Gd" "Gd" ...
 $ TotRmsAbvGrd : int  8 6 6 7 9 5 7 7 8 5 ...
 $ Functional   : chr  "Typ" "Typ" "Typ" "Typ" ...
 $ Fireplaces   : int  0 1 1 1 1 0 1 2 2 2 ...
 $ FireplaceQu  : chr  NA "TA" "TA" "Gd" ...
 $ Heating      : chr  "GasA" "GasA" "GasA" "GasA" ...
 $ HeatingQC    : chr  "Ex" "Ex" "Ex" "Gd" ...
 $ CentralAir   : chr  "Y" "Y" "Y" "Y" ...
 $ Electrical   : chr  "SBrkr" "SBrkr" "SBrkr" "SBrkr" ...
 $ X1stFlrSF    : int  856 1262 920 961 1145 796 1694 1107 1022 1077 ...
 $ X2ndFlrSF    : int  854 0 866 756 1053 566 0 983 752 0 ...
 $ LowQualFinSF : int  0 0 0 0 0 0 0 0 0 0 ...
 $ GrLivArea    : int  1710 1262 1786 1717 2198 1362 1694 2090 1774 1077 ...
 $ BsmtQual     : chr  "Gd" "Gd" "Gd" "TA" ...
 $ BsmtCond     : chr  "TA" "TA" "TA" "Gd" ...
 $ BsmtExposure : chr  "No" "Gd" "Mn" "No" ...
 $ BsmtFinType1 : chr  "GLQ" "ALQ" "GLQ" "ALQ" ...
 $ BsmtFinSF1   : int  706 978 486 216 655 732 1369 859 0 851 ...
 $ BsmtFinType2 : chr  "Unf" "Unf" "Unf" "Unf" ...
 $ BsmtFinSF2   : int  0 0 0 0 0 0 0 32 0 0 ...
 $ BsmtUnfSF    : int  150 284 434 540 490 64 317 216 952 140 ...
 $ TotalBsmtSF  : int  856 1262 920 756 1145 796 1686 1107 952 991 ...
 $ Exterior2nd  : chr  "VinylSd" "MetalSd" "VinylSd" "Wd Shng" ...
 $ MasVnrType   : chr  "BrkFace" "None" "BrkFace" "None" ...
 $ MasVnrArea   : int  196 0 162 0 350 0 186 240 0 0 ...
 $ ExterQual    : chr  "Gd" "TA" "Gd" "TA" ...
 $ ExterCond    : chr  "TA" "TA" "TA" "TA" ...
 $ Foundation   : chr  "PConc" "CBlock" "PConc" "BrkTil" ...
 $ YearBuilt    : int  2003 1976 2001 1915 2000 1993 2004 1973 1931 1939 ...
 $ YearRemodAdd : int  2003 1976 2002 1970 2000 1995 2005 1973 1950 1950 ...
 $ RoofStyle    : chr  "Gable" "Gable" "Gable" "Gable" ...
 $ RoofMatl     : chr  "CompShg" "CompShg" "CompShg" "CompShg" ...
 $ Exterior1st  : chr  "VinylSd" "MetalSd" "VinylSd" "Wd Sdng" ...
 $ BldgType     : chr  "1Fam" "1Fam" "1Fam" "1Fam" ...
 $ HouseStyle   : chr  "2Story" "1Story" "2Story" "2Story" ...
 $ OverallQual  : int  7 6 7 7 8 5 8 7 7 5 ...
 $ OverallCond  : int  5 8 5 5 5 5 5 6 5 6 ...
 $ Neighborhood : chr  "CollgCr" "Veenker" "CollgCr" "Crawfor" ...
 $ Condition1   : chr  "Norm" "Feedr" "Norm" "Norm" ...
 $ Condition2   : chr  "Norm" "Norm" "Norm" "Norm" ...
 $ LandContour  : chr  "Lvl" "Lvl" "Lvl" "Lvl" ...
 $ Utilities    : chr  "AllPub" "AllPub" "AllPub" "AllPub" ...
 $ LotConfig    : chr  "Inside" "FR2" "Inside" "Corner" ...
 $ LandSlope    : chr  "Gtl" "Gtl" "Gtl" "Gtl" ...
 $ LotArea      : int  8450 9600 11250 9550 14260 14115 10084 10382 6120 7420 ...
 $ Street       : chr  "Pave" "Pave" "Pave" "Pave" ...
 $ Alley        : chr  NA NA NA NA ...
 $ LotShape     : chr  "Reg" "Reg" "IR1" "IR1" ...
 $ MSZoning     : chr  "RL" "RL" "RL" "RL" ...
 $ LotFrontage  : int  65 80 68 60 84 85 75 NA 51 50 ...
 $ MoSold       : int  2 5 9 2 12 10 8 11 4 1 ...
 $ YrSold       : int  2008 2007 2008 2006 2008 2009 2007 2009 2008 2008 ...
 $ SaleType     : chr  "WD" "WD" "WD" "WD" ...
 $ SaleCondition: chr  "Normal" "Normal" "Normal" "Abnorml" ...
 $ SalePrice    : int  208500 181500 223500 140000 250000 143000 307000 200000 129900 1180
00 ...
> summary(house)
       Id          MSSubClass      ScreenPorch        PoolArea          PoolQC
 Fence          MiscFeature          MiscVal          GarageType
```

```
 Min.   :   1.0   Min.   : 20.00   Min.   :  0.00   Min.   :  0.000   Length:2919
Length:2919        Length:2919        Min.   :    0.00   Length:2919
 1st Qu.: 730.5   1st Qu.: 20.00   1st Qu.:  0.00   1st Qu.:  0.000   Class :character
Class :character   Class :character   1st Qu.:    0.00   Class :character
 Median :1460.0   Median : 50.00   Median :  0.00   Median :  0.000   Mode  :character
Mode  :character   Mode  :character   Median :    0.00   Mode  :character
 Mean   :1460.0   Mean   : 57.14   Mean   : 16.06   Mean   :  2.252
Mean   :   50.83
 3rd Qu.:2189.5   3rd Qu.: 70.00   3rd Qu.:  0.00   3rd Qu.:  0.000
3rd Qu.:    0.00
 Max.   :2919.0   Max.   :190.00   Max.   :576.00   Max.   :800.000
Max.   :17000.00

  GarageYrBlt   GarageFinish         GarageCars      GarageArea      GarageQual         G
arageCond          PavedDrive          WoodDeckSF       OpenPorchSF
 Min.   :1895   Length:2919        Min.   :0.000   Min.   :   0.0   Length:2919        Le
ngth:2919          Length:2919        Min.   :   0.00   Min.   :  0.00
 1st Qu.:1960   Class :character   1st Qu.:1.000   1st Qu.: 320.0   Class :character   Cl
ass :character   Class :character   1st Qu.:   0.00   1st Qu.:  0.00
 Median :1979   Mode  :character   Median :2.000   Median : 480.0   Mode  :character   Mo
de  :character   Mode  :character   Median :   0.00   Median : 26.00
 Mean   :1978                      Mean   :1.767   Mean   : 472.9
Mean   :  93.71   Mean   : 47.49
 3rd Qu.:2002                      3rd Qu.:2.000   3rd Qu.: 576.0
3rd Qu.: 168.00   3rd Qu.: 70.00
 Max.   :2207                      Max.   :5.000   Max.   :1488.0
Max.   :1424.00   Max.   :742.00
 NA's   :159                       NA's   :1       NA's   :1
 EnclosedPorch      X3SsnPorch       BsmtFullBath     BsmtHalfBath        FullBath
HalfBath       BedroomAbvGr   KitchenAbvGr   KitchenQual
 Min.   :   0.0   Min.   :  0.000   Min.   :0.0000   Min.   :0.00000   Min.   :0.000   Mi
n.   :0.0000   Min.   :0.00   Min.   :0.000   Length:2919
 1st Qu.:   0.0   1st Qu.:  0.000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:1.000   1s
t Qu.:0.0000   1st Qu.:2.00   1st Qu.:1.000   Class :character
 Median :   0.0   Median :  0.000   Median :0.0000   Median :0.00000   Median :2.000   Me
dian :0.0000   Median :3.00   Median :1.000   Mode  :character
 Mean   :  23.1   Mean   :  2.602   Mean   :0.4299   Mean   :0.06136   Mean   :1.568   Me
an   :0.3803   Mean   :2.86   Mean   :1.045
 3rd Qu.:   0.0   3rd Qu.:  0.000   3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:2.000   3r
d Qu.:1.0000   3rd Qu.:3.00   3rd Qu.:1.000
 Max.   :1012.0   Max.   :508.000   Max.   :3.0000   Max.   :2.00000   Max.   :4.000   Ma
x.   :2.0000   Max.   :8.00   Max.   :3.000
                                   NA's   :2        NA's   :2
  TotRmsAbvGrd     Functional          Fireplaces     FireplaceQu          Heating
HeatingQC          CentralAir         Electrical          X1stFlrSF
 Min.   : 2.000   Length:2919        Min.   :0.0000   Length:2919        Length:2919
Length:2919        Length:2919        Length:2919        Min.   : 334
 1st Qu.: 5.000   Class :character   1st Qu.:0.0000   Class :character   Class :character
Class :character   Class :character   Class :character   1st Qu.: 876
 Median : 6.000   Mode  :character   Median :1.0000   Mode  :character   Mode  :character
Mode  :character   Mode  :character   Mode  :character   Median :1082
 Mean   : 6.452                     Mean   :0.5971
Mean   :1160
 3rd Qu.: 7.000                     3rd Qu.:1.0000
3rd Qu.:1388
 Max.   :15.000                     Max.   :4.0000
Max.   :5095

   X2ndFlrSF       LowQualFinSF        GrLivArea      BsmtQual           BsmtCond
BsmtExposure       BsmtFinType1         BsmtFinSF1     BsmtFinType2
 Min.   :   0.0   Min.   :   0.000   Min.   : 334   Length:2919        Length:2919
Length:2919        Length:2919        Min.   :   0.0   Length:2919
 1st Qu.:   0.0   1st Qu.:   0.000   1st Qu.:1126   Class :character   Class :character
Class :character   Class :character   1st Qu.:   0.0   Class :character
 Median :   0.0   Median :   0.000   Median :1444   Mode  :character   Mode  :character
Mode  :character   Mode  :character   Median : 368.5   Mode  :character
 Mean   : 336.5   Mean   :   4.694   Mean   :1501
Mean   : 441.4
 3rd Qu.: 704.0   3rd Qu.:   0.000   3rd Qu.:1744
3rd Qu.: 733.0
 Max.   :2065.0   Max.   :1064.000   Max.   :5642
Max.   :5644.0

NA's   :1
   BsmtFinSF2        BsmtUnfSF       TotalBsmtSF     Exterior2nd         MasVnrType
MasVnrArea      ExterQual          ExterCond          Foundation
```

```
 Min.   :   0.00   Min.   :   0.0   Min.   :   0.0   Length:2919        Length:2919
Min.   :   0.0   Length:2919        Length:2919        Length:2919
 1st Qu.:   0.00   1st Qu.: 220.0   1st Qu.: 793.0   Class :character   Class :character
1st Qu.:   0.0   Class :character   Class :character   Class :character
 Median :   0.00   Median : 467.0   Median : 989.5   Mode  :character   Mode  :character
Median :   0.0   Mode  :character   Mode  :character   Mode  :character
 Mean   :  49.58   Mean   : 560.8   Mean   :1051.8
Mean   : 102.2
 3rd Qu.:   0.00   3rd Qu.: 805.5   3rd Qu.:1302.0
3rd Qu.: 164.0
 Max.   :1526.00   Max.   :2336.0   Max.   :6110.0
Max.   :1600.0
 NA's   :1         NA's   :1        NA's   :1
NA's   :23
   YearBuilt     YearRemodAdd   RoofStyle           RoofMatl         Exterior1st
BldgType          HouseStyle         OverallQual      OverallCond
 Min.   :1872   Min.   :1950   Length:2919        Length:2919        Length:2919        L
ength:2919        Length:2919        Min.   : 1.000   Min.   :1.000
 1st Qu.:1954   1st Qu.:1965   Class :character   Class :character   Class :character   C
lass :character   Class :character   1st Qu.: 5.000   1st Qu.:5.000
 Median :1973   Median :1993   Mode  :character   Mode  :character   Mode  :character   M
ode  :character   Mode  :character   Median : 6.000   Median :5.000
 Mean   :1971   Mean   :1984
Mean   : 6.089   Mean   :5.565
 3rd Qu.:2001   3rd Qu.:2004
3rd Qu.: 7.000   3rd Qu.:6.000
 Max.   :2010   Max.   :2010
Max.   :10.000   Max.   :9.000

 Neighborhood        Condition1         Condition2        LandContour         Utilities
LotConfig          LandSlope          LotArea
 Length:2919        Length:2919        Length:2919        Length:2919        Length:2919
Length:2919        Length:2919        Min.   :  1300
 Class :character   Class :character   Class :character   Class :character   Class :chara
cter   Class :character   Class :character   1st Qu.:  7478
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :chara
cter   Mode  :character   Mode  :character   Median :  9453

Mean   : 10168

3rd Qu.: 11570

Max.   :215245

    Street             Alley             LotShape           MSZoning          LotFrontage
MoSold           YrSold       SaleType          SaleCondition
 Length:2919        Length:2919        Length:2919        Length:2919        Min.   : 21.
00   Min.   : 1.000   Min.   :2006   Length:2919        Length:2919
 Class :character   Class :character   Class :character   Class :character   1st Qu.: 59.
00   1st Qu.: 4.000   1st Qu.:2007   Class :character   Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Median : 68.
00   Median : 6.000   Median :2008   Mode  :character   Mode  :character
                                                                             Mean   : 69.
31   Mean   : 6.213   Mean   :2008

00   3rd Qu.: 8.000   3rd Qu.:2009                                           3rd Qu.: 80.

00   Max.   :12.000   Max.   :2010                                           Max.   :313.
                                                                             NA's   :486

   SalePrice
 Min.   : 34900
 1st Qu.:129975
 Median :163000
 Mean   :180921
 3rd Qu.:214000
 Max.   :755000
 NA's   :1459
```

```
> head(house)

  Id MSSubClass ScreenPorch PoolArea PoolQC Fence MiscFeature MiscVal GarageType GarageYrB
lt GarageFinish GarageCars GarageArea GarageQual GarageCond PavedDrive
1  1         60           0        0   <NA>  <NA>        <NA>       0     Attchd        200
3          RFn          2        548         TA         TA          Y
2  2         20           0        0   <NA>  <NA>        <NA>       0     Attchd        197
6          RFn          2        460         TA         TA          Y
```

```
3  3         60           0        0   <NA>  <NA>         <NA>        0     Attchd         200
1          RFn          2        608         TA          TA          Y
4  4         70           0        0   <NA>  <NA>         <NA>        0     Detchd         199
8          Unf          3        642         TA          TA          Y
5  5         60           0        0   <NA>  <NA>         <NA>        0     Attchd         200
0          RFn          3        836         TA          TA          Y
6  6         50           0        0   <NA> MnPrv         Shed      700     Attchd         199
3          Unf          2        480         TA          TA          Y
  WoodDeckSF OpenPorchSF EnclosedPorch X3SsnPorch BsmtFullBath BsmtHalfBath FullBath HalfB
ath BedroomAbvGr KitchenAbvGr KitchenQual TotRmsAbvGrd Functional Fireplaces
1          0          61             0          0            1            0        2        1
3            1          Gd             8        Typ           0
2        298           0             0          0            0            1        2        0
3            1          TA             6        Typ           1
3          0          42             0          0            1            0        2        1
3            1          Gd             6        Typ           1
4          0          35           272          0            1            0        1        0
3            1          Gd             7        Typ           1
5        192          84             0          0            1            0        2        1
4            1          Gd             9        Typ           1
6         40          30             0        320            1            0        1        1
1            1          TA             5        Typ           0
  FireplaceQu Heating HeatingQC CentralAir Electrical X1stFlrSF X2ndFlrSF LowQualFinSF GrL
ivArea BsmtQual BsmtCond BsmtExposure BsmtFinType1 BsmtFinSF1 BsmtFinType2
1        <NA>    GasA        Ex          Y      SBrkr       856       854            0      1
710       Gd       TA            No          GLQ        706          Unf
2         TA    GasA        Ex          Y      SBrkr      1262         0            0      1
262       Gd       TA            Gd          ALQ        978          Unf
3         TA    GasA        Ex          Y      SBrkr       920       866            0      1
786       Gd       TA            Mn          GLQ        486          Unf
4         Gd    GasA        Gd          Y      SBrkr       961       756            0      1
717       TA       Gd            No          ALQ        216          Unf
5         TA    GasA        Ex          Y      SBrkr      1145      1053            0      2
198       Gd       TA            Av          GLQ        655          Unf
6        <NA>    GasA        Ex          Y      SBrkr       796       566            0      1
362       Gd       TA            No          GLQ        732          Unf
  BsmtFinSF2 BsmtUnfSF TotalBsmtSF Exterior2nd MasVnrType MasVnrArea ExterQual ExterCond F
oundation YearBuilt YearRemodAdd RoofStyle RoofMatl Exterior1st BldgType
1          0       150          856     VinylSd    BrkFace         196        Gd        TA
PConc       2003         2003     Gable  CompShg     VinylSd     1Fam
2          0       284         1262     MetalSd       None           0        TA        TA
CBlock       1976         1976     Gable  CompShg     MetalSd     1Fam
3          0       434          920     VinylSd    BrkFace         162        Gd        TA
PConc       2001         2002     Gable  CompShg     VinylSd     1Fam
4          0       540          756     Wd Shng       None           0        TA        TA
BrkTil       1915         1970     Gable  CompShg     Wd Sdng     1Fam
5          0       490         1145     VinylSd    BrkFace         350        Gd        TA
PConc       2000         2000     Gable  CompShg     VinylSd     1Fam
6          0        64          796     VinylSd       None           0        TA        TA
Wood       1993         1995     Gable  CompShg     VinylSd     1Fam
  HouseStyle OverallQual OverallCond Neighborhood Condition1 Condition2 LandContour Utilit
ies LotConfig LandSlope LotArea Street Alley LotShape MSZoning LotFrontage
1     2Story           7           5      CollgCr       Norm       Norm         Lvl    AllP
ub    Inside       Gtl    8450   Pave  <NA>      Reg       RL          65
2     1Story           6           8      Veenker      Feedr       Norm         Lvl    AllP
ub       FR2       Gtl    9600   Pave  <NA>      Reg       RL          80
3     2Story           7           5      CollgCr       Norm       Norm         Lvl    AllP
ub    Inside       Gtl   11250   Pave  <NA>      IR1       RL          68
4     2Story           7           5      Crawfor       Norm       Norm         Lvl    AllP
ub    Corner       Gtl    9550   Pave  <NA>      IR1       RL          60
5     2Story           8           5      NoRidge       Norm       Norm         Lvl    AllP
ub       FR2       Gtl   14260   Pave  <NA>      IR1       RL          84
6     1.5Fin           5           5      Mitchel       Norm       Norm         Lvl    AllP
ub    Inside       Gtl   14115   Pave  <NA>      IR1       RL          85
  MoSold YrSold SaleType SaleCondition SalePrice
1      2   2008       WD        Normal    208500
2      5   2007       WD        Normal    181500
3      9   2008       WD        Normal    223500
4      2   2006       WD       Abnorml    140000
```

```
5      12    2008        WD         Normal     250000
6      10    2009        WD         Normal     143000
```

Data Visualization

The first step to any data science project is simple exploration and visualization of the data. Since the ultimate purpose of this competition is price prediction, it's a good idea to visualize price trends over the time span of the training data set. The visualization below shows monthly average realty prices over time.

```
> ##Data Visualization
> cat_var <- names(train)[which(sapply(train, is.character))]
> cat_car <- c(cat_var, 'BedroomAbvGr', 'HalfBath', ' KitchenAbvGr','BsmtFullBath', 'BsmtH
alfBath', 'MSSubClass')
> numeric_var <- names(train)[which(sapply(train, is.numeric))]
```

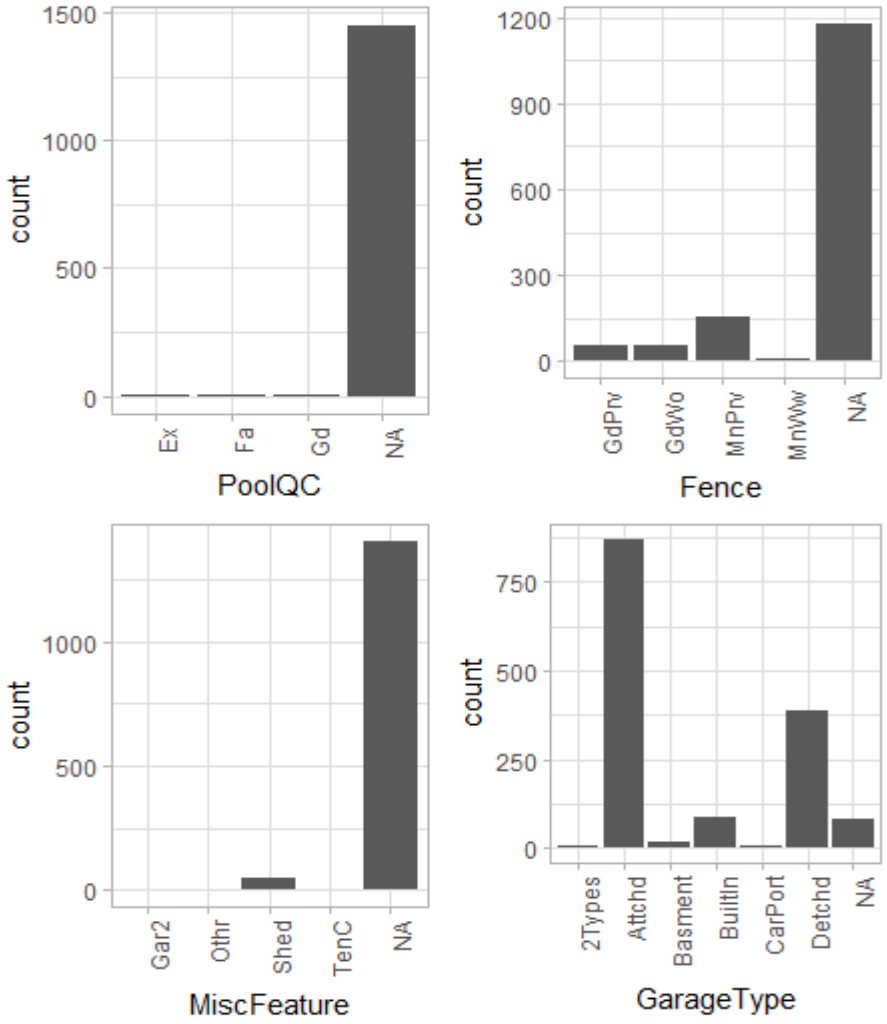
Creating one training dataset with categorical variable and one with numeric variable. I will use this for data visualization.

```
> train1_cat<-train[cat_var]
> train1_num<-train[numeric_var]
```

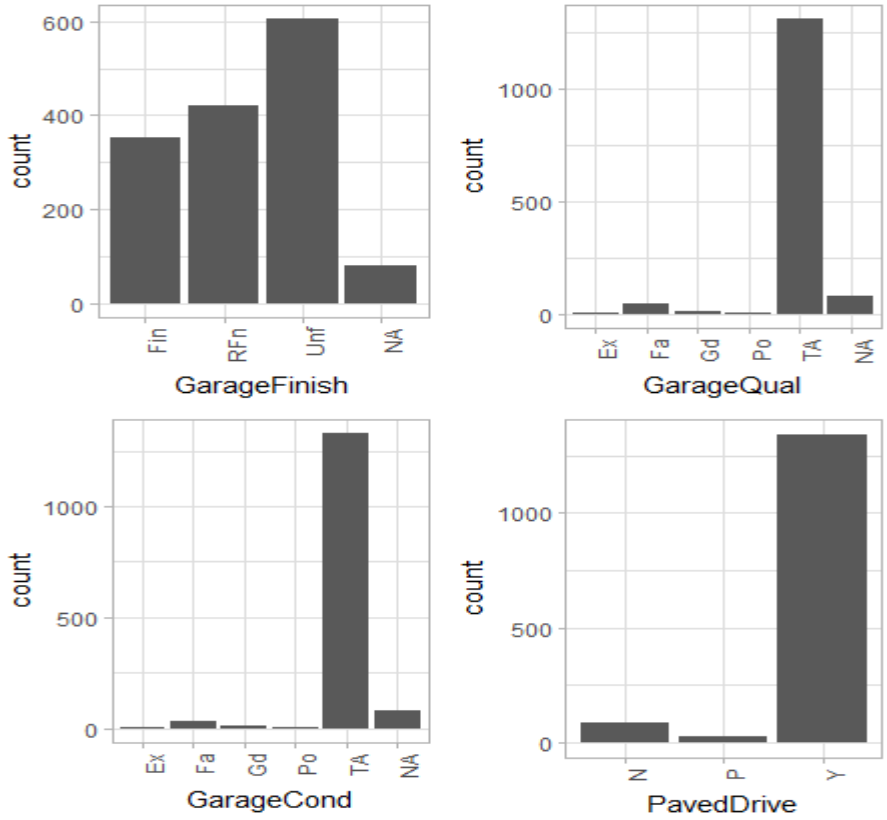
Create Bar Plot and Density Plot function and then make a function to call both the plot function.

```
> # Bar plot/Density plot function
> ## Bar plot function
> plotHist <- function(data_in, i)
+ {
+   data <- data.frame(x=data_in[[i]])
+   p <- ggplot(data=data, aes(x=factor(x))) + stat_count() + xlab(colnames(data_in)[i]) +
theme_light() +
+     theme(axis.text.x = element_text(angle = 90, hjust =1))
+   return (p)
+ }

> ## Density plot function
> plotDen <- function(data_in, i){
+   data <- data.frame(x=data_in[[i]], SalePrice = data_in$SalePrice)
+   p <- ggplot(data= data) + geom_line(aes(x = x), stat = 'density', size = 1,alpha = 1.0
) +
+     xlab(paste0((colnames(data_in)[i]), '\n', 'Skewness: ',round(skewness(data_in[[i]],
na.rm = TRUE), 2))) + theme_light()
+   return(p)
+ }

> ## Function to call both Bar plot and Density plot function
> doPlots <- function(data_in, fun, ii, ncol=3)
+ {
+   pp <- list()
+   for (i in ii) {
+     p <- fun(data_in=data_in, i=i)
+     pp <- c(pp, list(p))
+   }
+   do.call("grid.arrange", c(pp, ncol=ncol))
+ }
```

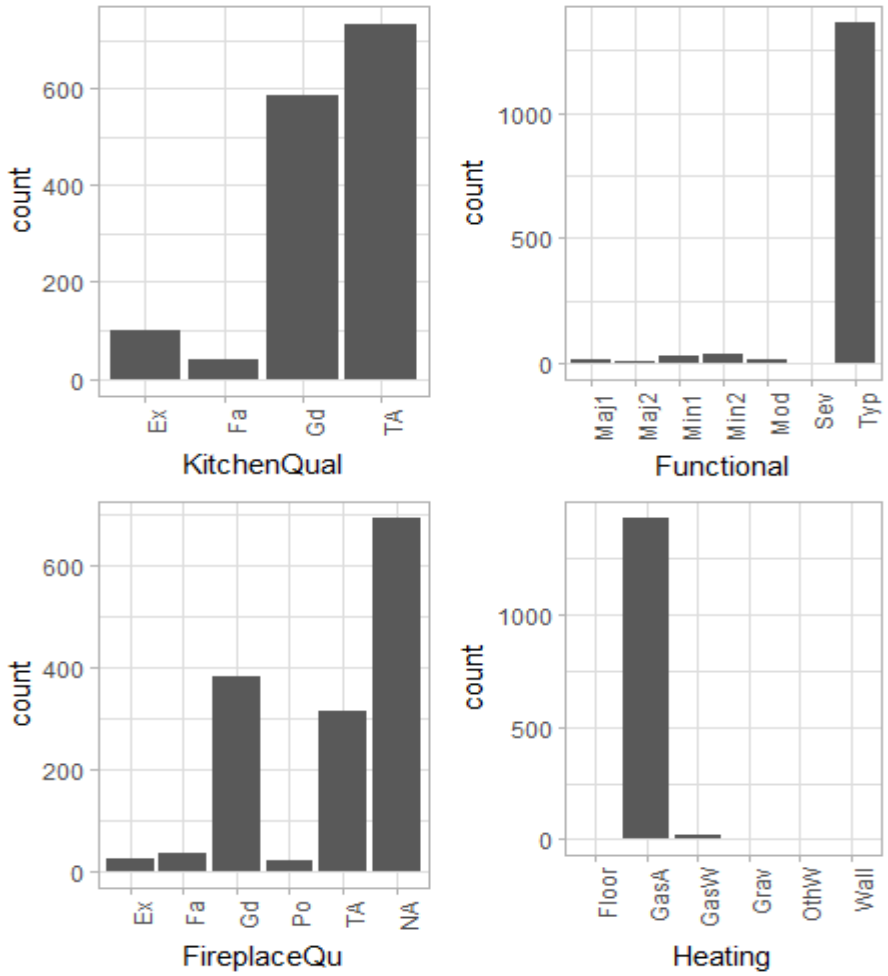
```
> ## Barplots for the categorical features
> doPlots(train1_cat, fun = plotHist, ii = 1:4, ncol = 2)
```



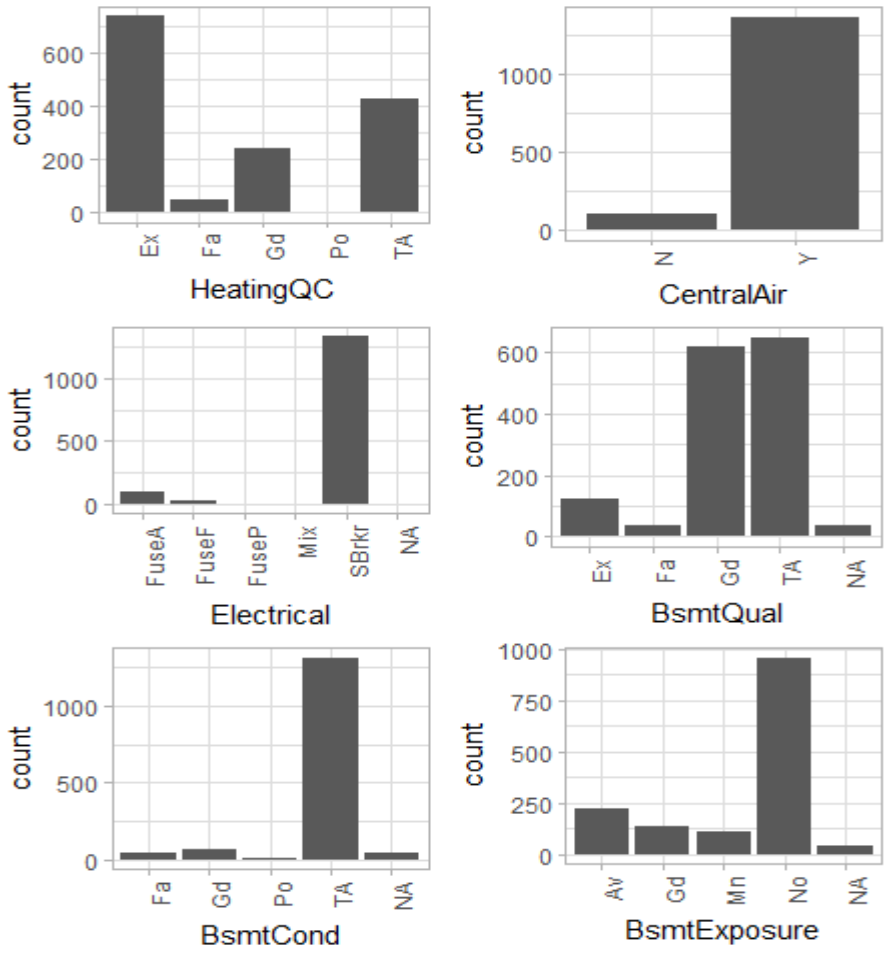

```
> doPlots(train1_cat, fun = plotHist, ii = 5:8, ncol = 2)
```



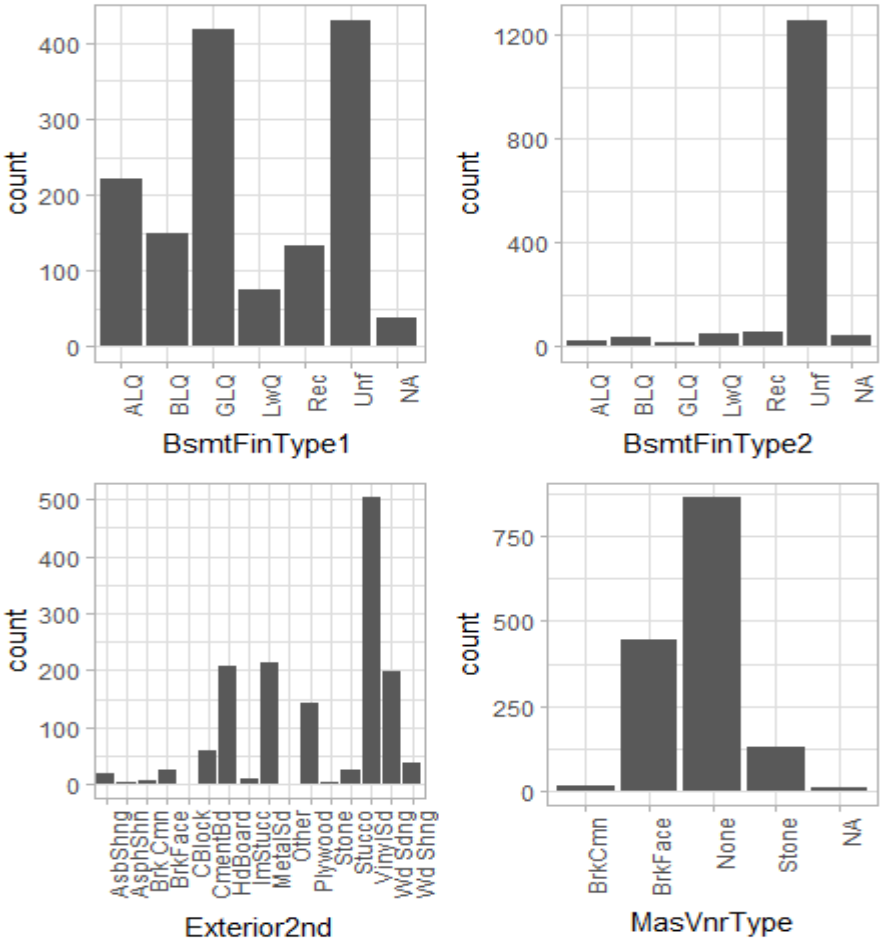
```
> doPlots(train1_cat, fun = plotHist, ii = 9:12, ncol = 2)
```




```
> doPlots(train1_cat, fun = plotHist, ii = 13:18, ncol = 2)
```



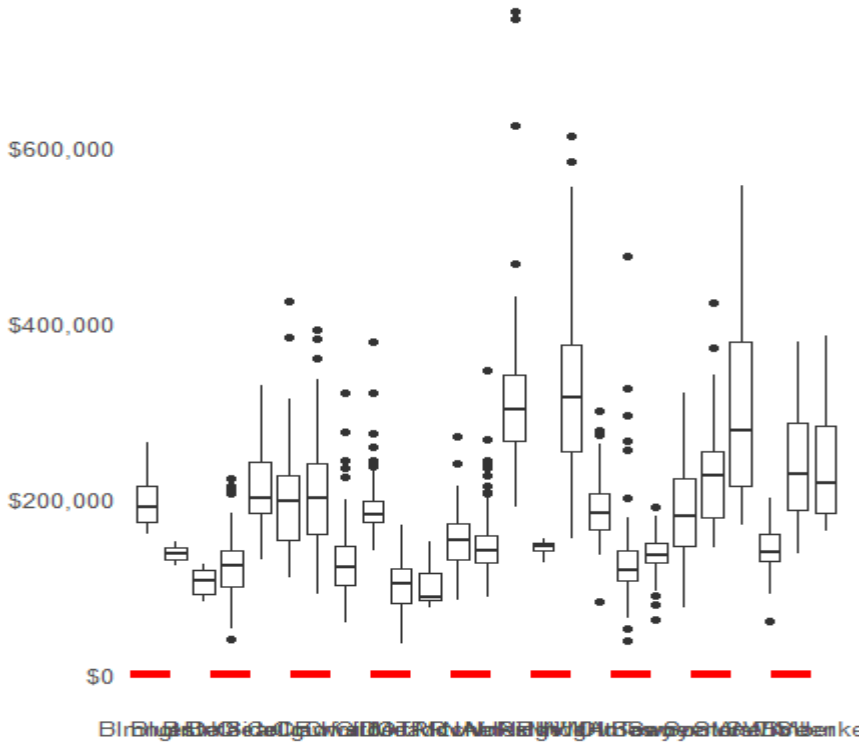
```
> doPlots(train1_cat, fun = plotHist, ii = 19:22, ncol = 2)
```



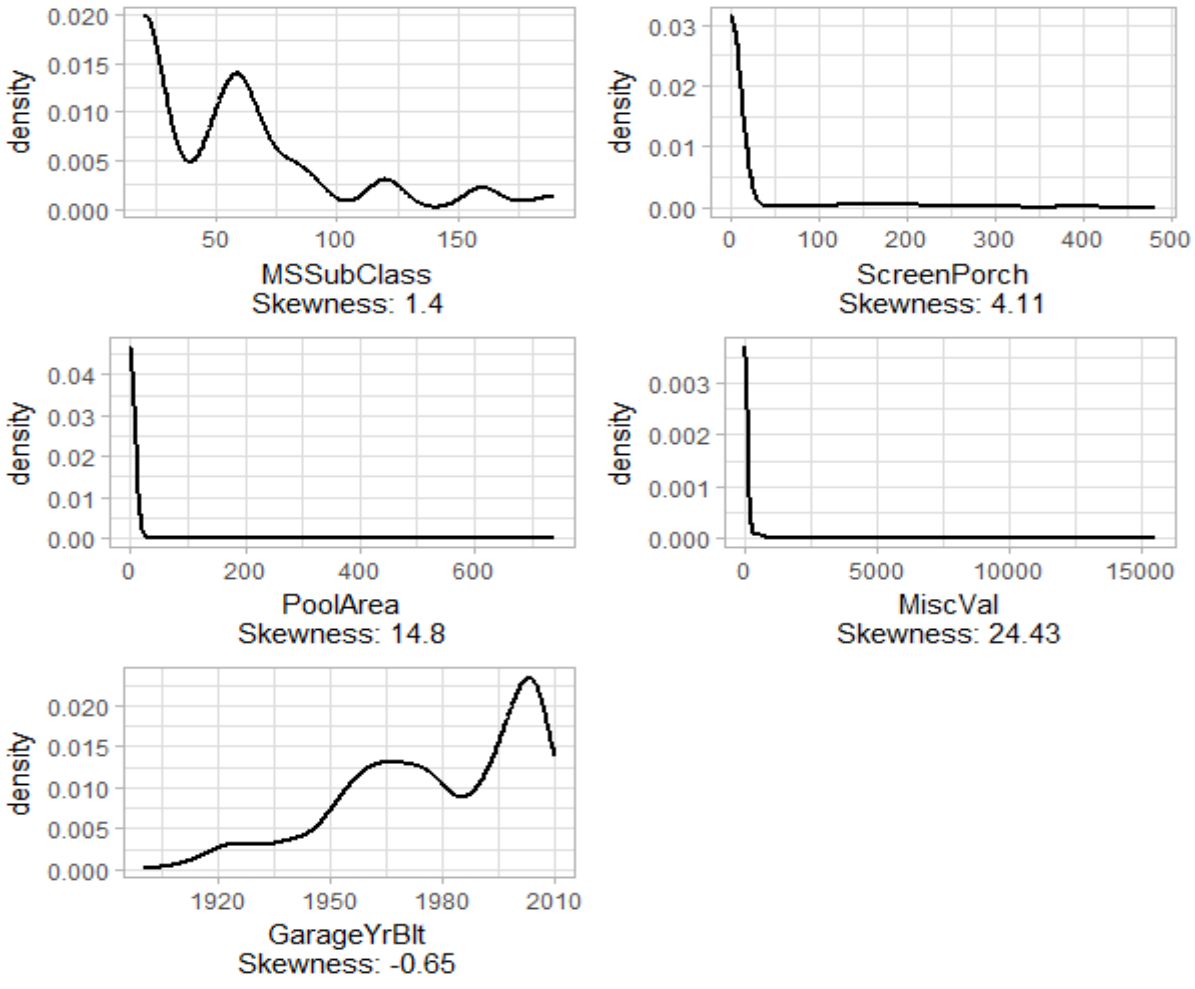
Bar-Plot represents the total number of entities count vs entities present in the data-set. These entities are mainly of categorical type.

```
> ##Boxplot
> ggplot(train, aes(x = Neighborhood, y = SalePrice)) +
+   geom_boxplot() +
+   geom_hline(aes(yintercept=80),
+              colour='red', linetype='dashed', lwd=2) +
+   scale_y_continuous(labels=dollar_format()) +
+   theme_few()
```

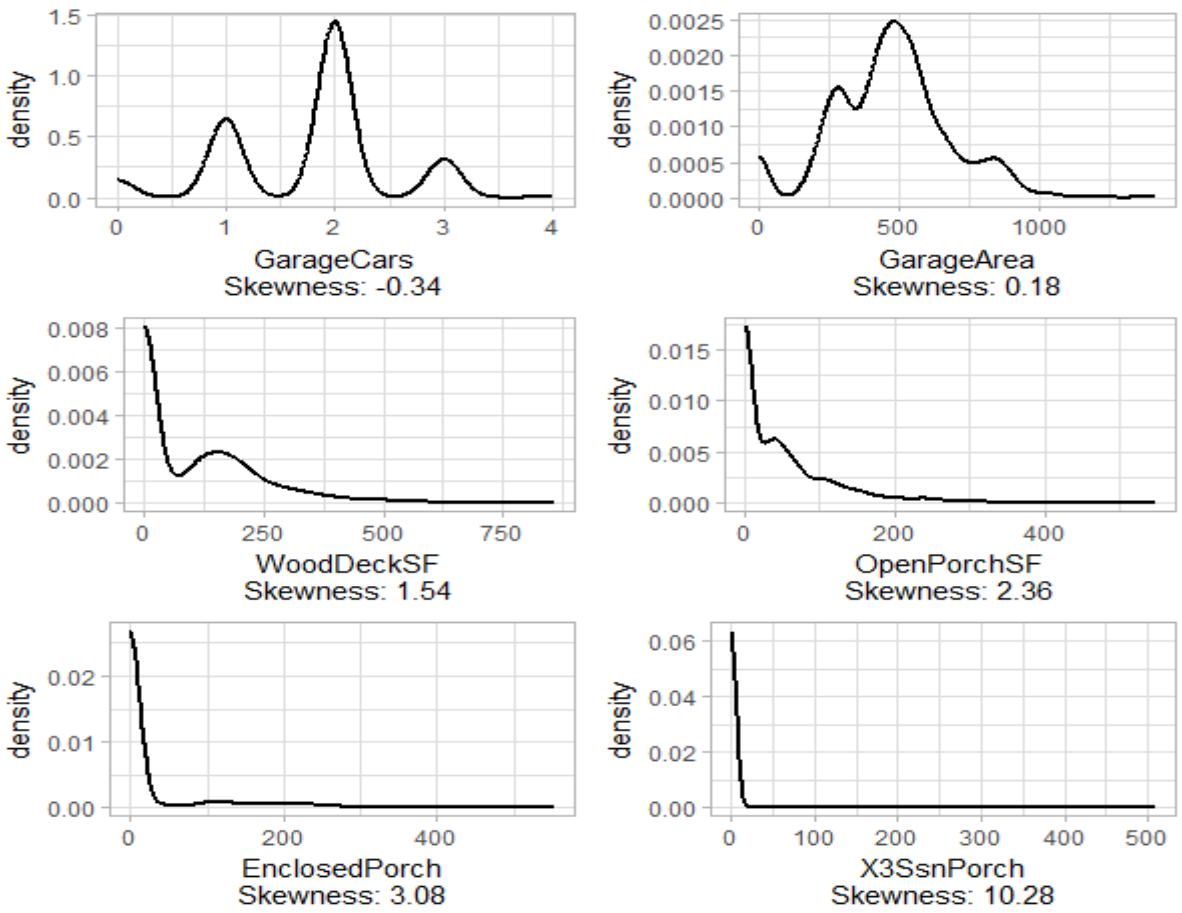
Box-Plot was plotted between Neighborhood and SalePrice. It represent the maximum and minimum value due to neighborhood locality. In the graph we can see that there are two neighborhood where there the minimum value of saleprice is much higher than other neighborhood locations.



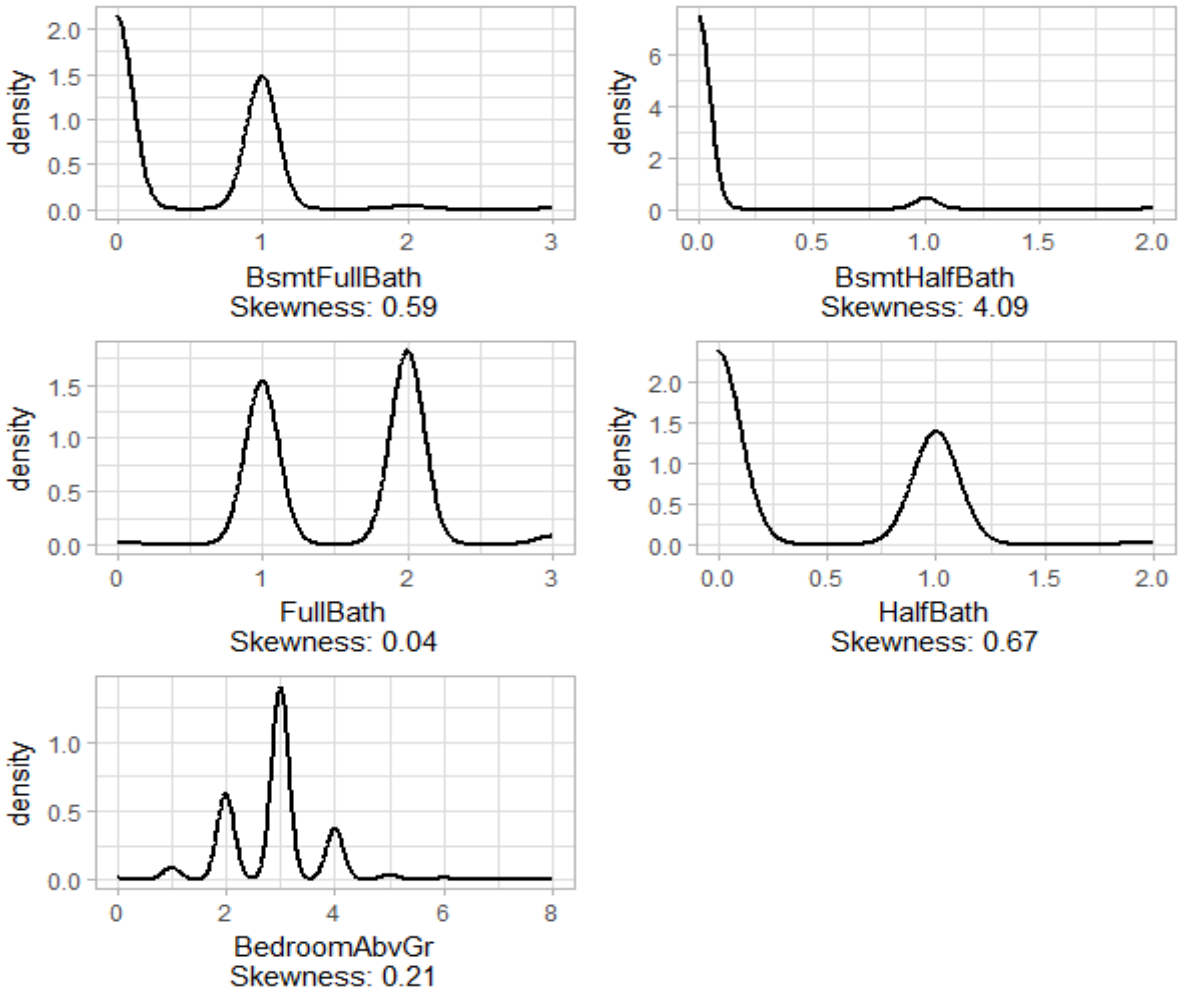

```
> #Density plots for numeric variables.
> doPlots(train1_num, fun = plotDen, ii = 2:6, ncol = 2)
```



```
> doPlots(train1_num, fun = plotDen, ii = 7:12, ncol = 2)
```

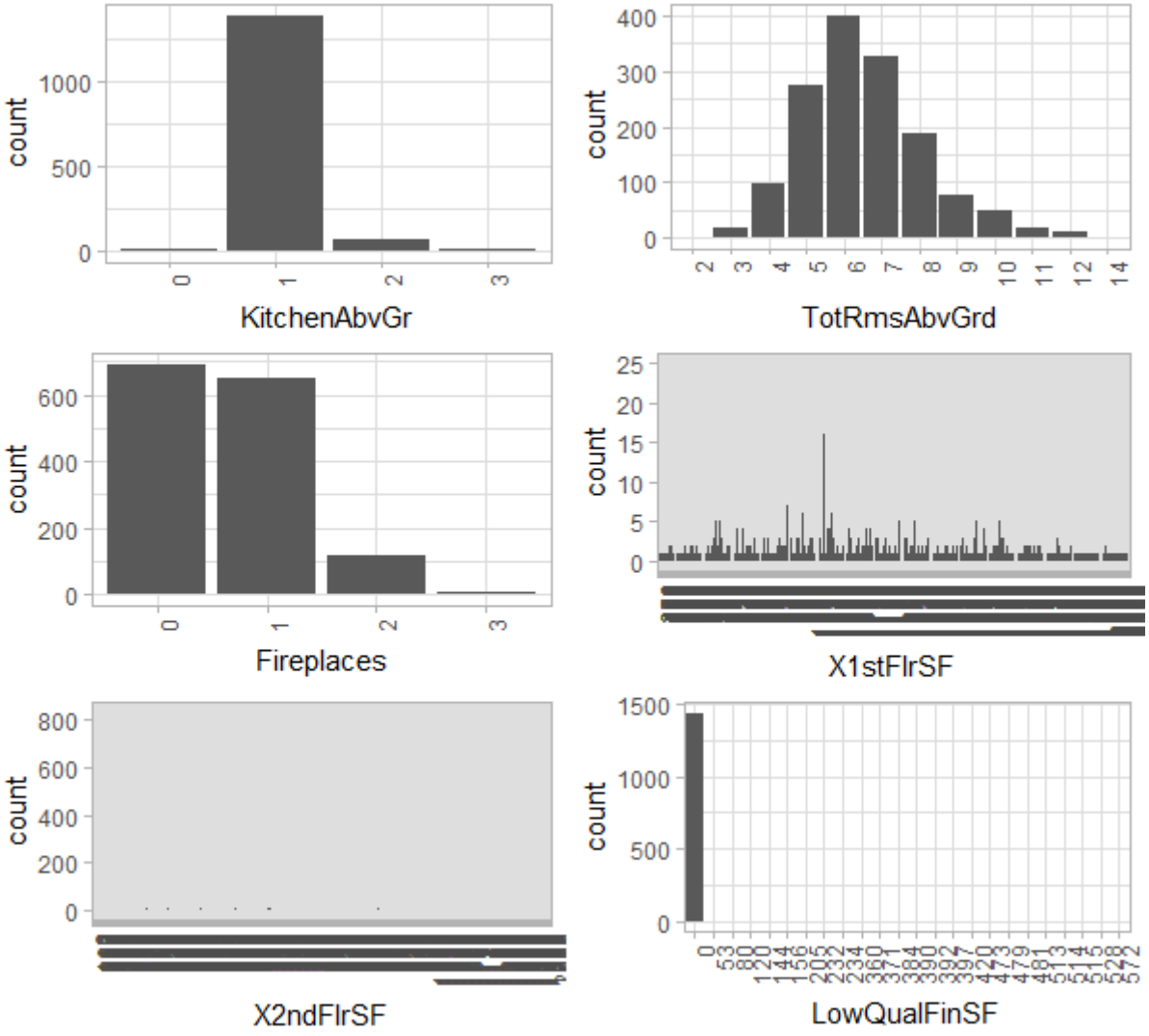



```
> doPlots(train1_num, fun = plotDen, ii = 13:17, ncol = 2)
```



Density Plot Curve gives the idea about the skewness of the data. Skewness is a measure of the asymmetry of the probability distribution of a real-valued random variable about its mean. The skewness value can be positive or negative, or undefined.

```
> #Histogram for few numeric variable
> doPlots(train1_num, fun = plotHist, ii = 18:23, ncol = 2)
```

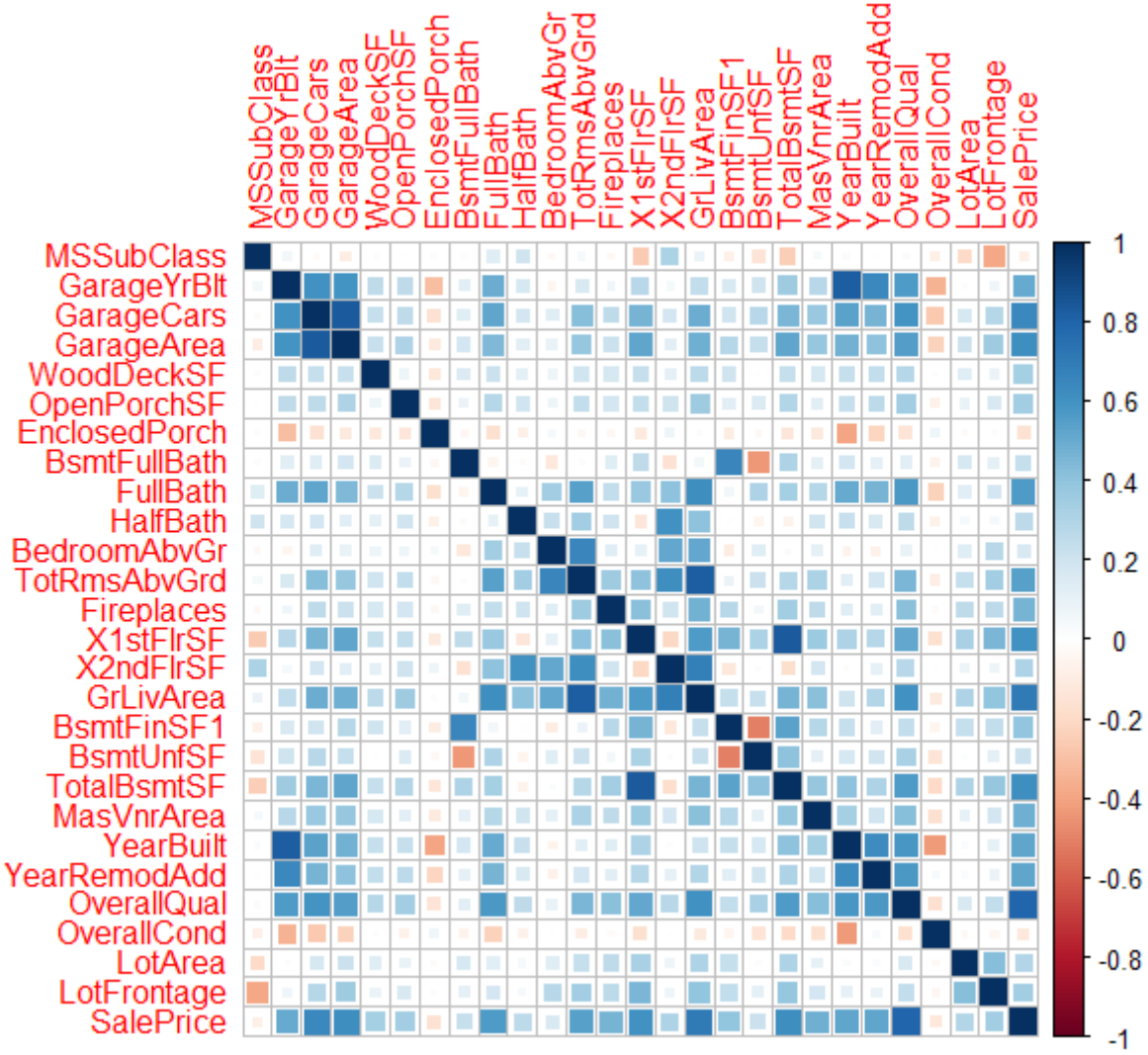


Histogram represents the total number of entities count vs entities present in the data-set. These entities are mainly of integer type. This represent there are mainly 1 Kitchen, 6-7 Total Rooms, 0 or 1 Fireplaces in majority of the houses.

```
> ##Explore the correlation
> correlations <- cor(na.omit(train1_num[,-1]))
> #correlations
> row_indic <- apply(correlations, 1, function(x) sum(x > 0.3 | x < -0.3) > 1)
> correlations<- correlations[row_indic ,row_indic ]
> corrplot(correlations, method="square")
```

Correlation is a statistical technique that can show whether and how strongly pairs of variables are related. In the below curve blue square represent positive correlation and red square represent negative correlation. Darker the part higher will be the correlation either it will be

positive or negative correlation.



Boxplot between the neighborhoods and sale price shows that Brookside and South & West of Iowa State University have cheap houses. While Northridge and Northridge Heights are rich neighborhoods with several outliers in terms of price.

Density plots of the features indicates that the features are skewed. The density plot for YearBuilt shows that the data set contains a mix of new and old houses. It shows a downturn in the number of houses in recent years, possibly due to the housing crisis.

The histograms shows that majority of the houses have 2 full baths, 0 half baths, and have an average of 3 bedrooms.

Missing Values

Now we look at the distribution and summary of target variables. From summary, it was observed that minimum price is greater than 0. After plotting the histogram we could see that it deviates from normal distribution and has positive skewness. For checking outliers we plotted 'GrLivArea' too. We found the outliers in GrLivArea field so remove those outliers. Then find the missing values in combined dataset and merge them in one variable.

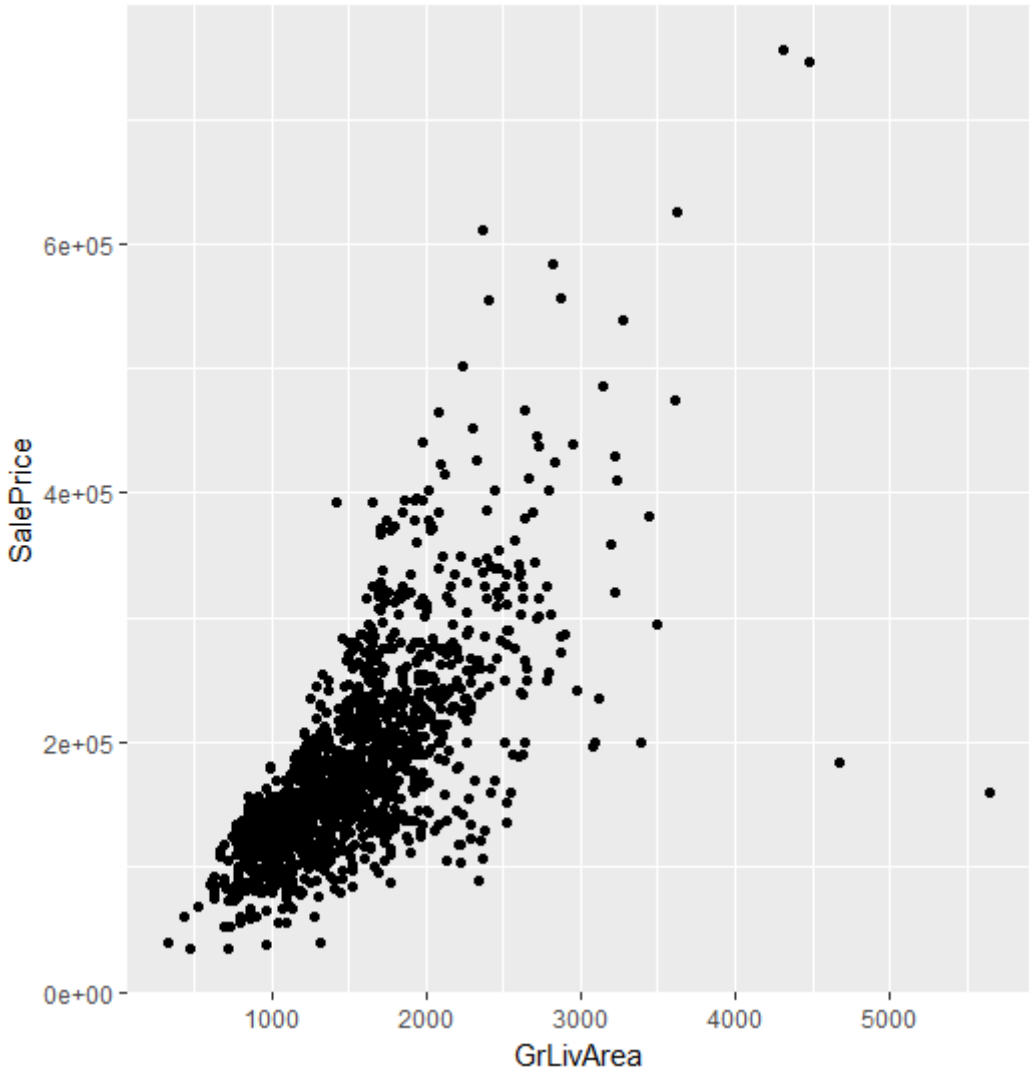
```
> #Looking for missing value
> ##Looking at the distribution and summary of the target variable
> summary(train$SalePrice)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  34900  129975  163000  180921  214000  755000
> quantile(train$SalePrice)
    0%    25%    50%    75%   100%
 34900 129975 163000 214000 755000
```

From summary, it was observed that minimum price is greater than 0

```
>
> # Conclusion: From summary, it was observed that minimum price is greater
than 0
> ## Histogram for target variable
> hist(train$SalePrice)
```

From Histogram, we could see that it deviates from normal distribution and has positive skewness.

```
> ## Conclusion: From Histogram, we could see that it deviates from normal
distribution and has positive skewness.
> # Plotting 'GrLivArea' too see if there are any outliers
> ggplot(train,aes(y=SalePrice,x=GrLivArea))+geom_point()
```




```
> summary(train$GrLivArea)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    334    1130    1464    1515    1777    5642
```

An outlier is an observation that is numerically distant from the rest of the data. There are some outliers in 'GrLivArea' field. Let's remove those outliers.

```
> # There are outliers in 'GrLivArea' field. Let's remove those outliers.
> train <- train[train$GrLivArea<=4000,]
>
> ## To find number of missing value for all variable in train dataset
> colSums(sapply(train, is.na))
          Id    MSSubClass   ScreenPorch      PoolArea        PoolQC         Fence   Misc
Feature       MiscVal    GarageType   GarageYrBlt  GarageFinish
           0           0           0           0        1451        1176           1
402             0            81            81            81
   GarageCars    GarageArea    GarageQual    GarageCond    PavedDrive    WoodDeckSF   Open
PorchSF EnclosedPorch    X3SsnPorch  BsmtFullBath  BsmtHalfBath
           0             0            81            81             0             0
0             0             0             0             0
     FullBath      HalfBath  BedroomAbvGr  KitchenAbvGr   KitchenQual  TotRmsAbvGrd     Fun
ctional    Fireplaces   FireplaceQu       Heating     HeatingQC
           0             0             0             0             0             0
0             0           690             0             0
   CentralAir    Electrical     X1stFlrSF     X2ndFlrSF  LowQualFinSF     GrLivArea      B
smtQual      BsmtCond  BsmtExposure  BsmtFinType1    BsmtFinSF1
           0             1             0             0             0             0
37            37            38            37             0
 BsmtFinType2    BsmtFinSF2     BsmtUnfSF   TotalBsmtSF   Exterior2nd    MasVnrType    Mas
VnrArea     ExterQual     ExterCond    Foundation     YearBuilt
          38             0             0             0             0             8
8             0             0             0             0
```

```
 YearRemodAdd     RoofStyle      RoofMatl   Exterior1st      BldgType    HouseStyle   Over
allQual   OverallCond  Neighborhood    Condition1    Condition2
            0              0              0             0             0             0
0             0             0             0             0
  LandContour     Utilities     LotConfig     LandSlope       LotArea        Street         A
lley      LotShape      MSZoning   LotFrontage        MoSold
            0             0             0             0             0             0         1
365             0             0           259             0
       YrSold      SaleType SaleCondition     SalePrice
            0             0             0             0
>## To find number of missing value for all variable in combined dataset
(Train+Test)
> sapply(house[,1:80], function(x) sum(is.na(x)))
           Id    MSSubClass   ScreenPorch      PoolArea        PoolQC         Fence   Misc
Feature       MiscVal    GarageType   GarageYrBlt  GarageFinish
            0             0             0             0          2909          2348         2
814             0           157           159           159
   GarageCars    GarageArea    GarageQual    GarageCond    PavedDrive    WoodDeckSF   Open
PorchSF EnclosedPorch    X3SsnPorch  BsmtFullBath  BsmtHalfBath
            1             1           159           159             0             0
0             0             0             2             2
     FullBath      HalfBath  BedroomAbvGr  KitchenAbvGr   KitchenQual  TotRmsAbvGrd     Fun
ctional    Fireplaces   FireplaceQu       Heating     HeatingQC
            0             0             0             0             1             0
2             0          1420             0             0
   CentralAir    Electrical     X1stFlrSF     X2ndFlrSF  LowQualFinSF     GrLivArea      B
smtQual      BsmtCond  BsmtExposure  BsmtFinType1    BsmtFinSF1
            0             1             0             0             0             0
81            82            82           79             1
 BsmtFinType2    BsmtFinSF2     BsmtUnfSF   TotalBsmtSF   Exterior2nd    MasVnrType    Mas
VnrArea     ExterQual     ExterCond     Foundation     YearBuilt
           80             1             1             1             1            24
23             0             0             0             0
 YearRemodAdd     RoofStyle      RoofMatl   Exterior1st      BldgType    HouseStyle   Over
allQual   OverallCond  Neighborhood    Condition1    Condition2
            0             0             0             1             0             0
0             0             0             0             0
  LandContour     Utilities     LotConfig     LandSlope       LotArea        Street         A
lley      LotShape      MSZoning   LotFrontage        MoSold
            0             2             0             0             0             0         2
721             0             4           486             0
       YrSold      SaleType SaleCondition
            0             1             0
```

Let's deal with the missing values. Change the NA values of the categorical data to None so that it become purely categorical and also use mean and median for missing values in numerical data in the place of NA values so that data should be readable and then factorizing all the data.

```
> ## Taking all the missing data indices in one variables
> Missing_indices <- sapply(train,function(x) sum(is.na(x)))
> Missing_Summary <- data.frame(index = names(train),Missing_Values=Missing_indices)
> Missing_Summary[Missing_Summary$Missing_Values > 0,]
                  index Missing_Values
PoolQC             PoolQC          1451
Fence               Fence          1176
MiscFeature   MiscFeature          1402
GarageType     GarageType            81
GarageYrBlt   GarageYrBlt            81
GarageFinish GarageFinish            81
GarageQual     GarageQual            81
GarageCond     GarageCond            81
FireplaceQu   FireplaceQu           690
Electrical     Electrical             1
BsmtQual         BsmtQual            37
BsmtCond         BsmtCond            37
BsmtExposure BsmtExposure            38
BsmtFinType1 BsmtFinType1            37
```

```
BsmtFinType2 BsmtFinType2             38
MasVnrType     MasVnrType              8
MasVnrArea     MasVnrArea              8
Alley               Alley           1365
LotFrontage   LotFrontage            259
> #Combining train and test data for quicker data prep
> test$SalePrice <- NA
> train$isTrain <- 1
> test$isTrain <- 0
> house <- rbind(train,test)
> #MasVnrArea
> house$MasVnrArea[which(is.na(house$MasVnrArea))] <- mean(house$MasVnrArea,na.rm=T)
> #Alley
> ##Changing NA in Alley to None
> house$Alley1 <- as.character(house$Alley)
> house$Alley1[which(is.na(house$Alley))] <- "None"
> table(house$Alley1)

Grvl None Pave
 120 2717   78
> house$Alley <- as.factor(house$Alley1)
> house <- subset(house,select = -Alley1)
>
> #MasVnrType
> #Changing NA in MasVnrType to None
> house$MasVnrType1 <- as.character(house$MasVnrType)
> house$MasVnrType1[which(is.na(house$MasVnrType))] <- "None"
> house$MasVnrType <- as.factor(house$MasVnrType1)
> house <- subset(house,select = -MasVnrType1)
> table(house$MasVnrType)

 BrkCmn BrkFace    None   Stone
     25     878    1765     247
>
> #LotFrontage
> ##Imputing missing Lot Frontage by the median
> house$LotFrontage[which(is.na(house$LotFrontage))] <- median(house$LotFrontage,na.rm = T
)
>
> #FireplaceQu
> ##Changing NA in FireplaceQu to None
> house$FireplaceQu1 <- as.character(house$FireplaceQu)
> house$FireplaceQu1[which(is.na(house$FireplaceQu))] <- "None"
> house$FireplaceQu <- as.factor(house$FireplaceQu1)
> house <- subset(house,select = -FireplaceQu1)
>
> #PoolQC
> ##Changing NA in PoolQC to None
> house$PoolQC1 <- as.character(house$PoolQC)
> house$PoolQC1[which(is.na(house$PoolQC))] <- "None"
> house$PoolQC <- as.factor(house$PoolQC1)
> house <- subset(house,select = -PoolQC1)
>
> #Fence
> ##Changing NA in Fence to None
> house$Fence1 <- as.character(house$Fence)
> house$Fence1[which(is.na(house$Fence))] <- "None"
> house$Fence <- as.factor(house$Fence1)
> house <- subset(house,select = -Fence1)
>
> #MiscFeature
> ##Changing NA in MiscFeature to None
> house$MiscFeature1 <- as.character(house$MiscFeature)
> house$MiscFeature1[which(is.na(house$MiscFeature))] <- "None"
> house$MiscFeature <- as.factor(house$MiscFeature1)
> house <- subset(house,select = -MiscFeature1)
>
> #GarageType
```

```
> ##Changing NA in GarageType to None
> house$GarageType1 <- as.character(house$GarageType)
> house$GarageType1[which(is.na(house$GarageType))] <- "None"
> house$GarageType <- as.factor(house$GarageType1)
> house <- subset(house,select = -GarageType1)
>
> #GarageYrBlt
> ##Changing NA in GarageYrBlt to None
> house$GarageYrBlt[which(is.na(house$GarageYrBlt))] <- 0
>
> #GarageFinish
> ##Changing NA in GarageFinish to None
> house$GarageFinish1 <- as.character(house$GarageFinish)
> house$GarageFinish1[which(is.na(house$GarageFinish))] <- "None"
> house$GarageFinish <- as.factor(house$GarageFinish1)
> house <- subset(house,select = -GarageFinish1)
>
> #GarageQual
> ##Changing NA in GarageQual to None
> house$GarageQual1 <- as.character(house$GarageQual)
> house$GarageQual1[which(is.na(house$GarageQual))] <- "None"
> house$GarageQual <- as.factor(house$GarageQual1)
> house <- subset(house,select = -GarageQual1)
>
> #GarageCond
> ##Changing NA in GarageCond to None
> house$GarageCond1 <- as.character(house$GarageCond)
> house$GarageCond1[which(is.na(house$GarageCond))] <- "None"
> house$GarageCond <- as.factor(house$GarageCond1)
> house <- subset(house,select = -GarageCond1)
>
> #BsmtQual
> ##Changing NA in BsmtQual to None
> house$BsmtQual1 <- as.character(house$BsmtQual)
> house$BsmtQual1[which(is.na(house$BsmtQual))] <- "None"
> house$BsmtQual <- as.factor(house$BsmtQual1)
> house <- subset(house,select = -BsmtQual1)
>
> #BsmtCond
> ##Changing NA in BsmtCond to None
> house$BsmtCond1 <- as.character(house$BsmtCond)
> house$BsmtCond1[which(is.na(house$BsmtCond))] <- "None"
> house$BsmtCond <- as.factor(house$BsmtCond1)
> house <- subset(house,select = -BsmtCond1)
>
> #BsmtExposure
> ##Changing NA in BsmtExposure to None
> house$BsmtExposure1 <- as.character(house$BsmtExposure)
> house$BsmtExposure1[which(is.na(house$BsmtExposure))] <- "None"
> house$BsmtExposure <- as.factor(house$BsmtExposure1)
> house <- subset(house,select = -BsmtExposure1)
>
> #BsmtFinType1
> ##Changing NA in BsmtFinType1 to None
> house$BsmtFinType11 <- as.character(house$BsmtFinType1)
> house$BsmtFinType11[which(is.na(house$BsmtFinType1))] <- "None"
> house$BsmtFinType1 <- as.factor(house$BsmtFinType11)
> house <- subset(house,select = -BsmtFinType11)
>
> #BsmtFinType2
> #Changing NA in BsmtFinType2 to None
> house$BsmtFinType21 <- as.character(house$BsmtFinType2)
> house$BsmtFinType21[which(is.na(house$BsmtFinType2))] <- "None"
> house$BsmtFinType2 <- as.factor(house$BsmtFinType21)
> house <- subset(house,select = -BsmtFinType21)
>
> #Electrical
> ##Changing NA in Electrical to None
```

```
> house$Electrical1 <- as.character(house$Electrical)
> house$Electrical1[which(is.na(house$Electrical))] <- "None"
> house$Electrical <- as.factor(house$Electrical1)
> house <- subset(house,select = -Electrical1)
>
> #Factorizing
> house$MSZoning<- factor(house$MSZoning)
> house$Street <- factor(house$Street)
> house$LotShape <-factor(house$LotShape )
> house$LandContour<-factor(house$LandContour)
> house$Utilities<-factor(house$Utilities)
> house$LotConfig<-factor(house$LotConfig)
> house$LandSlope<-factor(house$LandSlope)
> house$Neighborhood<-factor(house$Neighborhood)
> house$Condition1<-factor(house$Condition1)
> house$Condition2<-factor(house$Condition2)
> house$BldgType<-factor(house$BldgType)
> house$HouseStyle<-factor(house$HouseStyle)
> house$RoofStyle<-factor(house$RoofStyle)
> house$RoofMatl<-factor(house$RoofMatl)
> house$Exterior1st<-factor(house$Exterior1st)
> house$Exterior2nd<-factor(house$Exterior2nd)
> house$ExterQual<-factor(house$ExterQual)
> house$ExterCond<-factor(house$ExterCond)
> house$Foundation<-factor(house$Foundation)
> house$Heating<-factor(house$Heating)
> house$HeatingQC<-factor(house$HeatingQC)
> house$CentralAir<-factor(house$CentralAir)
> house$KitchenQual<-factor(house$KitchenQual)
> house$Functional<-factor(house$Functional)
> house$PavedDrive<-factor(house$PavedDrive)
> house$SaleType<-factor(house$SaleType)
> house$SaleCondition<-factor(house$SaleCondition)
> str(house)
'data.frame':  2915 obs. of  82 variables:
 $ Id           : int  1 2 3 4 5 6 7 8 9 10 ...
 $ MSSubClass   : int  60 20 60 70 60 50 20 60 50 190 ...
 $ ScreenPorch  : int  0 0 0 0 0 0 0 0 0 0 ...
 $ PoolArea     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ PoolQC       : Factor w/ 4 levels "Ex","Fa","Gd",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ Fence        : Factor w/ 5 levels "GdPrv","GdWo",..: 5 5 5 5 5 3 5 5 5 5 ...
 $ MiscFeature  : Factor w/ 5 levels "Gar2","None",..: 2 2 2 2 2 4 2 4 2 2 ...
 $ MiscVal      : int  0 0 0 0 0 700 0 350 0 0 ...
 $ GarageType   : Factor w/ 7 levels "2Types","Attchd",..: 2 2 2 6 2 2 2 2 6 2 ...
 $ GarageYrBlt  : num  2003 1976 2001 1998 2000 ...
 $ GarageFinish : Factor w/ 4 levels "Fin","None","RFn",..: 3 3 3 4 3 4 3 3 4 3 ...
 $ GarageCars   : int  2 2 2 3 3 2 2 2 2 1 ...
 $ GarageArea   : int  548 460 608 642 836 480 636 484 468 205 ...
 $ GarageQual   : Factor w/ 6 levels "Ex","Fa","Gd",..: 6 6 6 6 6 6 6 6 2 3 ...
 $ GarageCond   : Factor w/ 6 levels "Ex","Fa","Gd",..: 6 6 6 6 6 6 6 6 6 6 ...
 $ PavedDrive   : Factor w/ 3 levels "N","P","Y": 3 3 3 3 3 3 3 3 3 3 ...
 $ WoodDeckSF   : int  0 298 0 0 192 40 255 235 90 0 ...
 $ OpenPorchSF  : int  61 0 42 35 84 30 57 204 0 4 ...
 $ EnclosedPorch: int  0 0 0 272 0 0 0 228 205 0 ...
 $ X3SsnPorch   : int  0 0 0 0 0 320 0 0 0 0 ...
 $ BsmtFullBath : int  1 0 1 1 1 1 1 1 0 1 ...
 $ BsmtHalfBath : int  0 1 0 0 0 0 0 0 0 0 ...
 $ FullBath     : int  2 2 2 1 2 1 2 2 2 1 ...
 $ HalfBath     : int  1 0 1 0 1 1 0 1 0 0 ...
 $ BedroomAbvGr : int  3 3 3 3 4 1 3 3 2 2 ...
 $ KitchenAbvGr : int  1 1 1 1 1 1 1 1 2 2 ...
 $ KitchenQual  : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 4 3 3 3 4 3 4 4 4 ...
 $ TotRmsAbvGrd : int  8 6 6 7 9 5 7 7 8 5 ...
 $ Functional   : Factor w/ 7 levels "Maj1","Maj2",..: 7 7 7 7 7 7 7 7 3 7 ...
 $ Fireplaces   : int  0 1 1 1 1 0 1 2 2 2 ...
 $ FireplaceQu  : Factor w/ 6 levels "Ex","Fa","Gd",..: 4 6 6 3 6 4 3 6 6 6 ...
 $ Heating      : Factor w/ 6 levels "Floor","GasA",..: 2 2 2 2 2 2 2 2 2 2 ...
 $ HeatingQC    : Factor w/ 5 levels "Ex","Fa","Gd",..: 1 1 1 3 1 1 1 1 3 1 ...
 $ CentralAir   : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...
```

```
 $ Electrical   : Factor w/ 6 levels "FuseA","FuseF",..: 6 6 6 6 6 6 6 6 2 6 ...
 $ X1stFlrSF    : int  856 1262 920 961 1145 796 1694 1107 1022 1077 ...
 $ X2ndFlrSF    : int  854 0 866 756 1053 566 0 983 752 0 ...
 $ LowQualFinSF : int  0 0 0 0 0 0 0 0 0 0 ...
 $ GrLivArea    : int  1710 1262 1786 1717 2198 1362 1694 2090 1774 1077 ...
 $ BsmtQual     : Factor w/ 5 levels "Ex","Fa","Gd",..: 3 3 3 5 3 3 1 3 5 5 ...
 $ BsmtCond     : Factor w/ 5 levels "Fa","Gd","None",..: 5 5 5 2 5 5 5 5 5 5 ...
 $ BsmtExposure : Factor w/ 5 levels "Av","Gd","Mn",..: 4 2 3 4 1 4 1 3 4 4 ...
 $ BsmtFinType1 : Factor w/ 7 levels "ALQ","BLQ","GLQ",..: 3 1 3 1 3 3 3 1 7 3 ...
 $ BsmtFinSF1   : int  706 978 486 216 655 732 1369 859 0 851 ...
 $ BsmtFinType2 : Factor w/ 7 levels "ALQ","BLQ","GLQ",..: 7 7 7 7 7 7 7 2 7 7 ...
 $ BsmtFinSF2   : int  0 0 0 0 0 0 0 32 0 0 ...
 $ BsmtUnfSF    : int  150 284 434 540 490 64 317 216 952 140 ...
 $ TotalBsmtSF  : int  856 1262 920 756 1145 796 1686 1107 952 991 ...
 $ Exterior2nd  : Factor w/ 16 levels "AsbShng","AsphShn",..: 14 9 14 16 14 14 14 7 16 9 .
..
 $ MasVnrType   : Factor w/ 4 levels "BrkCmn","BrkFace",..: 2 3 2 3 2 3 4 4 3 3 ...
 $ MasVnrArea   : num  196 0 162 0 350 0 186 240 0 0 ...
 $ ExterQual    : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 4 3 4 3 4 3 4 4 4 ...
 $ ExterCond    : Factor w/ 5 levels "Ex","Fa","Gd",..: 5 5 5 5 5 5 5 5 5 5 ...
 $ Foundation   : Factor w/ 6 levels "BrkTil","CBlock",..: 3 2 3 1 3 6 3 2 1 1 ...
 $ YearBuilt    : int  2003 1976 2001 1915 2000 1993 2004 1973 1931 1939 ...
 $ YearRemodAdd : int  2003 1976 2002 1970 2000 1995 2005 1973 1950 1950 ...
 $ RoofStyle    : Factor w/ 6 levels "Flat","Gable",..: 2 2 2 2 2 2 2 2 2 2 ...
 $ RoofMatl     : Factor w/ 7 levels "CompShg","Membran",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Exterior1st  : Factor w/ 15 levels "AsbShng","AsphShn",..: 13 9 13 14 13 13 13 7 4 9 ..
.
 $ BldgType     : Factor w/ 5 levels "1Fam","2fmCon",..: 1 1 1 1 1 1 1 1 1 2 ...
 $ HouseStyle   : Factor w/ 8 levels "1.5Fin","1.5Unf",..: 6 3 6 6 6 1 3 6 1 2 ...
 $ OverallQual  : int  7 6 7 7 8 5 8 7 7 5 ...
 $ OverallCond  : int  5 8 5 5 5 5 5 6 5 6 ...
 $ Neighborhood : Factor w/ 25 levels "Blmngtn","Blueste",..: 6 25 6 7 14 12 21 17 18 4 ..
.
 $ Condition1   : Factor w/ 9 levels "Artery","Feedr",..: 3 2 3 3 3 3 3 5 1 1 ...
 $ Condition2   : Factor w/ 8 levels "Artery","Feedr",..: 3 3 3 3 3 3 3 3 3 1 ...
 $ LandContour  : Factor w/ 4 levels "Bnk","HLS","Low",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ Utilities    : Factor w/ 2 levels "AllPub","NoSeWa": 1 1 1 1 1 1 1 1 1 1 ...
 $ LotConfig    : Factor w/ 5 levels "Corner","CulDSac",..: 5 3 5 1 3 5 5 1 5 1 ...
 $ LandSlope    : Factor w/ 3 levels "Gtl","Mod","Sev": 1 1 1 1 1 1 1 1 1 1 ...
 $ LotArea      : int  8450 9600 11250 9550 14260 14115 10084 10382 6120 7420 ...
 $ Street       : Factor w/ 2 levels "Grvl","Pave": 2 2 2 2 2 2 2 2 2 2 ...
 $ Alley        : Factor w/ 3 levels "Grvl","None",..: 2 2 2 2 2 2 2 2 2 2 ...
 $ LotShape     : Factor w/ 4 levels "IR1","IR2","IR3",..: 4 4 1 1 1 1 4 1 4 4 ...
 $ MSZoning     : Factor w/ 5 levels "C (all)","FV",..: 4 4 4 4 4 4 4 4 5 4 ...
 $ LotFrontage  : int  65 80 68 60 84 85 75 68 51 50 ...
 $ MoSold       : int  2 5 9 2 12 10 8 11 4 1 ...
 $ YrSold       : int  2008 2007 2008 2006 2008 2009 2007 2009 2008 2008 ...
 $ SaleType     : Factor w/ 9 levels "COD","Con","ConLD",..: 9 9 9 9 9 9 9 9 9 9 ...
 $ SaleCondition: Factor w/ 6 levels "Abnorml","AdjLand",..: 5 5 5 1 5 5 5 5 1 5 ...
 $ SalePrice    : int  208500 181500 223500 140000 250000 143000 307000 200000 129900 1180
00 ...
 $ isTrain      : num  1 1 1 1 1 1 1 1 1 1 ...
>
> #Taking all the column classes in one variable so as to seperate factors
from numerical variables.
> Column_classes <- sapply(names(house),function(x){class(house[[x]])})
> numeric_columns <-names(Column_classes[Column_classes != "factor"])
>
> #determining skew of each numeric variable
> skew <- sapply(numeric_columns,function(x){skewness(house[[x]],na.rm = T)})
>
> # Let us determine a threshold skewness and transform all variables above
the treshold.
> skew <- skew[skew > 0.75]
>
> # transform excessively skewed features with log(x + 1)
> for(x in names(skew))
+ {
+   house[[x]] <- log(house[[x]] + 1)
```

```
+ }
>
> #Train and test dataset creation
> train <- house[house$isTrain==1,]
> test <- house[house$isTrain==0,]
> smp_size <- floor(0.75 * nrow(train))
>
> ## setting the seed to make the partition reproducible
> set.seed(123)
> train_ind <- sample(seq_len(nrow(train)), size = smp_size)
> train_new <- train[train_ind, ]
> validate <- train[-train_ind, ]
> train_new <- subset(train_new,select=-c(Id,isTrain))
> validate <- subset(validate,select=-c(Id,isTrain))
> nrow(train_new)
[1] 1092
> nrow(validate)
[1] 364
> str(validate)
'data.frame':  364 obs. of  80 variables:
 $ MSSubClass   : num  3.04 4.11 3.04 4.11 3.83 ...
 $ ScreenPorch  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ PoolArea     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ PoolQC       : Factor w/ 4 levels "Ex","Fa","Gd",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ Fence        : Factor w/ 5 levels "GdPrv","GdWo",..: 5 5 5 5 1 3 5 5 5 5 ...
 $ MiscFeature  : Factor w/ 5 levels "Gar2","None",..: 2 2 2 2 2 2 2 2 2 2 ...
 $ MiscVal      : num  0 0 0 0 0 0 0 0 0 0 ...
 $ GarageType   : Factor w/ 7 levels "2Types","Attchd",..: 2 2 2 4 2 2 6 2 2 2 ...
 $ GarageYrBlt  : num  1976 2000 2004 2005 1930 ...
 $ GarageFinish : Factor w/ 4 levels "Fin","None","RFn",..: 3 3 3 3 4 4 4 3 1 3 ...
 $ GarageCars   : int  2 3 2 3 1 1 2 2 2 1 ...
 $ GarageArea   : int  460 836 636 853 280 270 576 484 498 308 ...
 $ GarageQual   : Factor w/ 6 levels "Ex","Fa","Gd",..: 6 6 6 6 6 6 6 6 6 6 ...
 $ GarageCond   : Factor w/ 6 levels "Ex","Fa","Gd",..: 6 6 6 6 6 6 6 6 6 6 ...
 $ PavedDrive   : Factor w/ 3 levels "N","P","Y": 3 3 3 3 1 3 3 3 3 2 ...
 $ WoodDeckSF   : num  5.7 5.26 5.55 5.48 0 ...
 $ OpenPorchSF  : num  0 4.44 4.06 5.04 0 ...
 $ EnclosedPorch: num  0 0 0 0 5.33 ...
 $ X3SsnPorch   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ BsmtFullBath : int  0 1 1 0 0 1 0 0 0 0 ...
 $ BsmtHalfBath : num  0.693 0 0 0 0 ...
 $ FullBath     : int  2 2 2 3 1 1 1 2 1 1 ...
 $ HalfBath     : int  0 1 0 1 0 0 0 0 0 1 ...
 $ BedroomAbvGr : int  3 4 3 4 3 3 3 3 3 2 ...
 $ KitchenAbvGr : num  0.693 0.693 0.693 0.693 0.693 ...
 $ KitchenQual  : Factor w/ 4 levels "Ex","Fa","Gd",..: 4 3 3 3 3 3 3 3 4 4 ...
 $ TotRmsAbvGrd : int  6 9 7 9 6 6 5 7 5 5 ...
 $ Functional   : Factor w/ 7 levels "Maj1","Maj2",..: 7 7 7 7 7 7 7 7 7 7 ...
 $ Fireplaces   : int  1 1 1 1 1 1 0 0 1 2 ...
 $ FireplaceQu  : Factor w/ 6 levels "Ex","Fa","Gd",..: 6 6 3 3 3 6 4 4 6 3 ...
 $ Heating      : Factor w/ 6 levels "Floor","GasA",..: 2 2 2 2 2 2 2 2 2 2 ...
 $ HeatingQC    : Factor w/ 5 levels "Ex","Fa","Gd",..: 1 1 1 1 1 1 5 1 3 3 ...
 $ CentralAir   : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...
 $ Electrical   : Factor w/ 6 levels "FuseA","FuseF",..: 6 6 6 6 2 6 6 6 6 6 ...
 $ X1stFlrSF    : num  7.14 7.04 7.44 7.06 7.01 ...
 $ X2ndFlrSF    : num  0 6.96 0 7.11 0 ...
 $ LowQualFinSF : num  0 0 0 0 0 0 0 0 0 0 ...
 $ GrLivArea    : num  7.14 7.7 7.44 7.77 7.01 ...
 $ BsmtQual     : Factor w/ 5 levels "Ex","Fa","Gd",..: 3 3 1 1 5 5 5 1 5 5 ...
 $ BsmtCond     : Factor w/ 5 levels "Fa","Gd","None",..: 5 5 5 5 5 5 5 5 5 5 ...
 $ BsmtExposure : Factor w/ 5 levels "Av","Gd","Mn",..: 2 1 1 1 4 3 3 1 4 2 ...
 $ BsmtFinType1 : Factor w/ 7 levels "ALQ","BLQ","GLQ",..: 1 3 3 7 7 6 2 7 6 2 ...
 $ BsmtFinSF1   : num  6.89 6.49 7.22 0 0 ...
 $ BsmtFinType2 : Factor w/ 7 levels "ALQ","BLQ","GLQ",..: 7 7 7 7 7 1 6 7 7 7 ...
 $ BsmtFinSF2   : num  0 0 0 0 0 ...
 $ BsmtUnfSF    : num  5.65 6.2 5.76 7.06 6.46 ...
 $ TotalBsmtSF  : int  1262 1145 1686 1158 637 1060 900 1234 1297 1350 ...
 $ Exterior2nd  : Factor w/ 16 levels "AsbShng","AsphShn",..: 9 14 14 14 15 11 15 14 15 14
...
```

```
 $ MasVnrType   : Factor w/ 4 levels "BrkCmn","BrkFace",..: 3 2 4 2 3 3 3 3 2 3 ...
 $ MasVnrArea   : num  0 5.86 5.23 5.94 0 ...
 $ ExterQual    : Factor w/ 4 levels "Ex","Fa","Gd",..: 4 3 3 3 4 4 4 3 4 4 ...
 $ ExterCond    : Factor w/ 5 levels "Ex","Fa","Gd",..: 5 5 5 5 5 3 5 5 5 3 ...
 $ Foundation   : Factor w/ 6 levels "BrkTil","CBlock",..: 2 3 3 3 3 2 2 3 2 2 ...
 $ YearBuilt    : int  1976 2000 2004 2005 1930 1968 1951 2007 1954 1959 ...
 $ YearRemodAdd : int  1976 2000 2005 2006 1950 2001 2000 2007 1990 1959 ...
 $ RoofStyle    : Factor w/ 6 levels "Flat","Gable",..: 2 2 2 2 2 2 2 2 4 2 ...
 $ RoofMatl     : Factor w/ 7 levels "CompShg","Membran",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Exterior1st  : Factor w/ 15 levels "AsbShng","AsphShn",..: 9 13 13 13 14 10 14 13 14 13
...
 $ BldgType     : Factor w/ 5 levels "1Fam","2fmCon",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ HouseStyle   : Factor w/ 8 levels "1.5Fin","1.5Unf",..: 3 6 3 6 2 3 3 3 3 3 ...
 $ OverallQual  : int  6 8 8 8 7 5 5 8 5 5 ...
 $ OverallCond  : int  8 5 5 5 7 8 7 5 6 6 ...
 $ Neighborhood : Factor w/ 25 levels "Blmngtn","Blueste",..: 25 14 21 16 10 19 13 6 13 24
...
 $ Condition1   : Factor w/ 9 levels "Artery","Feedr",..: 2 3 3 3 3 3 3 3 3 3 ...
 $ Condition2   : Factor w/ 8 levels "Artery","Feedr",..: 3 3 3 3 3 3 3 3 3 3 ...
 $ LandContour  : Factor w/ 4 levels "Bnk","HLS","Low",..: 4 4 4 4 1 4 4 4 4 4 ...
 $ Utilities    : Factor w/ 2 levels "AllPub","NoSeWa": 1 1 1 1 1 1 1 1 1 1 ...
 $ LotConfig    : Factor w/ 5 levels "Corner","CulDSac",..: 3 3 5 1 5 5 1 1 5 5 ...
 $ LandSlope    : Factor w/ 3 levels "Gtl","Mod","Sev": 1 1 1 1 1 1 1 1 1 1 ...
 $ LotArea      : num  9.17 9.57 9.22 9.56 8.92 ...
 $ Street       : Factor w/ 2 levels "Grvl","Pave": 2 2 2 2 2 2 2 2 2 2 ...
 $ Alley        : Factor w/ 3 levels "Grvl","None",..: 2 2 2 2 1 2 2 2 2 2 ...
 $ LotShape     : Factor w/ 4 levels "IR1","IR2","IR3",..: 4 1 4 1 4 1 4 4 4 4 ...
 $ MSZoning     : Factor w/ 5 levels "C (all)","FV",..: 4 4 4 4 5 4 4 4 4 4 ...
 $ LotFrontage  : num  4.39 4.44 4.33 4.62 4.06 ...
 $ MoSold       : int  5 12 8 11 6 5 5 1 10 7 ...
 $ YrSold       : int  2007 2008 2007 2006 2007 2010 2010 2008 2009 2007 ...
 $ SaleType     : Factor w/ 9 levels "COD","Con","ConLD",..: 9 9 9 7 9 9 9 9 9 9 ...
 $ SaleCondition: Factor w/ 6 levels "Abnorml","AdjLand",..: 5 5 5 6 5 5 5 5 5 5 ...
 $ SalePrice    : num  12.1 12.4 12.6 12.7 11.8 ...
```

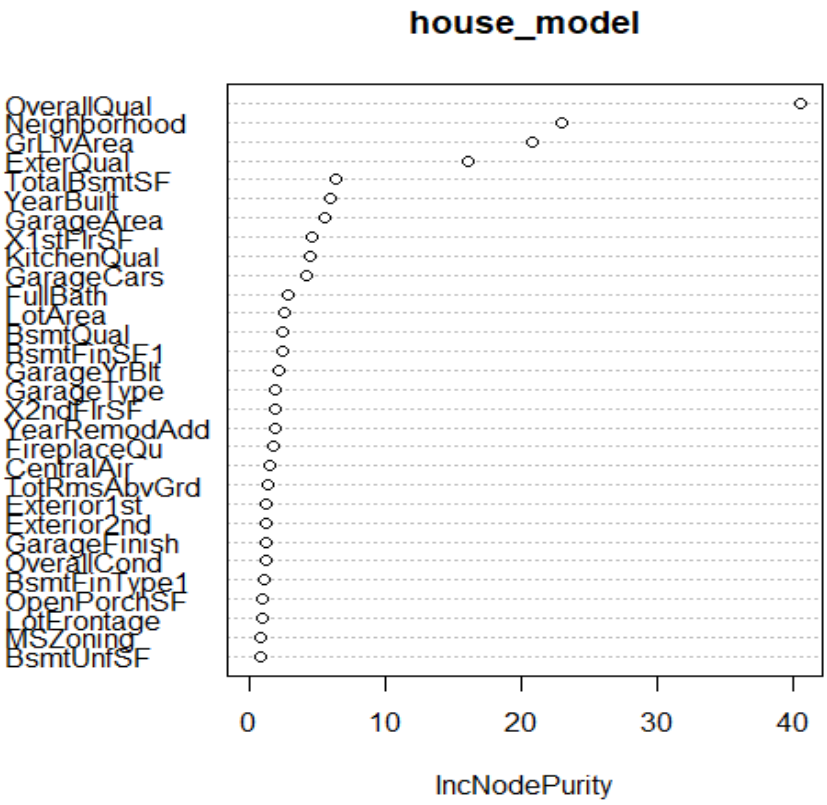
Final Prediction

Although our base dataset has been prepared for modelling, we still just need to define the dataset to be used for training the model. So the first thing I do in the above snippet is define the variables and observations required to train the model. I do this by merging the data with the original train.csv dataset

After doing this I elect to keep out a portion of the data to use as an out of sample test set, so I can see how well the trained model will perform on a randomly split portion of the observations that were not used to train the model. Doing this kind of test should give me an idea of how well the model can predict never-before-seen (out of sample) cases. And we will see the results of that test shortly.

```
> #Build the model
> library(randomForest)
> house_model <- randomForest(SalePrice~.,data = train_new)
> #Variable importance
> importance    <- importance(house_model)
> varImpPlot(house_model)
```

By viewing the importance from below curve we can see that SalePrice is much affected by Overall Quality of house, Neighborhood Location, GrLivArea and ExterQual rest does not affect that much.



Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set. A random forest model has now been created and this can be used to make predictions;

```
> #Final Prediction
> ##Predict using the test set
>
> prediction <- predict(house_model,test)
>
> ##Evaluation RMSE function
>
> RMSE <- function(x,y){
+   a <- sqrt(sum((log(x)-log(y))^2)/length(y))
+   return(a)
+ }
>
> ##RMSE
> RMSE1 <- RMSE(prediction, validate$SalePrice)
Warning message:
In log(x) - log(y) :
  longer object length is not a multiple of shorter object length
> RMSE1
[1] 0.08843116


> RMSE1 <- round(RMSE1, digits = 5)
```

We find the saleprice for test dataset and also find the RMSE (Root mean square error) for our model which comes out to be 0.08843116 which is very less. Our scores the root mean square error (RMSE) of our predictions, which is a metric for describing the difference between the observed values and our predicted values for Sale Price; scores closer to zero are better.

Now I created an output file of my prediction named as House_Price_Prediction_Abhishak.csv

```
> #Output file
>
> prediction[which(is.na(prediction))] <- mean(prediction,na.rm=T)
> submit <- data.frame(Id=test$Id,MSSubClass=test$MSSubClass,ScreenPorch=test$ScreenPorch,
PoolArea=test$PoolArea,
+                      PoolQC=test$PoolQC, Fence=test$Fence, MiscFeature=test$MiscFeature,
MiscVal=test$MiscVal,
+                      GarageType=test$GarageType,GarageYrBlt=test$GarageYrBlt,GarageFinis
h=test$GarageFinish,
+                      GarageCars=test$GarageCars, GarageArea=test$GarageArea, GarageQual=
test$GarageQual,
+                      GarageCond=test$GarageCond, PavedDrive=test$PavedDrive,WoodDeckSF=t
est$WoodDeckSF,
+                      OpenPorchSF=test$OpenPorchSF, EnclosedPorch=test$EnclosedPorch,X3Ss
nPorch=test$X3SsnPorch,
+                      BsmtFullBath=test$BsmtFullBath, BsmtHalfBath=test$BsmtHalfBath,Full
Bath=test$FullBath,
+                      HalfBath=test$HalfBath, BedroomAbvGr=test$BedroomAbvGr,KitchenAbvGr
=test$KitchenAbvGr,
+                      KitchenQual=test$KitchenQual, TotRmsAbvGrd=test$TotRmsAbvGrd, Funct
ional=test$Functional,
+                      Fireplaces=test$Fireplaces,FireplaceQu=test$FireplaceQu, Heating=te
st$Heating,
+                      HeatingQC=test$HeatingQC, CentralAir=test$CentralAir, Electrical=te
st$Electrical,
+                      X1stFlrSF=test$X1stFlrSF, X2ndFlrSF=test$X2ndFlrSF,LowQualFinSF=tes
t$LowQualFinSF,
+                      GrLivArea=test$GrLivArea,BsmtQual=test$BsmtQual, BsmtCond=test$Bsmt
Cond,
+                      BsmtExposure=test$BsmtExposure,BsmtFinType1=test$BsmtFinType1, Bsmt
FinSF1=test$BsmtFinSF1,
+                      BsmtFinType2=test$BsmtFinType2,BsmtFinSF2=test$BsmtFinSF2, BsmtUnfS
F=test$BsmtUnfSF,
+                      TotalBsmtSF=test$TotalBsmtSF,Exterior2nd=test$Exterior2nd,MasVnrTyp
e=test$MasVnrType,
+                      MasVnrArea=test$MasVnrArea, ExterQual=test$ExterQual, ExterCond=tes
t$ExterCond,
+                      Foundation=test$Foundation,YearBuilt=test$YearBuilt,YearRemodAdd=te
st$YearRemodAdd,
+                      RoofStyle=test$RoofStyle, RoofMatl=test$RoofMatl,Exterior1st=test$E
xterior1st,
+                      BldgType=test$BldgType,HouseStyle=test$HouseStyle, OverallQual=test
$OverallQual,
+                      OverallCond=test$OverallCond,Neighborhood=test$Neighborhood, Condit
ion1=test$Condition1,
+                      Condition2=test$Condition2,LandContour=test$LandContour, Utilities=
test$Utilities,
+                      LotConfig=test$LotConfig,LandSlope=test$LandSlope,LotArea=test$LotA
rea, Street=test$Street,
+                      Alley=test$Alley, LotShape=test$LotShape,MSZoning=test$MSZoning, Lo
tFrontage=test$LotFrontage,
```

```
+                       MoSold=test$MoSold, YrSold=test$YrSold, SaleType=test$SaleType, Sal
eCondition=test$SaleCondition,
+                       SalePrice=prediction)
> write.csv(submit,file="House_Price_Prediction_Abhishak.csv",row.names=F)
```

Predictors related to square footage (Area), quality (different Quality measures), and age (Year Built) have the strongest impact on model's predictions. The variables seen as most important or as strongest predictors through our models were those related to square footage, the age and condition of the home, the neighborhood where the house was located, the city zone where the house was located, and the year the house was sold.