

Prediction of sale prices of house

Problem Statement

To predict the sale prices of houses

Project Team

1. Tejaswini Nutalapati
2. Aditi Bhargava

About the Data

Ask a home buyer to describe their dream house, and they probably won't begin with the height of the basement ceiling or the proximity to an east-west railroad. But this dataset proves that much more influences price negotiations than the number of bedrooms or a white-picket fence. With 79 explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, this dataset allows us to predict the final price of each home.

The Ames Housing dataset was compiled by Dean De Cock for use in data science education. It's an incredible alternative for data scientists looking for a modernized and expanded version of the often-cited Boston Housing dataset.

Data Source: <https://www.kaggle.com/c/house-prices-advanced-regression-techniques>

Data Cleaning and Exploratory data Analysis

```
#Loading the Libraries
library(reshape2)
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(ggplot2)
library(lattice)
library(caret)
```

```
library(scales)
library(dummies)

## dummies-1.5.6 provided by Decision Patterns

library(fmsb)

## Registered S3 methods overwritten by 'fmsb':
##   method      from
##   print.roc    pROC
##   plot.roc     pROC

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:ggplot2':
##
##   margin

## The following object is masked from 'package:dplyr':
##
##   combine

library(DescTools)

##
## Attaching package: 'DescTools'

## The following objects are masked from 'package:fmsb':
##
##   CronbachAlpha, VIF

## The following objects are masked from 'package:caret':
##
##   MAE, RMSE

library(outliers)

##
## Attaching package: 'outliers'

## The following object is masked from 'package:randomForest':
##
##   outlier

library(VIM)

## Loading required package: colorspace
```

```
## Loading required package: grid

## VIM is ready to use.

## Suggestions and bug-reports can be submitted at:
https://github.com/statistikat/VIM/issues

##
## Attaching package: 'VIM'

## The following object is masked from 'package:datasets':
##
##      sleep

library(GGally)

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg      ggplot2

library(corrplot)

## corrplot 0.84 loaded
```

```
# Loading the dataset
```

```
list.files("../input")

## character(0)

Train<-read.csv("/Users/tejaswininutalapati/Documents/Multivariate
Analysis/Project/DataSet/train.csv")
Test<-read.csv("/Users/tejaswininutalapati/Documents/Multivariate
Analysis/Project/DataSet/test.csv")
```

```
# Add sale price new column in test dataset
```

```
Test["SalePrice"] <- NA
```

```
# Let's explore the 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 ...
```

```

## $ 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 NA 51 50 ...
## $ 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/ 2 levels "Grvl","Pave": NA NA NA NA NA NA NA
NA NA NA ...
## $ LotShape      : Factor w/ 4 levels "IR1","IR2","IR3",...: 4 4 1 1 1 1 4 1
4 4 ...
## $ 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 ...
## $ 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 ...
## $ 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 ...
## $ 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/ 8 levels "ClyTile","CompShg",...: 2 2 2 2 2 2 2
2 2 2 ...
## $ Exterior1st   : Factor w/ 15 levels "AsbShng","AsphShn",...: 13 9 13 14
13 13 13 7 4 9 ...
## $ 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    : int    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 ...
## $ BsmtQual        : Factor w/ 4 levels "Ex","Fa","Gd",...: 3 3 3 4 3 3 1 3 4
4 ...
## $ BsmtCond        : Factor w/ 4 levels "Fa","Gd","Po",...: 4 4 4 2 4 4 4 4 4
4 ...
## $ BsmtExposure    : Factor w/ 4 levels "Av","Gd","Mn",...: 4 2 3 4 1 4 1 3 4
4 ...
## $ BsmtFinType1    : Factor w/ 6 levels "ALQ","BLQ","GLQ",...: 3 1 3 1 3 3 3 1
6 3 ...
## $ BsmtFinSF1      : int    706 978 486 216 655 732 1369 859 0 851 ...
## $ BsmtFinType2    : Factor w/ 6 levels "ALQ","BLQ","GLQ",...: 6 6 6 6 6 6 6 2
6 6 ...
## $ 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 ...
## $ 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 ...
## $ Electrical      : Factor w/ 5 levels "FuseA","FuseF",...: 5 5 5 5 5 5 5 5 2
5 ...
## $ 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
...
## $ 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 3 7
...
## $ Fireplaces      : int    0 1 1 1 1 0 1 2 2 2 ...
## $ FireplaceQu     : Factor w/ 5 levels "Ex","Fa","Gd",...: NA 5 5 3 5 NA 3 5
5 5 ...
## $ GarageType      : Factor w/ 6 levels "2Types","Attchd",...: 2 2 2 6 2 2 2 2
6 2 ...
## $ GarageYrBlt     : int    2003 1976 2001 1998 2000 1993 2004 1973 1931 1939
...
## $ GarageFinish    : Factor w/ 3 levels "Fin","RFn","Unf": 2 2 2 3 2 3 2 2 3
2 ...
## $ 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/ 5 levels "Ex","Fa","Gd",...: 5 5 5 5 5 5 5 5 2
3 ...
## $ GarageCond : Factor w/ 5 levels "Ex","Fa","Gd",...: 5 5 5 5 5 5 5 5 5
5 ...
## $ PavedDrive : Factor w/ 3 levels "N","P","Y": 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 ...
## $ 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/ 3 levels "Ex","Fa","Gd": NA NA NA NA NA NA NA
NA NA NA ...
## $ Fence : Factor w/ 4 levels "GdPrv","GdWo",...: NA NA NA NA NA 3
NA NA NA NA ...
## $ MiscFeature : Factor w/ 4 levels "Gar2","Othr",...: NA NA NA NA NA 3 NA
3 NA NA ...
## $ MiscVal : int 0 0 0 0 0 700 0 350 0 0 ...
## $ 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 118000 ...

```

```
dim(Test)
```

```
## [1] 1459 81
```

```
str(Test)
```

```

## 'data.frame': 1459 obs. of 81 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 ...
## $ MSZoning : Factor w/ 5 levels "C (all)","FV",...: 3 4 4 4 4 4 4 4 4
4 ...
## $ LotFrontage : int 80 81 74 78 43 75 NA 63 85 70 ...
## $ LotArea : int 11622 14267 13830 9978 5005 10000 7980 8402 10176
8400 ...
## $ Street : Factor w/ 2 levels "Grvl","Pave": 2 2 2 2 2 2 2 2 2 2
...
## $ Alley : Factor w/ 2 levels "Grvl","Pave": NA NA NA NA NA NA NA
NA NA NA ...
## $ LotShape : Factor w/ 4 levels "IR1","IR2","IR3",...: 4 1 1 1 1 1 1 1
4 4 ...
## $ LandContour : Factor w/ 4 levels "Bnk","HLS","Low",...: 4 4 4 4 2 4 4 4
4 4 ...

```

```

## $ Utilities      : Factor w/ 1 level "AllPub": 1 1 1 1 1 1 1 1 1 1 ...
## $ LotConfig      : Factor w/ 5 levels "Corner","CulDSac",...: 5 1 5 5 5 1 5
5 5 1 ...
## $ LandSlope      : Factor w/ 3 levels "Gtl","Mod","Sev": 1 1 1 1 1 1 1 1 1
1 ...
## $ Neighborhood  : Factor w/ 25 levels "Blmngtn","Blueste",...: 13 13 9 9 22
9 9 9 9 13 ...
## $ Condition1     : Factor w/ 9 levels "Artery","Feedr",...: 2 3 3 3 3 3 3 3
3 3 ...
## $ Condition2     : Factor w/ 5 levels "Artery","Feedr",...: 3 3 3 3 3 3 3 3
3 3 ...
## $ BldgType       : Factor w/ 5 levels "1Fam","2fmCon",...: 1 1 1 1 5 1 1 1 1
1 ...
## $ HouseStyle     : Factor w/ 7 levels "1.5Fin","1.5Unf",...: 3 3 5 5 3 5 3 5
3 3 ...
## $ OverallQual    : int    5 6 5 6 8 6 6 6 7 4 ...
## $ OverallCond    : int    6 6 5 6 5 5 7 5 5 5 ...
## $ 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      : Factor w/ 6 levels "Flat","Gable",...: 2 4 2 2 2 2 2 2 2
2 ...
## $ RoofMatl       : Factor w/ 4 levels "CompShg","Tar&Grv",...: 1 1 1 1 1 1 1 1
1 1 1 ...
## $ Exterior1st    : Factor w/ 13 levels "AsbShng","AsphShn",...: 11 12 11 11
7 7 7 11 7 9 ...
## $ Exterior2nd    : Factor w/ 15 levels "AsbShng","AsphShn",...: 13 14 13 13
7 7 7 13 7 10 ...
## $ MasVnrType     : Factor w/ 4 levels "BrkCmn","BrkFace",...: 3 2 3 2 3 3 3 3
3 3 3 ...
## $ MasVnrArea     : int    0 108 0 20 0 0 0 0 0 0 ...
## $ ExterQual      : Factor w/ 4 levels "Ex","Fa","Gd",...: 4 4 4 4 3 4 4 4 4
4 ...
## $ ExterCond      : Factor w/ 5 levels "Ex","Fa","Gd",...: 5 5 5 5 5 5 3 5 5
5 ...
## $ Foundation     : Factor w/ 6 levels "BrkTil","CBlock",...: 2 2 3 3 3 3 3 3
3 2 ...
## $ BsmtQual       : Factor w/ 4 levels "Ex","Fa","Gd",...: 4 4 3 4 3 3 3 3 3
4 ...
## $ BsmtCond       : Factor w/ 4 levels "Fa","Gd","Po",...: 4 4 4 4 4 4 4 4 4
4 ...
## $ BsmtExposure   : Factor w/ 4 levels "Av","Gd","Mn",...: 4 4 4 4 4 4 4 4 2
4 ...
## $ BsmtFinType1   : Factor w/ 6 levels "ALQ","BLQ","GLQ",...: 5 1 3 3 1 6 1 6
3 1 ...
## $ BsmtFinSF1     : int    468 923 791 602 263 0 935 0 637 804 ...
## $ BsmtFinType2   : Factor w/ 6 levels "ALQ","BLQ","GLQ",...: 4 6 6 6 6 6 6 6 6
6 5 ...
## $ 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 ...
## $ Heating        : Factor w/ 4 levels "GasA","GasW",...: 1 1 1 1 1 1 1 1 1 1
...
## $ HeatingQC      : Factor w/ 5 levels "Ex","Fa","Gd",...: 5 5 3 1 1 3 1 3 3
5 ...
## $ CentralAir     : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 ...
## $ Electrical     : Factor w/ 4 levels "FuseA","FuseF",...: 4 4 4 4 4 4 4 4 4
4 ...
## $ 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 ...
## $ 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     : Factor w/ 4 levels "Ex","Fa","Gd",...: 4 3 4 3 3 4 4 4 3
4 ...
## $ TotRmsAbvGrd    : int    5 6 6 7 5 7 6 7 5 4 ...
## $ Functional      : Factor w/ 7 levels "Maj1","Maj2",...: 7 7 7 7 7 7 7 7 7 7
...
## $ Fireplaces      : int    0 0 1 1 0 1 0 1 1 0 ...
## $ FireplaceQu     : Factor w/ 5 levels "Ex","Fa","Gd",...: NA NA 5 3 NA 5 NA
3 4 NA ...
## $ GarageType      : Factor w/ 6 levels "2Types","Attchd",...: 2 2 2 2 2 2 2 2 2
2 2 ...
## $ GarageYrBlt     : int   1961 1958 1997 1998 1992 1993 1992 1998 1990 1970
...
## $ GarageFinish    : Factor w/ 3 levels "Fin","RFn","Unf": 3 3 1 1 2 1 1 1 3
1 ...
## $ 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      : Factor w/ 4 levels "Fa","Gd","Po",...: 4 4 4 4 4 4 4 4 4
4 ...
## $ GarageCond      : Factor w/ 5 levels "Ex","Fa","Gd",...: 5 5 5 5 5 5 5 5 5
5 ...
## $ PavedDrive      : Factor w/ 3 levels "N","P","Y": 3 3 3 3 3 3 3 3 3 3 ...
## $ 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 ...
## $ 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          : Factor w/ 2 levels "Ex","Gd": NA NA NA NA NA NA NA NA NA
NA ...
## $ Fence           : Factor w/ 4 levels "GdPrv","GdWo",...: 3 NA 3 NA NA NA 1
NA NA 3 ...

```



```
## $ MiscFeature : Factor w/ 3 levels "Gar2","Othr",...: NA 1 NA NA NA NA 3
NA NA NA ...
## $ MiscVal      : int  0 12500 0 0 0 0 500 0 0 0 ...
## $ 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     : 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 5 5 5 5
5 5 5 ...
## $ SalePrice    : logi  NA NA NA NA NA NA NA ...
```

Combining the dataset

```
Test$SalePrice <- -1
df <- rbind(Train,Test)
str(df)

## '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 ...
## $ 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 NA 51 50 ...
## $ 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/ 2 levels "Grvl","Pave": NA NA NA NA NA NA NA
NA NA NA ...
## $ LotShape    : Factor w/ 4 levels "IR1","IR2","IR3",...: 4 4 1 1 1 1 4 1
4 4 ...
## $ 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 ...
## $ 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 ...
## $ 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 ...
## $ 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/ 8 levels "ClyTile","CompShg",...: 2 2 2 2 2 2 2
2 2 2 ...
## $ Exterior1st : Factor w/ 15 levels "AsbShng","AsphShn",...: 13 9 13 14
13 13 13 7 4 9 ...
## $ 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 : int 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 ...
## $ BsmtQual : Factor w/ 4 levels "Ex","Fa","Gd",...: 3 3 3 4 3 3 1 3 4
4 ...
## $ BsmtCond : Factor w/ 4 levels "Fa","Gd","Po",...: 4 4 4 2 4 4 4 4 4
4 ...
## $ BsmtExposure : Factor w/ 4 levels "Av","Gd","Mn",...: 4 2 3 4 1 4 1 3 4
4 ...
## $ BsmtFinType1 : Factor w/ 6 levels "ALQ","BLQ","GLQ",...: 3 1 3 1 3 3 3 1
6 3 ...
## $ BsmtFinSF1 : int 706 978 486 216 655 732 1369 859 0 851 ...
## $ BsmtFinType2 : Factor w/ 6 levels "ALQ","BLQ","GLQ",...: 6 6 6 6 6 6 6 2
6 6 ...
## $ 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 ...
## $ 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 ...
## $ Electrical : Factor w/ 5 levels "FuseA","FuseF",...: 5 5 5 5 5 5 5 5 2
5 ...
## $ 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
...

```

```

## $ 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 3 7
...
## $ Fireplaces    : int 0 1 1 1 1 0 1 2 2 2 ...
## $ FireplaceQu   : Factor w/ 5 levels "Ex","Fa","Gd",...: NA 5 5 3 5 NA 3 5
5 5 ...
## $ GarageType    : Factor w/ 6 levels "2Types","Attchd",...: 2 2 2 6 2 2 2 2
6 2 ...
## $ GarageYrBlt   : int 2003 1976 2001 1998 2000 1993 2004 1973 1931 1939
...
## $ GarageFinish  : Factor w/ 3 levels "Fin","RFn","Unf": 2 2 2 3 2 3 2 2 3
2 ...
## $ 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/ 5 levels "Ex","Fa","Gd",...: 5 5 5 5 5 5 5 5 2
3 ...
## $ GarageCond    : Factor w/ 5 levels "Ex","Fa","Gd",...: 5 5 5 5 5 5 5 5 5
5 ...
## $ 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 ...
## $ 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/ 3 levels "Ex","Fa","Gd": NA NA NA NA NA NA NA
NA NA NA ...
## $ Fence         : Factor w/ 4 levels "GdPrv","GdWo",...: NA NA NA NA NA 3
NA NA NA NA ...
## $ MiscFeature    : Factor w/ 4 levels "Gar2","Othr",...: NA NA NA NA NA 3 NA
3 NA NA ...
## $ MiscVal       : int 0 0 0 0 0 700 0 350 0 0 ...
## $ 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     : num 208500 181500 223500 140000 250000 ...

```

summary(df)

```
##           Id           MSSubClass           MSZoning           LotFrontage
##  Min.      : 1.0      Min.      : 20.00      C (all): 25      Min.      : 21.00
## 1st Qu.: 730.5      1st Qu.: 20.00      FV       : 139      1st Qu.: 59.00
## Median :1460.0      Median : 50.00      RH       : 26      Median : 68.00
## Mean    :1460.0      Mean    : 57.14      RL       :2265      Mean    : 69.31
## 3rd Qu.:2189.5      3rd Qu.: 70.00      RM       : 460      3rd Qu.: 80.00
## Max.    :2919.0      Max.    :190.00      NA's     : 4      Max.    :313.00
##                                     NA's     :486
##           LotArea           Street           Alley           LotShape           LandContour
Utilities
##  Min.      : 1300      Grvl: 12      Grvl: 120      IR1: 968      Bnk: 117
AllPub:2916
## 1st Qu.: 7478      Pave:2907      Pave: 78      IR2: 76      HLS: 120      NoSeWa:
1
## Median : 9453                                     NA's:2721      IR3: 16      Low: 60      NA's :
2
## Mean    : 10168                                     Reg:1859      Lvl:2622
## 3rd Qu.: 11570
## Max.    :215245
##
##           LotConfig           LandSlope           Neighborhood           Condition1           Condition2
## Corner : 511      Gtl:2778      NAmes : 443      Norm :2511      Norm :2889
## CulDSac: 176      Mod: 125      CollgCr: 267      Feedr : 164      Feedr : 13
## FR2     : 85      Sev: 16      OldTown: 239      Artery : 92      Artery : 5
## FR3     : 14                                     Edwards: 194      RRAn : 50      PosA : 4
## Inside :2133      Somerst: 182      PosN : 39      PosN : 4
##                                     NridgHt: 166      RRAe : 28      RRNn : 2
##                                     (Other):1428      (Other): 35      (Other): 2
##           BldgType           HouseStyle           OverallQual           OverallCond           YearBuilt
## 1Fam :2425      1Story :1471      Min. : 1.000      Min. :1.000      Min.
:1872
## 2fmCon: 62      2Story : 872      1st Qu.: 5.000      1st Qu.:5.000      1st
Qu.:1954
## Duplex: 109      1.5Fin : 314      Median : 6.000      Median :5.000      Median
:1973
## Twnhs : 96      SLvl : 128      Mean : 6.089      Mean :5.565      Mean
:1971
## TwnhsE: 227      SFoyer : 83      3rd Qu.: 7.000      3rd Qu.:6.000      3rd
Qu.:2001
##                                     2.5Unf : 24      Max. :10.000      Max. :9.000      Max.
:2010
##                                     (Other): 27
##           YearRemodAdd           RoofStyle           RoofMatl           Exterior1st           Exterior2nd
##  Min.      :1950      Flat : 20      CompShg:2876      VinylSd:1025      VinylSd:1014
## 1st Qu.:1965      Gable :2310      Tar&Grv: 23      MetalSd: 450      MetalSd: 447
## Median :1993      Gambrel: 22      WdShake: 9      HdBoard: 442      HdBoard: 406
## Mean :1984      Hip : 551      WdShngl: 7      Wd Sdng: 411      Wd Sdng: 391
## 3rd Qu.:2004      Mansard: 11      ClyTile: 1      Plywood: 221      Plywood: 270
```

```

## Max. :2010 Shed : 5 Membran: 1 (Other): 369 (Other): 390
## (Other): 2 NA's : 1 NA's : 1
## MasVnrType MasVnrArea ExterQual ExterCond Foundation
BsmtQual
## BrkCmn : 25 Min. : 0.0 Ex: 107 Ex: 12 BrkTil: 311 Ex :
258
## BrkFace: 879 1st Qu.: 0.0 Fa: 35 Fa: 67 CBlock:1235 Fa :
88
## None :1742 Median : 0.0 Gd: 979 Gd: 299 PConc :1308 Gd
:1209
## Stone : 249 Mean : 102.2 TA:1798 Po: 3 Slab : 49 TA
:1283
## NA's : 24 3rd Qu.: 164.0 TA:2538 Stone : 11 NA's:
81
## Max. :1600.0 Wood : 5
## NA's :23
## BsmtCond BsmtExposure BsmtFinType1 BsmtFinSF1 BsmtFinType2
## Fa : 104 Av : 418 ALQ :429 Min. : 0.0 ALQ : 52
## Gd : 122 Gd : 276 BLQ :269 1st Qu.: 0.0 BLQ : 68
## Po : 5 Mn : 239 GLQ :849 Median : 368.5 GLQ : 34
## TA :2606 No :1904 LwQ :154 Mean : 441.4 LwQ : 87
## NA's: 82 NA's: 82 Rec :288 3rd Qu.: 733.0 Rec : 105
## Unf :851 Max. :5644.0 Unf :2493
## NA's: 79 NA's :1 NA's: 80
## BsmtFinSF2 BsmtUnfSF TotalBsmtSF Heating
HeatingQC
## Min. : 0.00 Min. : 0.0 Min. : 0.0 Floor: 1 Ex:1493
## 1st Qu.: 0.00 1st Qu.: 220.0 1st Qu.: 793.0 GasA :2874 Fa: 92
## Median : 0.00 Median : 467.0 Median : 989.5 GasW : 27 Gd: 474
## Mean : 49.58 Mean : 560.8 Mean :1051.8 Grav : 9 Po: 3
## 3rd Qu.: 0.00 3rd Qu.: 805.5 3rd Qu.:1302.0 OthW : 2 TA: 857
## Max. :1526.00 Max. :2336.0 Max. :6110.0 Wall : 6
## NA's :1 NA's :1 NA's :1
## CentralAir Electrical X1stFlrSF X2ndFlrSF LowQualFinSF
## N: 196 FuseA: 188 Min. : 334 Min. : 0.0 Min. : 0.000
## Y:2723 FuseF: 50 1st Qu.: 876 1st Qu.: 0.0 1st Qu.: 0.000
## FuseP: 8 Median :1082 Median : 0.0 Median : 0.000
## Mix : 1 Mean :1160 Mean : 336.5 Mean : 4.694
## SBrkr:2671 3rd Qu.:1388 3rd Qu.: 704.0 3rd Qu.: 0.000
## NA's : 1 Max. :5095 Max. :2065.0 Max. :1064.000
##
## GrLivArea BsmtFullBath BsmtHalfBath FullBath
## Min. : 334 Min. :0.0000 Min. :0.00000 Min. :0.000
## 1st Qu.:1126 1st Qu.:0.0000 1st Qu.:0.00000 1st Qu.:1.000
## Median :1444 Median :0.0000 Median :0.00000 Median :2.000
## Mean :1501 Mean :0.4299 Mean :0.06136 Mean :1.568
## 3rd Qu.:1744 3rd Qu.:1.0000 3rd Qu.:0.00000 3rd Qu.:2.000
## Max. :5642 Max. :3.0000 Max. :2.00000 Max. :4.000
## NA's :2 NA's :2
## HalfBath BedroomAbvGr KitchenAbvGr KitchenQual TotRmsAbvGrd

```

```

## Min. :0.0000 Min. :0.00 Min. :0.000 Ex : 205 Min. :
2.000
## 1st Qu.:0.0000 1st Qu.:2.00 1st Qu.:1.000 Fa : 70 1st Qu.:
5.000
## Median :0.0000 Median :3.00 Median :1.000 Gd :1151 Median :
6.000
## Mean :0.3803 Mean :2.86 Mean :1.045 TA :1492 Mean :
6.452
## 3rd Qu.:1.0000 3rd Qu.:3.00 3rd Qu.:1.000 NA's: 1 3rd Qu.:
7.000
## Max. :2.0000 Max. :8.00 Max. :3.000 Max.
:15.000
##
## Functional Fireplaces FireplaceQu GarageType GarageYrBlt
## Typ :2717 Min. :0.0000 Ex : 43 2Types : 23 Min. :1895
## Min2 : 70 1st Qu.:0.0000 Fa : 74 Attchd :1723 1st Qu.:1960
## Min1 : 65 Median :1.0000 Gd : 744 Basment: 36 Median :1979
## Mod : 35 Mean :0.5971 Po : 46 BuiltIn: 186 Mean :1978
## Maj1 : 19 3rd Qu.:1.0000 TA : 592 CarPort: 15 3rd Qu.:2002
## (Other): 11 Max. :4.0000 NA's:1420 Detchd : 779 Max. :2207
## NA's : 2 NA's : 157 NA's :159
## GarageFinish GarageCars GarageArea GarageQual GarageCond
## Fin : 719 Min. :0.000 Min. : 0.0 Ex : 3 Ex : 3
## RFn : 811 1st Qu.:1.000 1st Qu.: 320.0 Fa : 124 Fa : 74
## Unf :1230 Median :2.000 Median : 480.0 Gd : 24 Gd : 15
## NA's: 159 Mean :1.767 Mean : 472.9 Po : 5 Po : 14
## 3rd Qu.:2.000 3rd Qu.: 576.0 TA :2604 TA :2654
## Max. :5.000 Max. :1488.0 NA's: 159 NA's: 159
## NA's :1 NA's :1
## PavedDrive WoodDeckSF OpenPorchSF EnclosedPorch
## N: 216 Min. : 0.00 Min. : 0.00 Min. : 0.0
## P: 62 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.: 0.0
## Y:2641 Median : 0.00 Median : 26.00 Median : 0.0
## Mean : 93.71 Mean : 47.49 Mean : 23.1
## 3rd Qu.: 168.00 3rd Qu.: 70.00 3rd Qu.: 0.0
## Max. :1424.00 Max. :742.00 Max. :1012.0
##
## X3SsnPorch ScreenPorch PoolArea PoolQC Fence
## Min. : 0.000 Min. : 0.00 Min. : 0.000 Ex : 4 GdPrv:
118
## 1st Qu.: 0.000 1st Qu.: 0.00 1st Qu.: 0.000 Fa : 2 GdWo :
112
## Median : 0.000 Median : 0.00 Median : 0.000 Gd : 4 MnPrv:
329
## Mean : 2.602 Mean : 16.06 Mean : 2.252 NA's:2909 MnWw :
12
## 3rd Qu.: 0.000 3rd Qu.: 0.00 3rd Qu.: 0.000 NA's
:2348
## Max. :508.000 Max. :576.00 Max. :800.000
##

```

```

## MiscFeature      MiscVal      MoSold      YrSold
SaleType
## Gar2:    5   Min.    :    0.00   Min.    : 1.000   Min.    :2006   WD
:2525
## Othr:    4   1st Qu.:    0.00   1st Qu.: 4.000   1st Qu.:2007   New    :
239
## Shed:   95   Median :    0.00   Median : 6.000   Median :2008   COD     :
87
## TenC:    1   Mean    :   50.83   Mean    : 6.213   Mean    :2008   ConLD   :
26
## NA's:2814   3rd Qu.:    0.00   3rd Qu.: 8.000   3rd Qu.:2009   CWD     :
12
##           Max.    :17000.00   Max.    :12.000   Max.    :2010   (Other):
29
##                                           NA's    :
1
## SaleCondition    SalePrice
## Abnorml: 190   Min.    :   -1
## AdjLand:  12   1st Qu.:   -1
## Alloca :  24   Median : 34900
## Family :  46   Mean    : 90491
## Normal :2402   3rd Qu.:163000
## Partial: 245   Max.    :755000
##

```

Missing Values

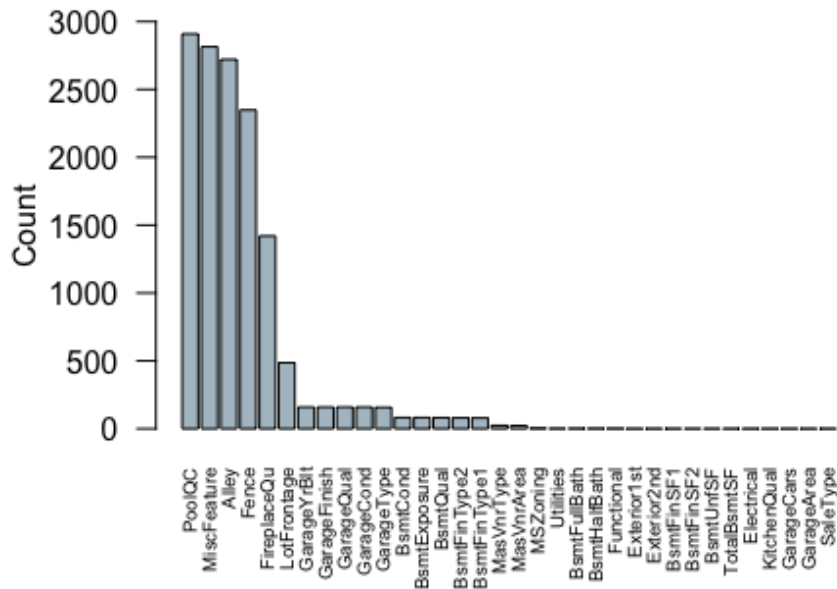
#finding how many variables with missing values are in the dataset

```

options(repr.plot.width=6, repr.plot.height=5)
cMiss = function(x){sum(is.na(x))}
CM <- sort(apply(df,2,cMiss),decreasing=T);
barplot(CM[CM!=0],
        las=2,
        cex.names=0.6,
        ylab="Count",
        ylim=c(0,3000),
        horiz=F,
        col="#AFC0CB",
        main=paste(toString(sum(CM!=0)), "variables with missing values in
dataset"))

```

34 variables with missing values in dataset



```
dfClean <-function(df)
{
  # Pool Variable: If PoolQC = NA and PoolArea = 0 , assign factor NoPool
  df$PoolQC <- as.character(df$PoolQC)
  df$PoolQC[df$PoolArea %in% c(0,NA) & is.na(df$PoolQC)] <- "NoPool"
  df$PoolQC <- as.factor(df$PoolQC)

  # MiscFeature Variable: If MiscFeature = NA and MiscVal = 0, assign factor
  None
  df$MiscFeature <- as.character(df$MiscFeature)
  df$MiscFeature[df$MiscVal %in% c(0,NA) & is.na(df$MiscFeature)] <- "None"
  df$MiscFeature <- as.factor(df$MiscFeature)

  # Alley Variable: If Alley = NA, assign factor NoAccess
  df$Alley <- as.character(df$Alley)
  df$Alley[is.na(df$Alley)] <- "NoAccess"
  df$Alley <- as.factor(df$Alley)

  # Fence Variable: If Fence = NA, assign factor NoFence
  df$Fence <- as.character(df$Fence)
  df$Fence[is.na(df$Fence)] <- "NoFence"
  df$Fence <- as.factor(df$Fence)

  # FireplaceQu Variable: If FireplaceQu = NA and Fireplaces = 0 , assign
  factor NoFirePlace
  df$FireplaceQu <- as.character(df$FireplaceQu)
  df$FireplaceQu[df$Fireplaces %in% c(0,NA) & is.na(df$FireplaceQu)] <-
```



```

"NoFirePlace"
df$FireplaceQu <- as.factor(df$FireplaceQu)

# GarageYrBlt Variable: If GarageYrBlt = NA and GarageArea = 0 assign
factor NoGarage
df$GarageYrBlt <- as.character(df$GarageYrBlt)
df$GarageYrBlt[df$GarageArea %in% c(0,NA) & is.na(df$GarageYrBlt)] <-
"NoGarage"
df$GarageYrBlt <- as.factor(df$GarageYrBlt)

# GarageFinish Variable: If GarageFinish = NA and GarageArea = 0 assign
factor NoGarage
df$GarageFinish <- as.character(df$GarageFinish)
df$GarageFinish[df$GarageArea %in% c(0,NA) & is.na(df$GarageFinish)] <-
"NoGarage"
df$GarageFinish <- as.factor(df$GarageFinish)

# GarageQual Variable: If GarageQual = NA and GarageArea = 0 assign factor
NoGarage
df$GarageQual <- as.character(df$GarageQual)
df$GarageQual[df$GarageArea %in% c(0,NA) & is.na(df$GarageQual)] <-
"NoGarage"
df$GarageQual <- as.factor(df$GarageQual)

# GarageCond Variable: If GarageCond = NA and GarageArea = 0 assign factor
NoGarage
df$GarageCond <- as.character(df$GarageCond)
df$GarageCond[df$GarageArea %in% c(0,NA) & is.na(df$GarageCond)] <-
"NoGarage"
df$GarageCond <- as.factor(df$GarageCond)

# GarageType Variable: If GarageType = NA and GarageArea = 0 assign factor
NoGarage
df$GarageType <- as.character(df$GarageType)
df$GarageType[df$GarageArea %in% c(0,NA) & is.na(df$GarageType)] <-
"NoGarage"
df$GarageType <- as.factor(df$GarageType)
df$GarageArea[is.na(df$GarageArea) & df$GarageCars %in% c(0,NA)] <- 0
df$GarageCars[is.na(df$GarageCars) & df$GarageArea %in% c(0,NA)] <- 0

# BsmtFullBath Variable: If BsmtFullBath = NA and TotalBsmtSF = 0 assign 0
df$BsmtFullBath[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtFullBath)] <- 0

# BsmtHalfBath Variable: If BsmtHalfBath = NA and TotalBsmtSF = 0 assign 0
df$BsmtHalfBath[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtHalfBath)] <- 0

# BsmtFinSF1 Variable: If BsmtFinSF1 = NA and TotalBsmtSF = 0 assign 0
df$BsmtFinSF1[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtFinSF1)] <- 0

```

```

# BsmtFinSF2 Variable: If BsmtFinSF2 = NA and TotalBsmtSF = 0 assign 0
df$BsmtFinSF2[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtFinSF2)] <- 0

# BsmtUnfSF Variable: If BsmtUnfSF = NA and TotalBsmtSF = 0 assign 0
df$BsmtUnfSF[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtUnfSF)] <- 0

# TotalBsmtSF Variable: If TotalBsmtSF = NA and TotalBsmtSF = 0 assign 0
df$TotalBsmtSF[df$TotalBsmtSF %in% c(0,NA) & is.na(df$TotalBsmtSF)] <- 0

# BsmtQual Variable: If BsmtQual = NA and TotalBsmtSF = 0 assign factor
NoBasement
df$BsmtQual <- as.character(df$BsmtQual)
df$BsmtQual[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtQual)] <-
"NoBasement"
df$BsmtQual <- as.factor(df$BsmtQual)

# BsmtFinType1 Variable: If BsmtFinType1 = NA and TotalBsmtSF = 0 assign
factor NoBasement
df$BsmtFinType1 <- as.character(df$BsmtFinType1)
df$BsmtFinType1[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtFinType1)] <-
"NoBasement"
df$BsmtFinType1 <- as.factor(df$BsmtFinType1)

# BsmtFinType2 Variable: If BsmtFinType2 = NA and TotalBsmtSF = 0 assign
factor NoBasement
df$BsmtFinType2 <- as.character(df$BsmtFinType2)
df$BsmtFinType2[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtFinType2)] <-
"NoBasement"
df$BsmtFinType2 <- as.factor(df$BsmtFinType2)

# BsmtExposure Variable: If BsmtExposure = NA and TotalBsmtSF = 0 assign
factor NoBasement
df$BsmtExposure <- as.character(df$BsmtExposure)
df$BsmtExposure[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtExposure)] <-
"NoBasement"
df$BsmtExposure <- as.factor(df$BsmtExposure)

# BsmtCond Variable: If BsmtCond = NA and TotalBsmtSF = 0 assign factor
NoBasement
df$BsmtCond <- as.character(df$BsmtCond)
df$BsmtCond[df$TotalBsmtSF %in% c(0,NA) & is.na(df$BsmtCond)] <-
"NoBasement"
df$BsmtCond <- as.factor(df$BsmtCond)
return(df)
}
df <- dfClean(df)

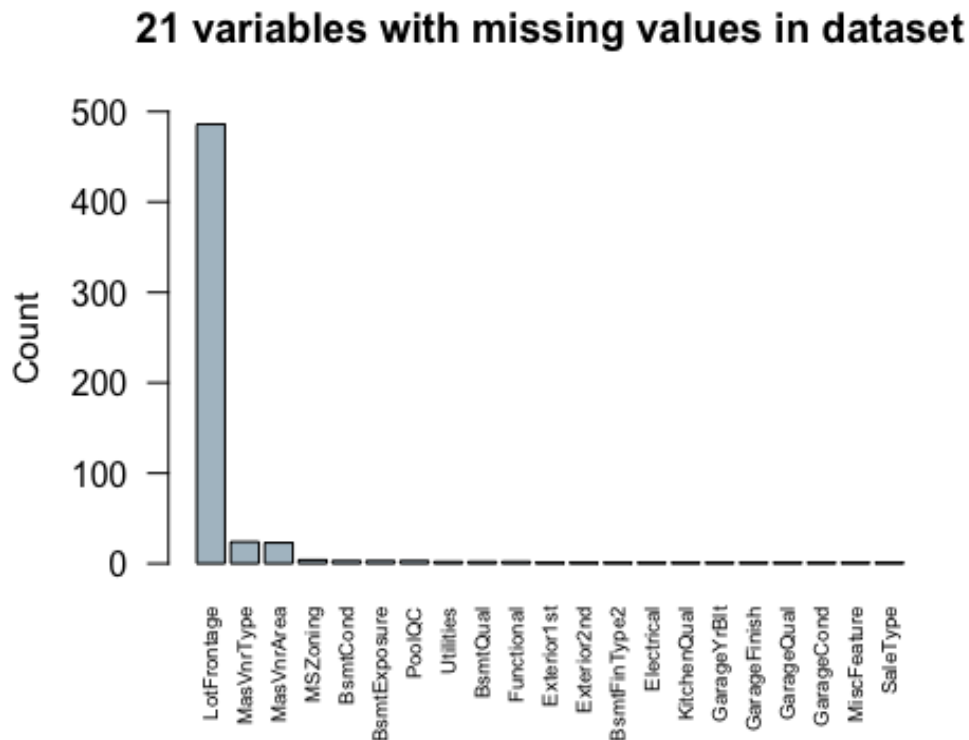
PM <- sort(apply(df,2,cMiss),decreasing=T);
barplot(PM[PM!=0],

```

```

las=2,
cex.names=0.6,
ylab="Count",
ylim=c(0,500),
horiz=F,
col="#AFC0CB",
main=paste(toString(sum(PM!=0)), "variables with missing values in
dataset"))

```



#That certainly helped a little bit. Let's see if there's a pattern to the remaining missing data.

```

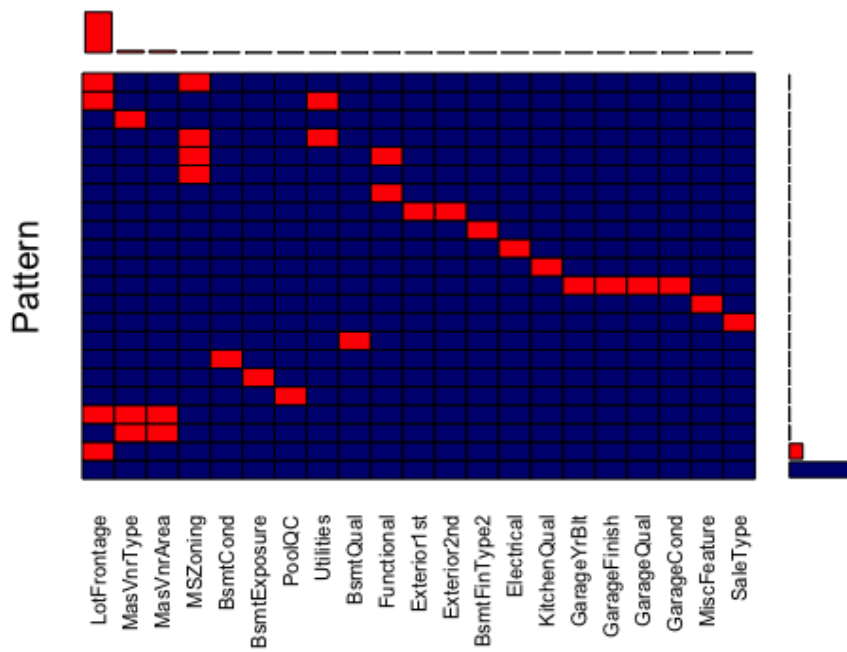
data = df[, names(PM[PM!=0])];
aggr_plot <- aggr(data,
  col=c('navyblue', 'red'),
  bars=T,
  numbers=T,
  combined = T,
  labels=names(data),
  cex.axis=.7,
  gap=3,
  ylab=c("Pattern"),
  cex.numbers=0.74)

```

```

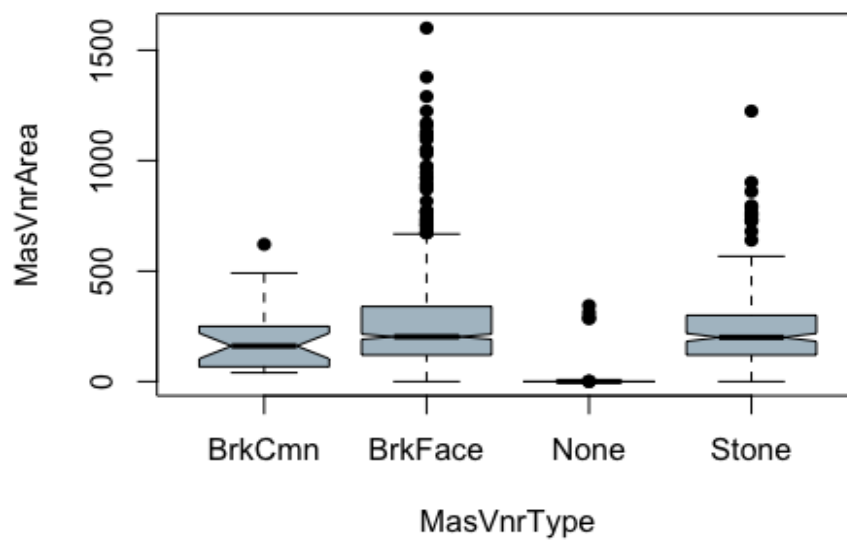
## Warning in plot.aggr(res, ...): not enough horizontal space to display
## frequencies

```



```
#MasVnrType and MasVnrArea
plot(df[,c("MasVnrType", "MasVnrArea")],
     pch=16,
     notch=TRUE,
     main="MasVnrArea vs MasVnrType boxplots",
     col="#AFC0CB")
```

MasVnrArea vs MasVnrType boxplots



```

df[ (is.na(df$MasVnrType) | is.na(df$MasVnrArea))
,c("MasVnrType", "MasVnrArea")]

##      MasVnrType MasVnrArea
## 235          <NA>        NA
## 530          <NA>        NA
## 651          <NA>        NA
## 937          <NA>        NA
## 974          <NA>        NA
## 978          <NA>        NA
## 1244         <NA>        NA
## 1279         <NA>        NA
## 1692         <NA>        NA
## 1707         <NA>        NA
## 1883         <NA>        NA
## 1993         <NA>        NA
## 2005         <NA>        NA
## 2042         <NA>        NA
## 2312         <NA>        NA
## 2326         <NA>        NA
## 2341         <NA>        NA
## 2350         <NA>        NA
## 2369         <NA>        NA
## 2593         <NA>        NA
## 2611         <NA>       198
## 2658         <NA>        NA
## 2687         <NA>        NA
## 2863         <NA>        NA

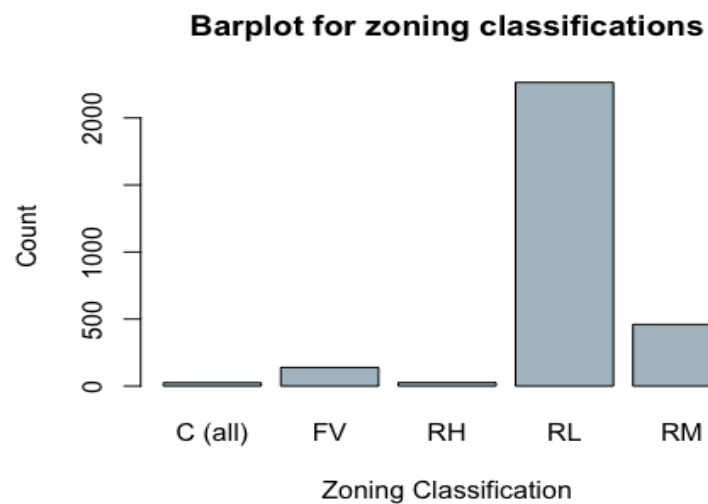
summary(df[ !(is.na(df$MasVnrType) | is.na(df$MasVnrArea))
,c("MasVnrType", "MasVnrArea")])

##      MasVnrType      MasVnrArea
## BrkCmn :   25   Min.   :   0.0
## BrkFace: 879   1st Qu.:   0.0
## None   :1742   Median :   0.0
## Stone  : 249   Mean    : 102.2
##                3rd Qu.: 164.0
##                Max.    :1600.0

df$MasVnrType <- as.character(df$MasVnrType)
df$MasVnrType[is.na(df$MasVnrType)] <- "None"
df$MasVnrType <- as.factor(df$MasVnrType)
df$MasVnrArea[is.na(df$MasVnrArea)] <- 0

#MSZoning
plot(df$MSZoning,
     col="#AFC0CB",
     xlab="Zoning Classification",
     ylab = "Count",
     main = "Barplot for zoning classifications")

```



```
df[ is.na(df$MSZoning) ,c("MSZoning","MSSubClass")]
```

```
##      MSZoning MSSubClass
## 1916      <NA>         30
## 2217      <NA>         20
## 2251      <NA>         70
## 2905      <NA>         20
```

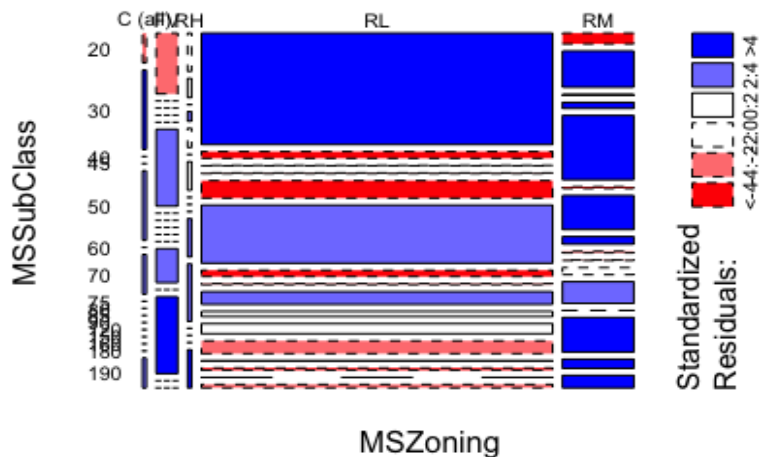
```
ZoneClassTable <- table(df[ ,c("MSZoning","MSSubClass")])
ZoneClassTable
```

```
##      MSSubClass
## MSZoning   20  30  40  45  50  60  70  75  80  85  90 120 150
160
## C (all)    3   8   0   0   7   0   4   0   0   0   0   0   0
0
## FV        34   0   0   0   0  43   0   0   0   0   0  19   0
43
## RH         4   2   0   1   2   0   3   0   0   0   4   6   0
0
## RL       1016  61   4   6  159 529  57   9  115  47  92 117   1
21
## RM         20  67   2  11  119   3  63  14   3   1  13  40   0
64
##      MSSubClass
## MSZoning   180  190
## C (all)    0    3
## FV         0    0
## RH         0    4
## RL         0   31
## RM        17   23
```

```
mosaicplot(ZoneClassTable,
            main="Mosaic Plot of MSZoning VS MSSubClass",
```

```
las=1,
color=T,
shade=T)
```

Mosaic Plot of MSZoning VS MSSubClass



```
GTest(ZoneClassTable)
```

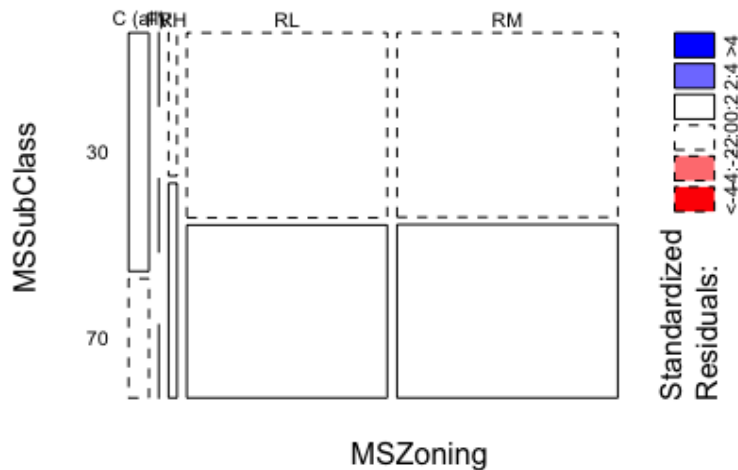
```
##
## Log likelihood ratio (G-test) test of independence without correction
##
## data: ZoneClassTable
## G = 1321.9, X-squared df = 60, p-value < 2.2e-16
```

```
Table<-table(df[ df$MSSubClass %in% c(30,70) ,c("MSZoning","MSSubClass")])
Table <- Table[ , colSums(Table != 0) > 0 ]
Table
```

```
##           MSSubClass
## MSZoning  30  70
## C (all)   8   4
## FV        0   0
## RH        2   3
## RL       61  57
## RM       67  63
```

```
mosaicplot(Table,
  main="Mosaic Plot of MSZoning VS MSSubClass (30,70)",
  las=1,
  color=T,
  shade=T)
```

Mosaic Plot of MSZoning VS MSSubClass (30,70)



```
Test1<-GTest(Table)
Test1

##
##  Log likelihood ratio (G-test) test of independence without correction
##
## data:  Table
## G = 1.3625, X-squared df = 4, p-value = 0.8507

paste("At a 95% confidence level, since the p-value =",
as.character(round(Test1$p.value,2)),
"> 0.05, we cannot reject the null hypothesis that MSZoning and
MSSubClass are independent when MSSubClass = 30 or 70.")

## [1] "At a 95% confidence level, since the p-value = 0.85 > 0.05, we cannot
reject the null hypothesis that MSZoning and MSSubClass are independent when
MSSubClass = 30 or 70."

df$MSZoning <- as.character(df$MSZoning)
df$MSZoning[is.na(df$MSZoning)] <- "RL"
df$MSZoning <- as.factor(df$MSZoning)

#Basement
MissBsmt = c('BsmtCond', 'BsmtExposure', 'BsmtQual', 'BsmtFinType2')
df[!complete.cases(df[,names(df) %in% MissBsmt]),names(df) %in%
names(df)[which(grepl("Bsmt",names(df)))]]

##      BsmtQual BsmtCond BsmtExposure BsmtFinType1 BsmtFinSF1 BsmtFinType2
## 333      Gd      TA      No      GLQ      1124      <NA>
## 949      Gd      TA      <NA>      Unf          0      Unf
## 1488     Gd      TA      <NA>      Unf          0      Unf
## 2041     Gd     <NA>      Mn      GLQ     1044      Rec
```


## 2186	TA	<NA>	No	BLQ	1033	Unf
## 2218	<NA>	Fa	No	Unf	0	Unf
## 2219	<NA>	TA	No	Unf	0	Unf
## 2349	Gd	TA	<NA>	Unf	0	Unf
## 2525	TA	<NA>	Av	ALQ	755	Unf
##	BsmtFinSF2	BsmtUnfSF	TotalBsmtSF	BsmtFullBath	BsmtHalfBath	
## 333	479	1603	3206	1	0	
## 949	0	936	936	0	0	
## 1488	0	1595	1595	0	0	
## 2041	382	0	1426	1	0	
## 2186	0	94	1127	0	1	
## 2218	0	173	173	0	0	
## 2219	0	356	356	0	0	
## 2349	0	725	725	0	0	
## 2525	0	240	995	0	0	

#BsmtExposure

```
df$BsmtExposure <- as.character(df$BsmtExposure)
df$BsmtExposure[is.na(df$BsmtExposure)]<-"No"
df$BsmtExposure <- as.factor(df$BsmtExposure)
```

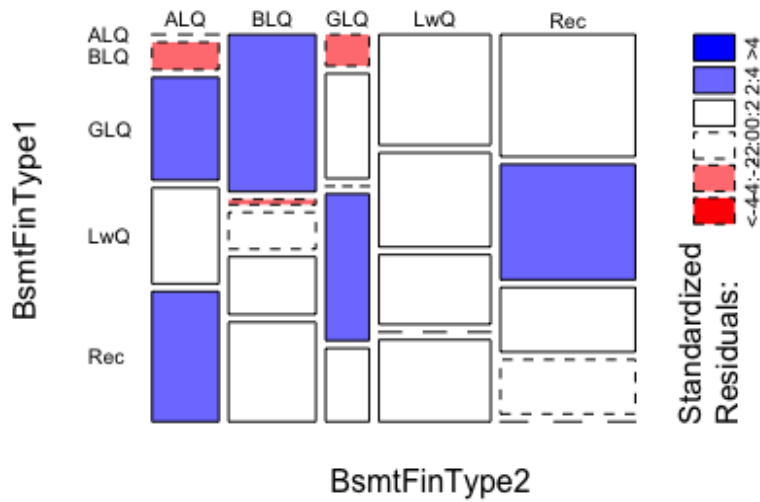
#BsmtFinType2

```
BsmtFinQuality<-table(df[!(df$BsmtFinType2 %in% c("NoBasement","Unf")) |
df$BsmtFinType1 %in% c("NoBasement","Unf"))
,c("BsmtFinType2","BsmtFinType1"]])
BsmtFinQuality<-BsmtFinQuality[rowSums(BsmtFinQuality != 0) > 0 ,
colSums(BsmtFinQuality != 0) > 0]
BsmtFinQuality
```

##		BsmtFinType1				
##	BsmtFinType2	ALQ	BLQ	GLQ	LwQ	Rec
##	ALQ	0	4	15	14	19
##	BLQ	30	1	7	11	19
##	GLQ	3	10	0	14	7
##	LwQ	27	23	17	0	20
##	Rec	36	34	19	16	0

```
mosaicplot(BsmtFinQuality,
main="Mosaic Plot of BsmtFinType",
las=1,
color=T,
shade=T)
```

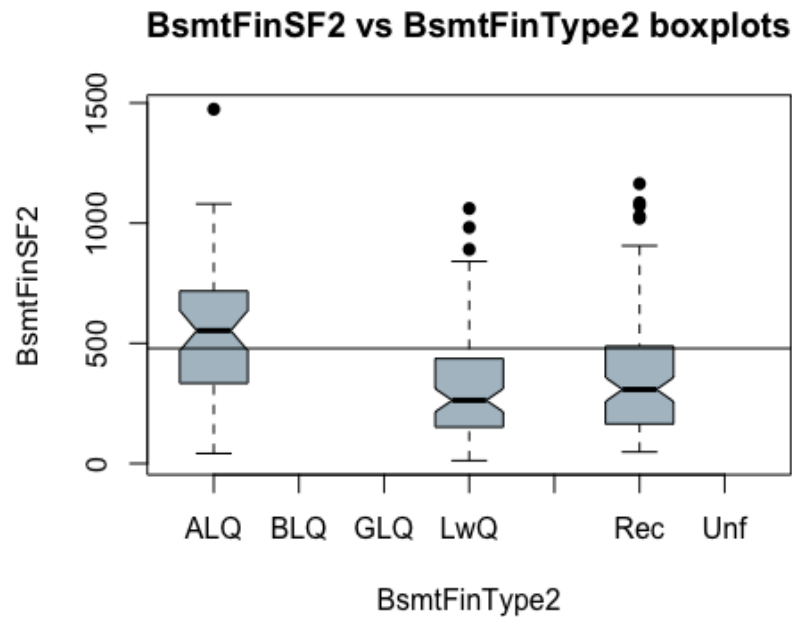
Mosaic Plot of BsmtFinType



```
TestQ<-GTest(BsmtFinQuality)
TestQ

##
## Log likelihood ratio (G-test) test of independence without correction
##
## data: BsmtFinQuality
## G = 184.71, X-squared df = 16, p-value < 2.2e-16

plot(df[df$BsmtFinType2 %in% c("ALQ","LwQ",
"Rec"),c("BsmtFinType2","BsmtFinSF2")],
     pch=16,
     notch=TRUE,
     main="BsmtFinSF2 vs BsmtFinType2 boxplots",
     col="#AFC0CB")
abline(h=df[is.na(df$BsmtFinType2) ,c("BsmtFinSF2")])
```



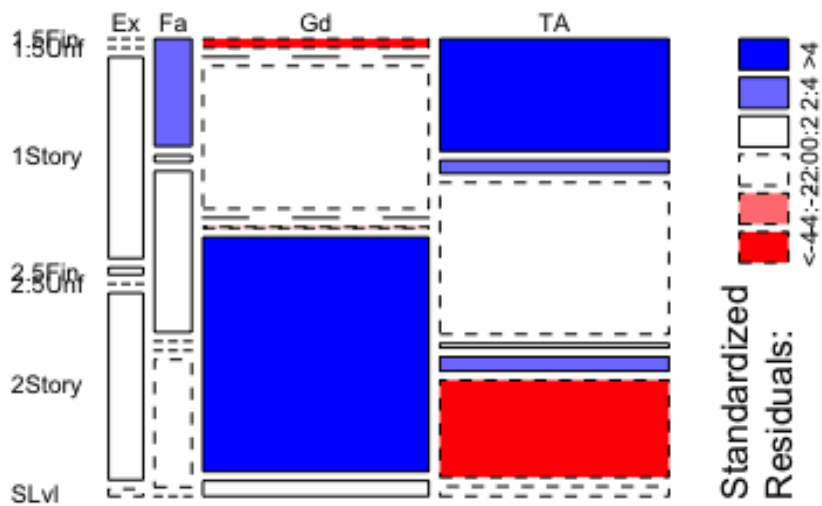
```
df$BsmtFinType2 <- as.character(df$BsmtFinType2)
df$BsmtFinType2[is.na(df$BsmtFinType2)]<-"ALQ"
df$BsmtFinType2 <- as.factor(df$BsmtFinType2)

#BsmtQual
BsmtQualUnf<-table(df$BsmtQual[df$BsmtUnfSF==df$TotalBsmtSF &
df$TotalBsmtSF>0],df$HouseStyle[df$BsmtUnfSF==df$TotalBsmtSF &
df$TotalBsmtSF>0])
BsmtQualUnf<-BsmtQualUnf[ rowSums(BsmtQualUnf != 0) > 0 , colSums(BsmtQualUnf
!= 0) > 0 ]
BsmtQualUnf

##
##      1.5Fin 1.5Unf 1Story 2.5Fin 2.5Unf 2Story SLvl
## Ex       0      0     28      1      0     26     1
## Fa      16      1     24      0      0     19     0
## Gd       8      0    129      0      1    212    14
## TA     103     12    139      4     13     89     9

mosaicplot(BsmtQualUnf,
  main="Mosaic Plot of Basement Quality",
  las=1,
  color=T,
  shade=T)
```

Mosaic Plot of Basement Quality



```
TestQ2<-GTest(BsmtQualUnf)
TestQ2

##
## Log likelihood ratio (G-test) test of independence without correction
##
## data: BsmtQualUnf
## G = 220.7, X-squared df = 18, p-value < 2.2e-16

df$HouseStyle[is.na(df$BsmtQual)]

## [1] 2Story 1.5Fin
## Levels: 1.5Fin 1.5Unf 1Story 2.5Fin 2.5Unf 2Story SFOyer SLvl

df$BsmtQual <- as.character(df$BsmtQual)
df$BsmtQual[is.na(df$BsmtQual) & df$HouseStyle == "2Story"]<-"Gd"
df$BsmtQual[is.na(df$BsmtQual) & df$HouseStyle == "1.5Fin"]<-"TA"
df$BsmtQual <- as.factor(df$BsmtQual)

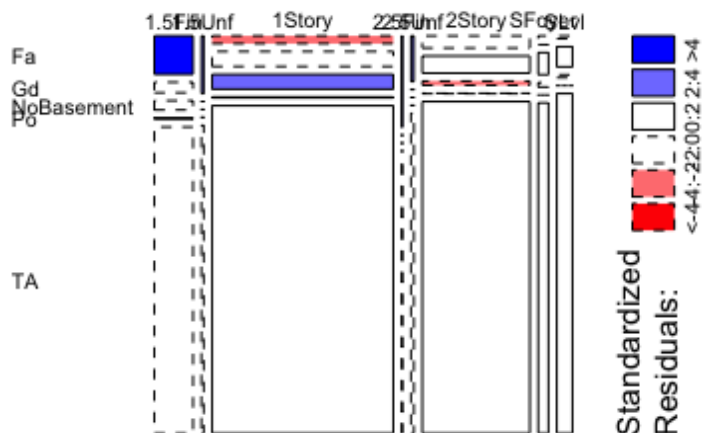
#BsmtCond
TableBsmtCond<-table(df$HouseStyle,df$BsmtCond)
TableBsmtCond<-TableBsmtCond[rowSums(TableBsmtCond != 0) > 0 ,
colSums(TableBsmtCond != 0) > 0]
TableBsmtCond

##
##          Fa   Gd NoBasement   Po   TA
## 1.5Fin    33    9           8    1  263
## 1.5Unf     3    0           0    0   16
## 1Story    31   60          59    3 1316
## 2.5Fin     2    0           0    0    6
## 2.5Unf     3    0           0    0   21
```

```
##    2Story    29    41          10    1   791
##    SFoyer     2     5           1    0    75
##    SLvl       1     7           1    0   118
```

```
mosaicplot(TableBsmtCond,
            main="Mosaic Plot of Basement Quality",
            las=1,
            color=T,
            shade=T)
```

Mosaic Plot of Basement Quality



```
TestQ2<-GTest(TableBsmtCond)
TestQ2

##
##  Log likelihood ratio (G-test) test of independence without correction
##
## data:  TableBsmtCond
## G = 89.202, X-squared df = 28, p-value = 2.64e-08

df$HouseStyle[is.na(df$BsmtCond)]

## [1] 1Story 1Story SLvl
## Levels: 1.5Fin 1.5Unf 1Story 2.5Fin 2.5Unf 2Story SFoyer SLvl

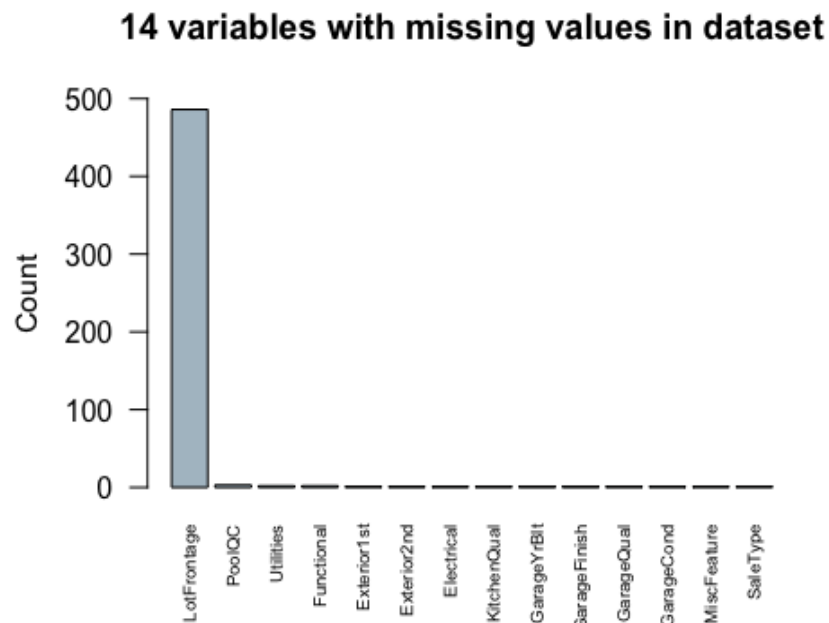
df$BsmtCond <- as.character(df$BsmtCond)
df$BsmtCond[is.na(df$BsmtCond)]<- "TA"
df$BsmtCond <- as.factor(df$BsmtCond)

PM <- sort(apply(df,2,cMiss),decreasing=T);
barplot(PM[PM!=0],
        las=2,
        cex.names=0.6,
        ylab="Count",
        ylim=c(0,500),
```

```

horiz=F,
col="#AFC0CB",
main=paste(toString(sum(PM!=0)), "variables with missing values in
dataset"))

```

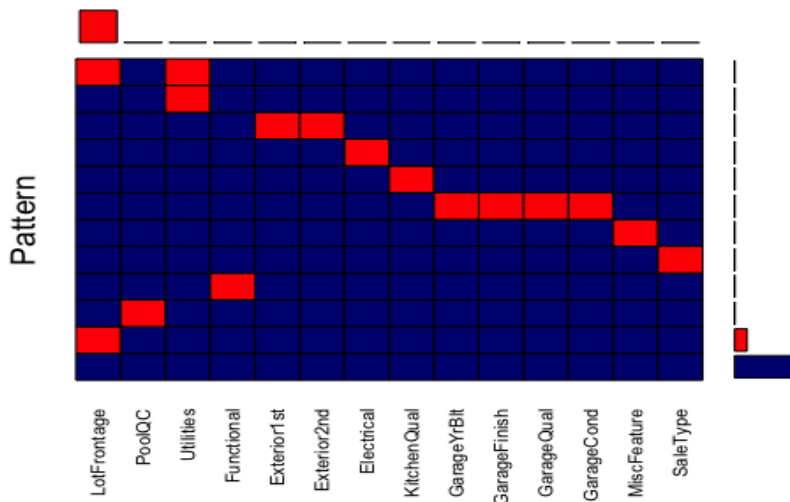


```

data = df[, names(PM[PM!=0])];
aggr_plot <- aggr(data,
  col=c('navyblue','red'),
  bars=T,
  numbers=T,
  combined = T,
  labels=names(data),
  cex.axis=.7,
  gap=3,
  ylab=c("Pattern"),
  cex.numbers=0.74)

## Warning in plot.aggr(res, ...): not enough horizontal space to display
## frequencies

```

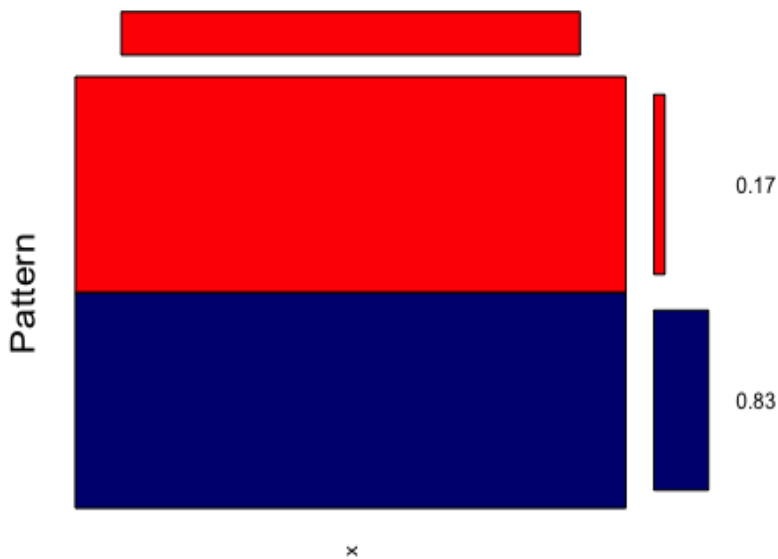


```
#The rest
fillMiss<- function(x)
{
  ux <- unique(x[!is.na(x)])
  x <- as.character(x)
  mode <- ux[which.max(tabulate(match(x[!is.na(x)], ux)))]
  x[is.na(x)] <- as.character(mode)
  x <- as.factor(x)
  return(x)
}
df[,sapply(df,function(x){!(is.numeric(x))}) ]<-
as.data.frame(apply(df[,sapply(df,function(x){!(is.numeric(x))})
],2,fillMiss))
PM <- sort(apply(df,2,cMiss),decreasing=T);
barplot(PM[PM!=0],
        las=2,
        cex.names=0.6,
        ylab="Count",
        ylim=c(0,500),
        horiz=F,
        col="#AFC0CB",
        main=paste(toString(sum(PM!=0)), "variables with missing values in
dataset"))
```

1 variables with missing values in dataset



```
data = df[, names(PM[PM!=0])];  
aggr_plot <- aggr(data,  
  col=c('navyblue','red'),  
  bars=T,  
  numbers=T,  
  combined = T,  
  labels=names(data),  
  cex.axis=.7,  
  gap=3,  
  ylab=c("Pattern"),  
  cex.numbers=0.74)
```



#LotFrontage Imputation

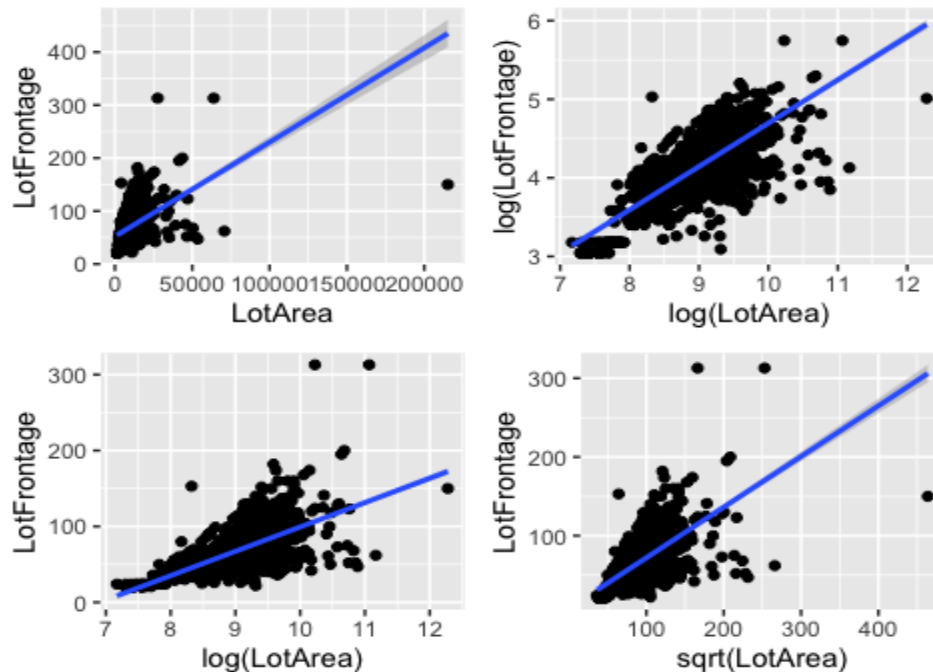
#Let's investigate this variable further. Maybe we could use a regression imputation technique to impute the missing LotFrontage variables.

#Since LotFrontage is defined as "Linear feet of street connected to property", we would suspect that this variable would be related to quantities like "LotArea", "Street", "LotShape", "LandContour", "LotConfig", "LandSlope", "Neighborhood", "BldgType".

```
multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL)
{
  library(grid)
  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)
  numPlots = length(plots)
  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout))
  {
    # Make the panel
    # ncol: Number of columns of plots
    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                      ncol = cols, nrow = ceiling(numPlots/cols))
  }
  if (numPlots==1)
  {
    print(plots[[1]])
  }
  else
  {
    # Set up the page
    grid.newpage()
    pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))
    # Make each plot, in the correct location
    for (i in 1:numPlots)
    {
      # Get the i,j matrix positions of the regions that contain this subplot
      matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))
      print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
                                       layout.pos.col = matchidx$col))
    }
  }
}

p1<-ggplot(df, aes(LotArea, LotFrontage)) + geom_point() + geom_smooth(method
= "lm", se = T)
p2<-ggplot(df, aes(log(LotArea), LotFrontage)) + geom_point() +
geom_smooth(method = "lm", se = T)
p3<-ggplot(df, aes(log(LotArea), log(LotFrontage))) + geom_point() +
geom_smooth(method = "lm", se = T)
p4<-ggplot(df, aes(sqrt(LotArea), LotFrontage)) + geom_point() +
geom_smooth(method = "lm", se = T)
multiplot(p1, p2, p3, p4, cols=2)

## `geom_smooth()` using formula 'y ~ x'
```



#To check outliers

```
chisq.out.test(df$LotArea,opposite=F)
```

```
##
```

```
## chi-squared test for outlier
```

```
##
```

```
## data: df$LotArea
```

```
## X-squared = 676.1, p-value < 2.2e-16
```

```
## alternative hypothesis: highest value 215245 is an outlier
```

```
chisq.out.test(df$LotFrontage,opposite=F)
```

```
##
```

```
## chi-squared test for outlier
```

```
##
```

```
## data: df$LotFrontage
```

```
## X-squared = 108.97, p-value < 2.2e-16
```

```
## alternative hypothesis: highest value 313 is an outlier
```

```
chisq.out.test(df$LotArea,opposite=T)
```

```
##
```

```
## chi-squared test for outlier
```

```
##
```

```
## data: df$LotArea
```

```
## X-squared = 1.2643, p-value = 0.2608
```

```
## alternative hypothesis: lowest value 1300 is an outlier
```

```
chisq.out.test(df$LotFrontage,opposite=T)
```

```
##
## chi-squared test for outlier
##
## data: df$LotFrontage
## X-squared = 4.2817, p-value = 0.03853
## alternative hypothesis: lowest value 21 is an outlier

grubbs.test(df$LotArea,type=11)

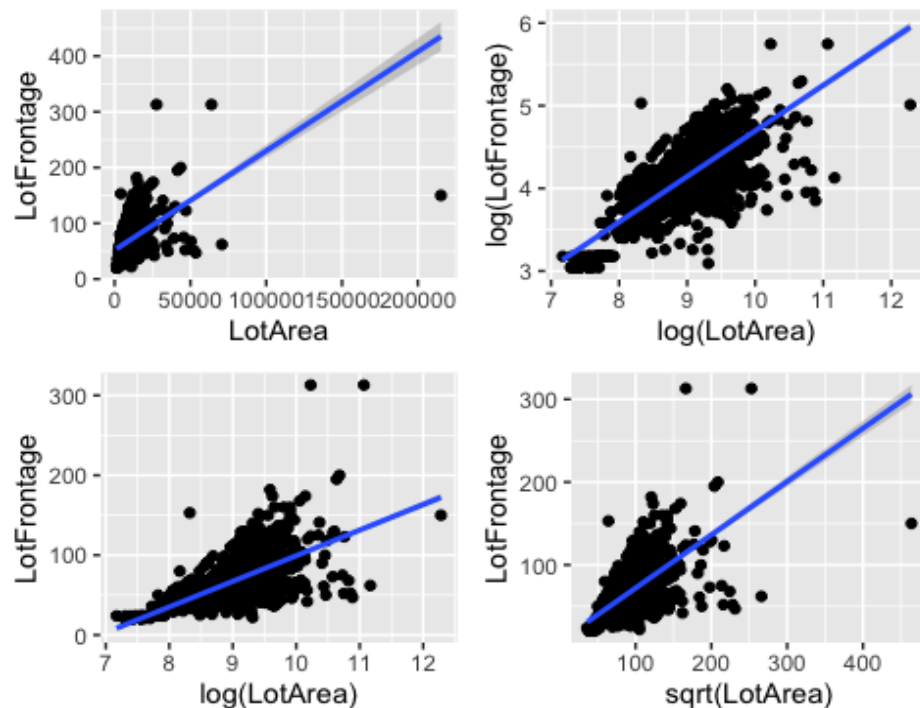
##
## Grubbs test for two opposite outliers
##
## data: df$LotArea
## G = 27.12630, U = 0.76779, p-value < 2.2e-16
## alternative hypothesis: 1300 and 215245 are outliers

grubbs.test(df$LotFrontage,type=11)

##
## Grubbs test for two opposite outliers
##
## data: df$LotFrontage
## G = 12.50808, U = 0.95342, p-value < 2.2e-16
## alternative hypothesis: 21 and 313 are outliers

p1<-ggplot(df , aes(LotArea, LotFrontage)) + geom_point() +
geom_smooth(method = "lm", se = T)
p2<-ggplot(df, aes(log(LotArea), LotFrontage)) + geom_point() +
geom_smooth(method = "lm", se = T)
p3<-ggplot(df, aes(log(LotArea), log(LotFrontage))) + geom_point() +
geom_smooth(method = "lm", se = T)
p4<-ggplot(df, aes(sqrt(LotArea), LotFrontage)) + geom_point() +
geom_smooth(method = "lm", se = T)
multiplot(p1, p2, p3, p4, cols=2)

## `geom_smooth()` using formula 'y ~ x'
```



```
cor(as.numeric(df$LotArea),as.numeric(df$LotFrontage),use="complete.obs")
## [1] 0.4898956

cor(log(as.numeric(df$LotArea)),log(as.numeric(df$LotFrontage)),use="complete
.obs")
## [1] 0.7662858

cor(log(as.numeric(df$LotArea)),as.numeric(df$LotFrontage),use="complete.obs"
)
## [1] 0.6835123

cor(sqrt(as.numeric(df$LotArea)),as.numeric(df$LotFrontage),use="complete.obs
")
## [1] 0.647658

str(df)

## '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 ...
## $ 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 NA 51 50 ...
## $ 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 "Grv1","NoAccess",...: 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 ...
## $ 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 ...
## $ 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 ...
## $ 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 ...
## $ BldgType    : Factor w/ 5 levels "1Fam","2fmCon",...: 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 ...
## $ 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 ...
## $ RoofMatl    : Factor w/ 8 levels "ClyTile","CompShg",...: 2 2 2 2 2 2 2 2
2 2 2 ...
## $ Exterior1st : Factor w/ 15 levels "AsbShng","AsphShn",...: 13 9 13 14
13 13 13 7 4 9 ...
## $ 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 ...
## $ ExterCond   : Factor w/ 5 levels "Ex","Fa","Gd",...: 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 ...
## $ BsmtQual    : Factor w/ 5 levels "Ex","Fa","Gd",...: 3 3 3 5 3 3 1 3
5 ...
## $ BsmtCond    : Factor w/ 5 levels "Fa","Gd","NoBasement",...: 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 : num 706 978 486 216 655 ...
## $ BsmtFinType2 : Factor w/ 7 levels "ALQ","BLQ","GLQ",...: 7 7 7 7 7 7 7 2
7 7 ...
## $ BsmtFinSF2 : num 0 0 0 0 0 0 0 32 0 0 ...
## $ BsmtUnfSF : num 150 284 434 540 490 64 317 216 952 140 ...
## $ TotalBsmtSF : num 856 1262 920 756 1145 ...
## $ 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/ 5 levels "FuseA","FuseF",...: 5 5 5 5 5 5 5 5 5 2
5 ...
## $ 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
...
## $ BsmtFullBath : num 1 0 1 1 1 1 1 1 0 1 ...
## $ BsmtHalfBath : num 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 ...
## $ GarageType : Factor w/ 7 levels "2Types","Attchd",...: 2 2 2 6 2 2 2 2
6 2 ...
## $ GarageYrBlt : Factor w/ 104 levels "1895","1896",...: 95 68 93 90 92 85
96 65 24 32 ...
## $ GarageFinish : Factor w/ 4 levels "Fin","NoGarage",...: 3 3 3 4 3 4 3 3
4 3 ...
## $ GarageCars : num 2 2 2 3 3 2 2 2 2 1 ...
## $ GarageArea : num 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 ...
## $ 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 ...
## $ 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 : num 208500 181500 223500 140000 250000 ...
```

We have cleaned all of the data

```
#splitting back to Test and Train
Traindata<-df[1:1460,]
Testdata<-df[(1461):nrow(df),]
#Testdata<- testdata[ , -which(names(Testdata) %in% c("SalePrice"))]
```

```
str(Testdata)
```

```
## 'data.frame': 1459 obs. of 81 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 ...
## $ MSZoning : Factor w/ 5 levels "C (all)","FV",...: 3 4 4 4 4 4 4 4 4 4 ...
## $ LotFrontage : int 80 81 74 78 43 75 NA 63 85 70 ...
## $ LotArea : int 11622 14267 13830 9978 5005 10000 7980 8402 10176 8400 ...
## $ Street : Factor w/ 2 levels "Grvl","Pave": 2 2 2 2 2 2 2 2 2 2 ...
## $ Alley : Factor w/ 3 levels "Grvl","NoAccess",...: 2 2 2 2 2 2 2 2 2 2 ...
## $ LotShape : Factor w/ 4 levels "IR1","IR2","IR3",...: 4 1 1 1 1 1 1 1 1 1 ...
## $ LandContour : Factor w/ 4 levels "Bnk","HLS","Low",...: 4 4 4 4 2 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 1 5 5 5 1 5
5 5 1 ...
## $ LandSlope      : Factor w/ 3 levels "Gtl","Mod","Sev": 1 1 1 1 1 1 1 1 1
1 ...
## $ Neighborhood  : Factor w/ 25 levels "Blmngtn","Blueste",...: 13 13 9 9 22
9 9 9 9 13 ...
## $ 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 ...
## $ BldgType       : Factor w/ 5 levels "1Fam","2fmCon",...: 1 1 1 1 5 1 1 1 1
1 ...
## $ HouseStyle     : Factor w/ 8 levels "1.5Fin","1.5Unf",...: 3 3 6 6 3 6 3 6
3 3 ...
## $ OverallQual    : int   5 6 5 6 8 6 6 6 7 4 ...
## $ OverallCond    : int   6 6 5 6 5 5 7 5 5 5 ...
## $ 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      : Factor w/ 6 levels "Flat","Gable",...: 2 4 2 2 2 2 2 2 2
2 ...
## $ RoofMatl       : Factor w/ 8 levels "ClyTile","CompShg",...: 2 2 2 2 2 2 2 2
2 2 2 ...
## $ Exterior1st    : Factor w/ 15 levels "AsbShng","AsphShn",...: 13 14 13 13
7 7 7 13 7 10 ...
## $ Exterior2nd    : Factor w/ 16 levels "AsbShng","AsphShn",...: 14 15 14 14
7 7 7 14 7 11 ...
## $ MasVnrType     : Factor w/ 4 levels "BrkCmn","BrkFace",...: 3 2 3 2 3 3 3 3
3 3 3 ...
## $ MasVnrArea     : num   0 108 0 20 0 0 0 0 0 0 ...
## $ ExterQual      : Factor w/ 4 levels "Ex","Fa","Gd",...: 4 4 4 4 3 4 4 4 4
4 ...
## $ ExterCond      : Factor w/ 5 levels "Ex","Fa","Gd",...: 5 5 5 5 5 5 3 5 5
5 ...
## $ Foundation     : Factor w/ 6 levels "BrkTil","CBlock",...: 2 2 3 3 3 3 3 3
3 2 ...
## $ BsmtQual       : Factor w/ 5 levels "Ex","Fa","Gd",...: 5 5 3 5 3 3 3 3 3
5 ...
## $ BsmtCond       : Factor w/ 5 levels "Fa","Gd","NoBasement",...: 5 5 5 5 5
5 5 5 5 5 ...
## $ BsmtExposure   : Factor w/ 5 levels "Av","Gd","Mn",...: 4 4 4 4 4 4 4 4 2
4 ...
## $ BsmtFinType1   : Factor w/ 7 levels "ALQ","BLQ","GLQ",...: 6 1 3 3 1 7 1 7
3 1 ...
## $ BsmtFinSF1     : num   468 923 791 602 263 0 935 0 637 804 ...
## $ BsmtFinType2   : Factor w/ 7 levels "ALQ","BLQ","GLQ",...: 4 7 7 7 7 7 7 7
7 6 ...

```



```

## $ BsmtFinSF2 : num 144 0 0 0 0 0 0 0 0 78 ...
## $ BsmtUnfSF : num 270 406 137 324 1017 ...
## $ TotalBsmtSF : num 882 1329 928 926 1280 ...
## $ 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",...: 5 5 3 1 1 3 1 3 3
5 ...
## $ CentralAir : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 ...
## $ Electrical : Factor w/ 5 levels "FuseA","FuseF",...: 5 5 5 5 5 5 5 5 5
5 ...
## $ 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 ...
## $ BsmtFullBath : num 0 0 0 0 0 0 1 0 1 1 ...
## $ BsmtHalfBath : num 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 : Factor w/ 4 levels "Ex","Fa","Gd",...: 4 3 4 3 3 4 4 4 3
4 ...
## $ TotRmsAbvGrd : int 5 6 6 7 5 7 6 7 5 4 ...
## $ Functional : Factor w/ 7 levels "Maj1","Maj2",...: 7 7 7 7 7 7 7 7 7 7
...
## $ Fireplaces : int 0 0 1 1 0 1 0 1 1 0 ...
## $ FireplaceQu : Factor w/ 6 levels "Ex","Fa","Gd",...: 4 4 6 3 4 6 4 3 5
4 ...
## $ GarageType : Factor w/ 7 levels "2Types","Attchd",...: 2 2 2 2 2 2 2 2 2
2 2 ...
## $ GarageYrBlt : Factor w/ 104 levels "1895","1896",...: 53 50 89 90 84 85
84 90 82 62 ...
## $ GarageFinish : Factor w/ 4 levels "Fin","NoGarage",...: 4 4 1 1 3 1 1 1
4 1 ...
## $ GarageCars : num 1 1 2 2 2 2 2 2 2 2 ...
## $ GarageArea : num 730 312 482 470 506 440 420 393 506 525 ...
## $ GarageQual : Factor w/ 6 levels "Ex","Fa","Gd",...: 6 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
6 ...
## $ PavedDrive : Factor w/ 3 levels "N","P","Y": 3 3 3 3 3 3 3 3 3 3 ...
## $ 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 ...
## $ 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 : Factor w/ 4 levels "Ex","Fa","Gd",...: 4 4 4 4 4 4 4 4 4 4
4 ...
## $ Fence : Factor w/ 5 levels "GdPrv","GdWo",...: 3 5 3 5 5 5 1 5 5

```

```

3 ...
## $ MiscFeature : Factor w/ 5 levels "Gar2","None",...: 2 1 2 2 2 2 4 2 2 2
...
## $ MiscVal : int 0 12500 0 0 0 0 500 0 0 0 ...
## $ 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 : Factor w/ 9 levels "COD","Con","ConLD",...: 9 9 9 9 9 9 9 9
9 9 9 ...
## $ SaleCondition: Factor w/ 6 levels "Abnorml","AdjLand",...: 5 5 5 5 5 5 5
5 5 5 ...
## $ SalePrice : num -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...

```

str(Traindata)

```

## '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 ...
## $ 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 NA 51 50 ...
## $ 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","NoAccess",...: 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 ...
## $ 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 ...
## $ 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 ...
## $ 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 ...
## $ 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/ 8 levels "ClyTile","CompShg",...: 2 2 2 2 2 2 2
2 2 2 ...
## $ Exterior1st : Factor w/ 15 levels "AsbShng","AsphShn",...: 13 9 13 14
13 13 13 7 4 9 ...
## $ 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 ...
## $ 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","NoBasement",...: 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 : num 706 978 486 216 655 ...
## $ BsmtFinType2 : Factor w/ 7 levels "ALQ","BLQ","GLQ",...: 7 7 7 7 7 7 7 2
7 7 ...
## $ BsmtFinSF2 : num 0 0 0 0 0 0 0 32 0 0 ...
## $ BsmtUnfSF : num 150 284 434 540 490 64 317 216 952 140 ...
## $ TotalBsmtSF : num 856 1262 920 756 1145 ...
## $ 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/ 5 levels "FuseA","FuseF",...: 5 5 5 5 5 5 5 5 5
5 ...
## $ 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
...
## $ BsmtFullBath : num 1 0 1 1 1 1 1 1 0 1 ...
## $ BsmtHalfBath : num 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 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 ...
## $ GarageType    : Factor w/ 7 levels "2Types","Attchd",...: 2 2 2 6 2 2 2 2
6 2 ...
## $ GarageYrBlt   : Factor w/ 104 levels "1895","1896",...: 95 68 93 90 92 85
96 65 24 32 ...
## $ GarageFinish  : Factor w/ 4 levels "Fin","NoGarage",...: 3 3 3 4 3 4 3 3
4 3 ...
## $ GarageCars    : num 2 2 2 3 3 2 2 2 2 1 ...
## $ GarageArea    : num 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 ...
## $ 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 ...
## $ 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     : num 208500 181500 223500 140000 250000 ...

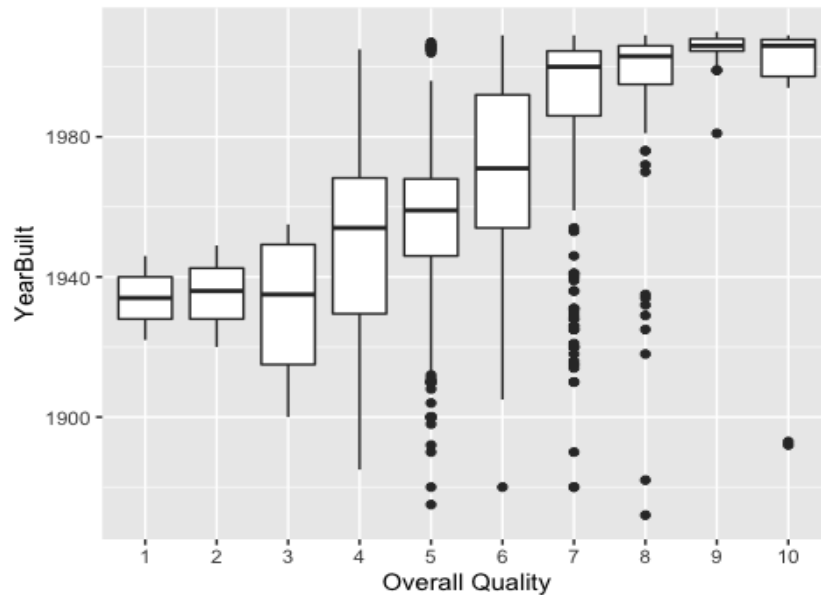
```

Exploratory Data Analysis

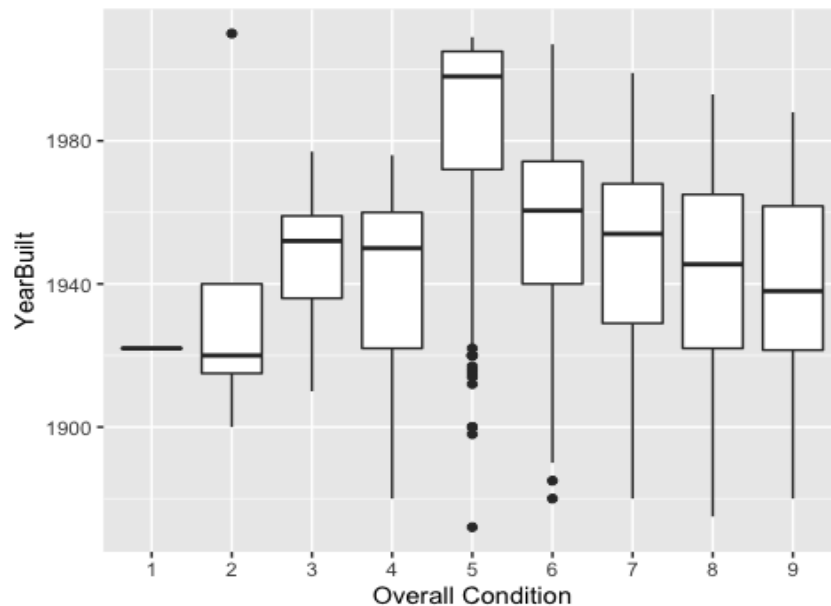
Here are the steps that we will take to understand the data and variables to get a better understanding of our data.

#Analyze the dependent variable sale price.

```
ggplot(Traindata, aes(factor(OverallQual),YearBuilt)) + geom_boxplot()  
+xlab("Overall Quality")
```



```
ggplot(Traindata, aes(factor(OverallCond),YearBuilt)) + geom_boxplot()  
+xlab("Overall Condition")
```



#The recently built homes have better Overall Quality, but the Overall condition of these recently built homes is worse than the old homes. Newer built homes are of mediocre quality.

#Let's plot the correlation matrix of numeric variables in the dataset

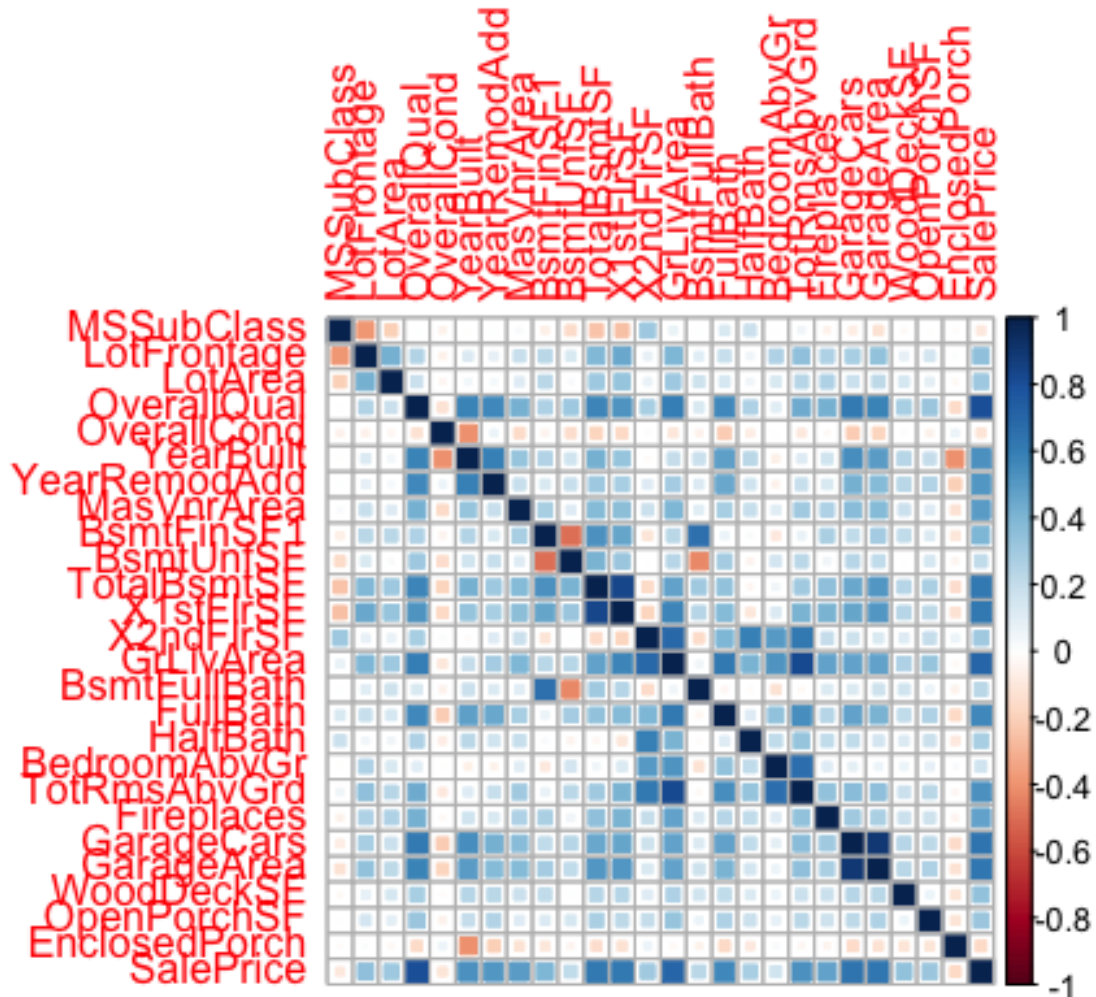
```
train_num <- Traindata[sapply(Traindata,is.numeric)]
```

```
correlations <- cor(na.omit(train_num))
```

```
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")
```



*#Let's make some scatter plot for some of the high correlation variables.
High correlation variables:*

#OverallQual: Rates the overall material and finish of the house 1-10.

#YearBuilt: Year house was built

#MasVnrArea: Masonary veener area in square feet

#TotalBsmtSF: Total Square feet of basement Area

#X1stFlrSF: First floor Square feet

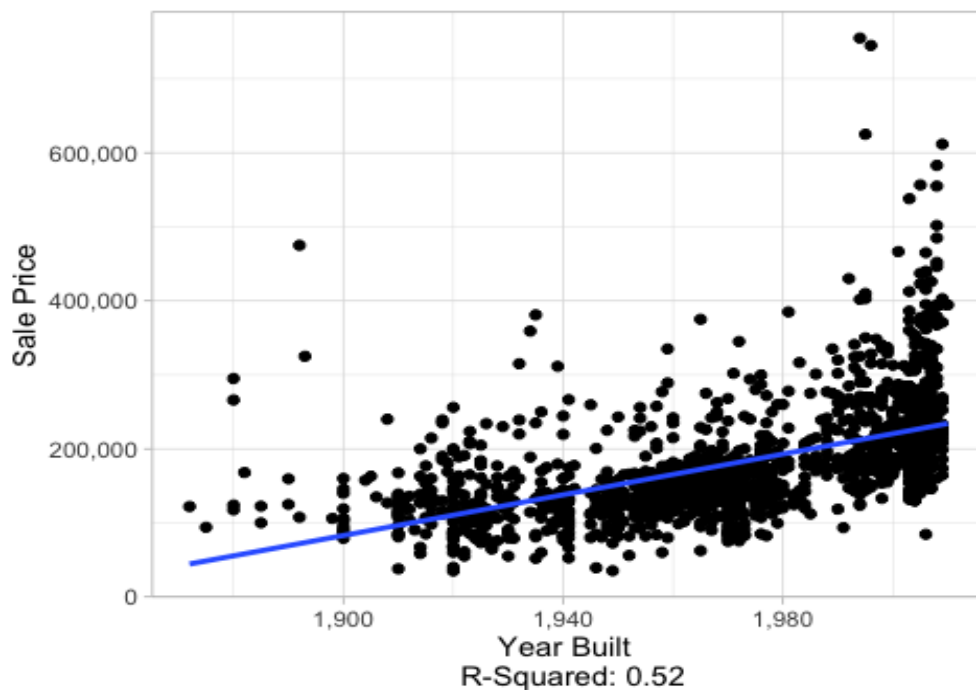
#GrLivArea: Ground Living Area

```

#FullBath: Full Bathrooms above grade
#TotRmsAbvGrd: Total rooms above grade (doesn't include bathrooms)
#GarageCars: Size of garage in car capacity
#GarageArea: Size of garage in square feet

makeScatterplots <- function(dataframe,x.variable, y.variable, xlabel,
ylabel)
{
  p = ggplot(dataframe, aes_string(x=x.variable,y= y.variable)) +
    geom_point() +
    geom_smooth(method=lm, se=FALSE) +
    ylab(ylabel) +
    xlab(paste(xlabel,'\n', 'R-Squared:', round(cor(x.variable, y.variable),
2))) +
    theme_light() +
    scale_x_continuous(labels = comma) +
    scale_y_continuous(labels = comma)
  return(p)
}
makeScatterplots(train_num, train_num$YearBuilt, train_num$SalePrice, "Year
Built", "Sale Price")
## `geom_smooth()` using formula 'y ~ x'

```

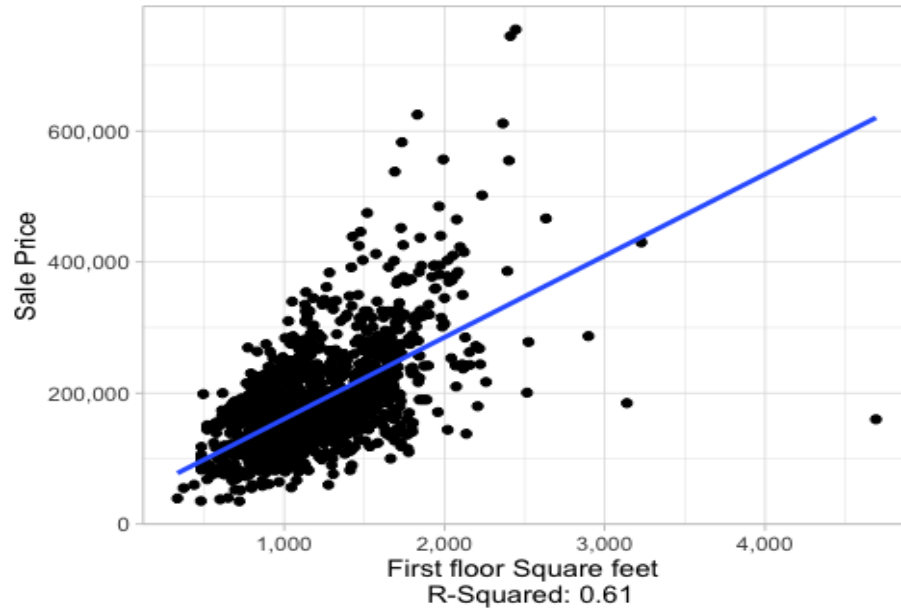


```

#There are a few recently built homes that are outliers and have much higher
sale price.
makeScatterplots(train_num, train_num$X1stFlrSF, train_num$SalePrice, "First
floor Square feet", "Sale Price")

```

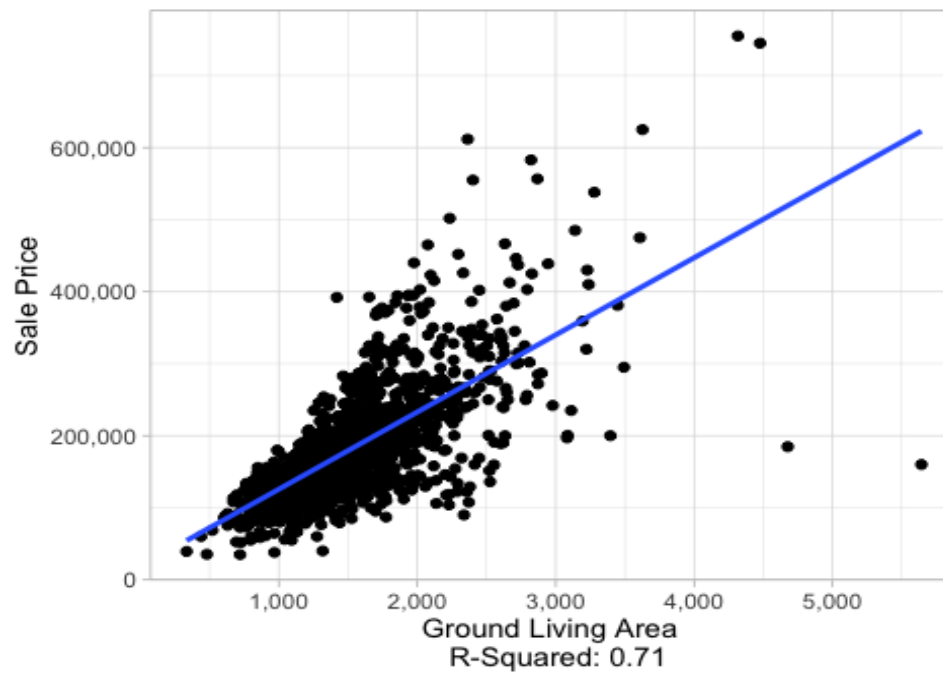
```
## `geom_smooth()` using formula 'y ~ x'
```



#A majority of homes are under \$200,000 and Average square foot of first floor in homes is ~1163 sq.feet.

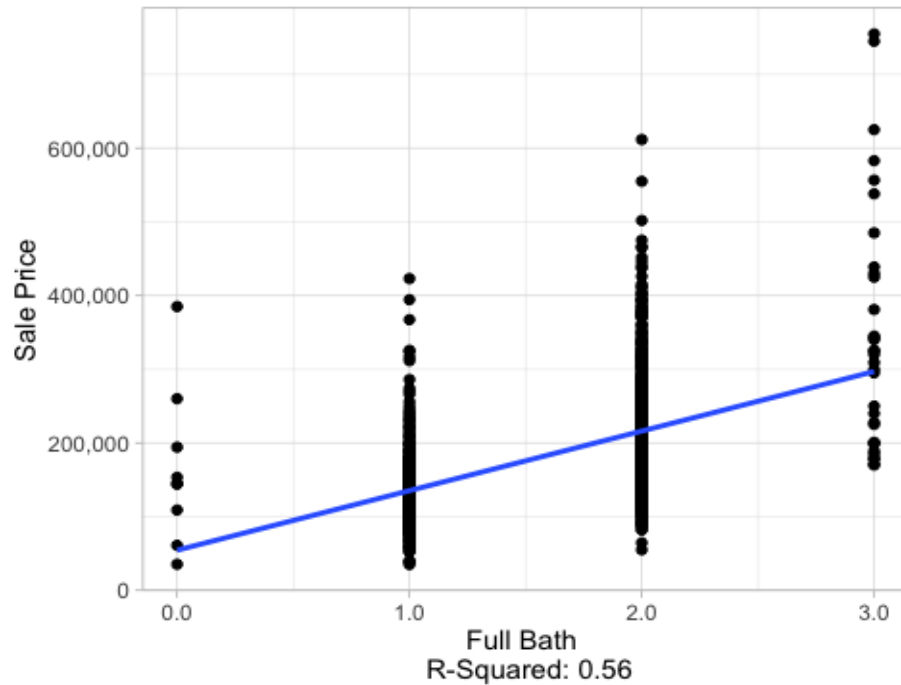
```
makeScatterplots(train_num, train_num$GrLivArea, train_num$SalePrice, "Ground  
Living Area", "Sale Price")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



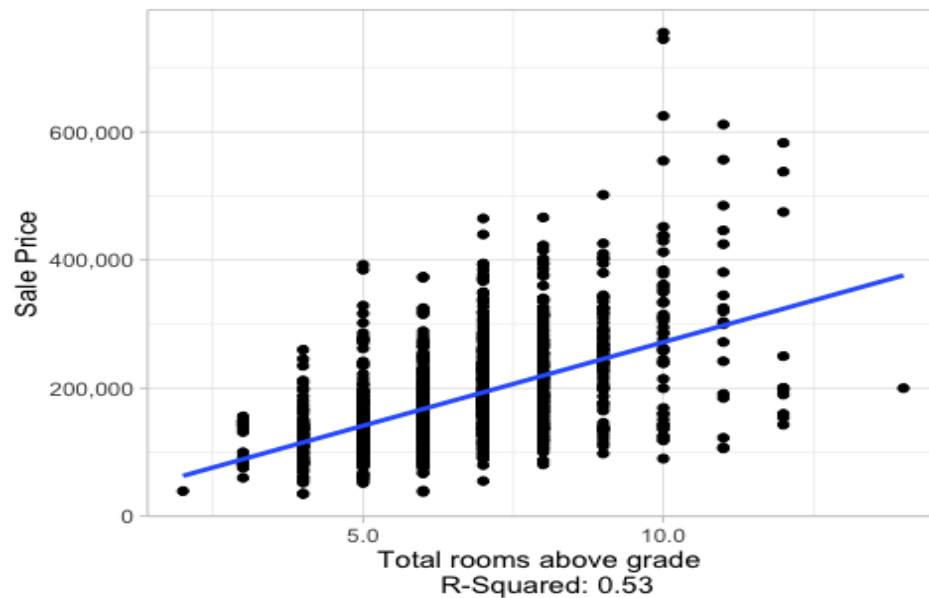
#Living areas tend to be around an average of 1500 sq.feet for most homes.
`makeScatterplots(train_num, train_num$FullBath, train_num$SalePrice, "Full Bath", "Sale Price")`

`## `geom_smooth()` using formula 'y ~ x'`



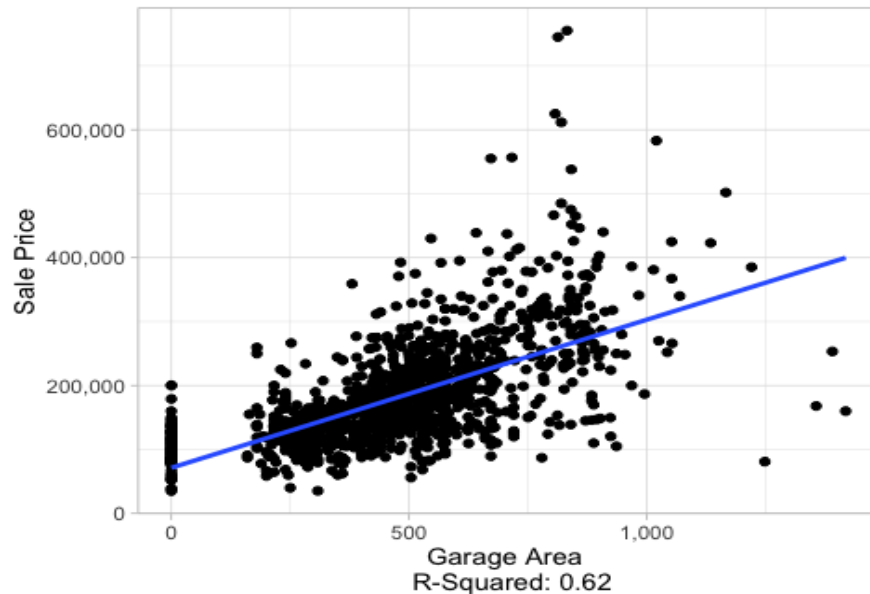
`makeScatterplots(train_num, train_num$TotRmsAbvGrd, train_num$SalePrice, "Total rooms above grade", "Sale Price")`

`## `geom_smooth()` using formula 'y ~ x'`



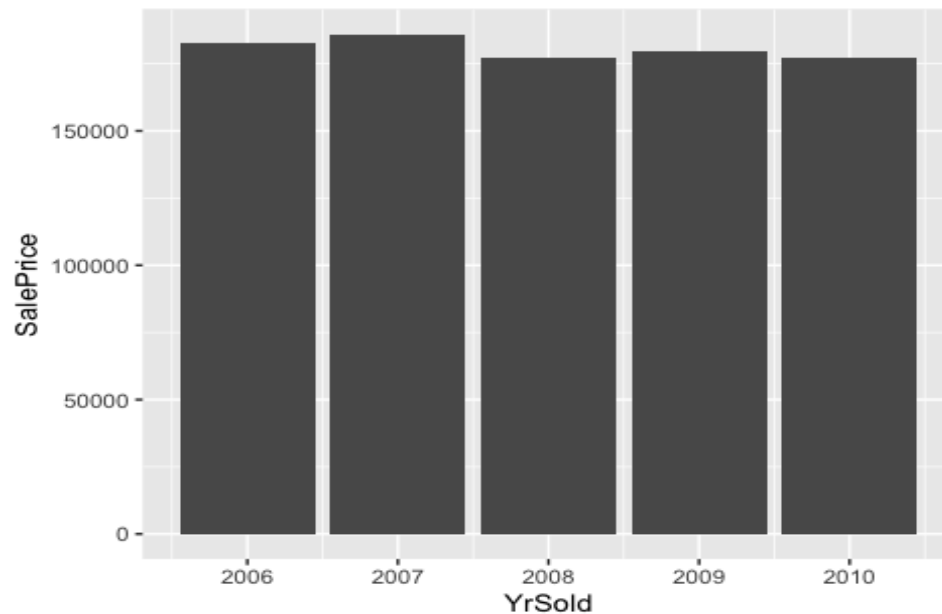
```
makeScatterplots(train_num, train_num$GarageArea, train_num$SalePrice,
"Garage Area", "Sale Price")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



#The sale prices of home is higher for garage areas between 750 to 1000 sq.feet. However, there are a few outliers where sale price drops for homes where garage area is greater than ~1000 sq.feet.

```
ggplot(Traindata, aes(x=YrSold, y=SalePrice)) + stat_summary(fun.y="mean",
geom="bar")
```



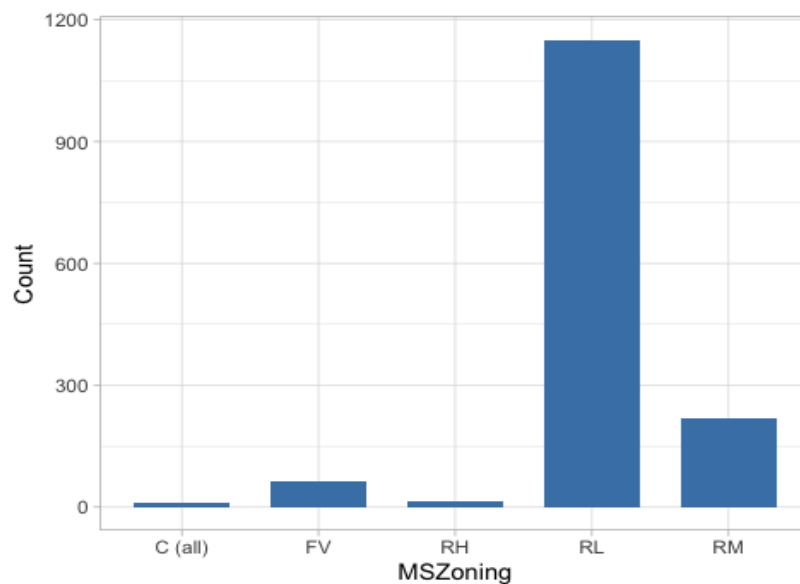
#Notice the drop in average sale of home price in year 2008, the housing market bubble crashed when Case-Shiller home price index reported it's

Largest price drop.

#Categorical Variables

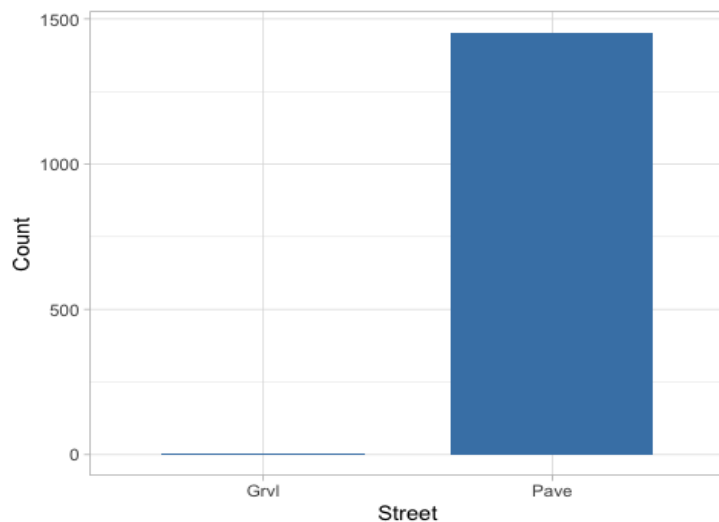
Let's make some barplots for categorical variables to get a deeper insight / understanding of our data.

```
makeBarplots <- function(dataframe,x.variable, xlabel, ylabel)
{
  p = ggplot(dataframe, aes(x=factor(x.variable))) +
  geom_bar(stat = "count", width=0.7, fill="steelblue") +
  ylab(ylabel) + xlab(xlabel) +
  theme_light()
  return(p)
}
makeBarplots(Traindata, Traindata$MSZoning, "MSZoning", "Count")
```

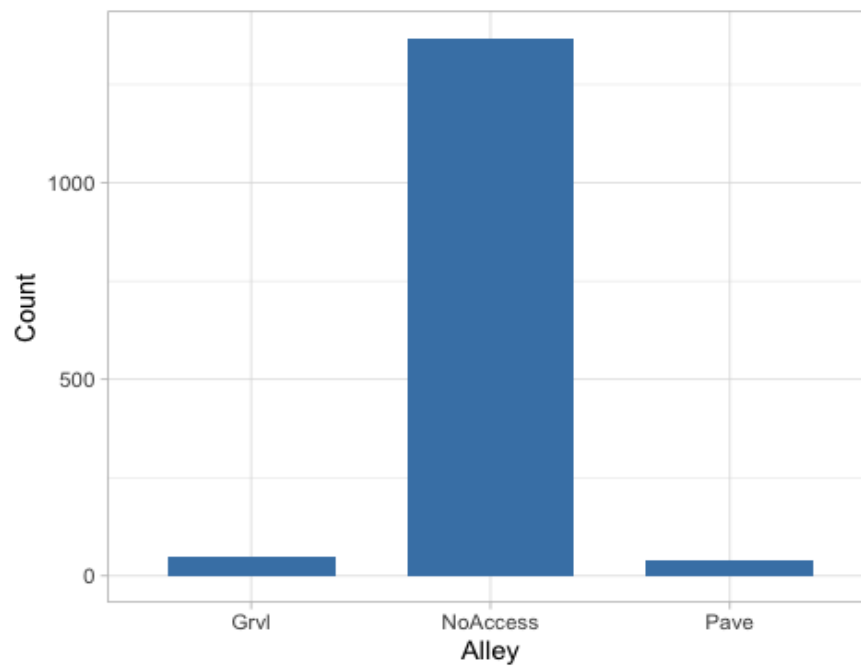


An overwhelming majority of homes are in Residential Low Density zone.

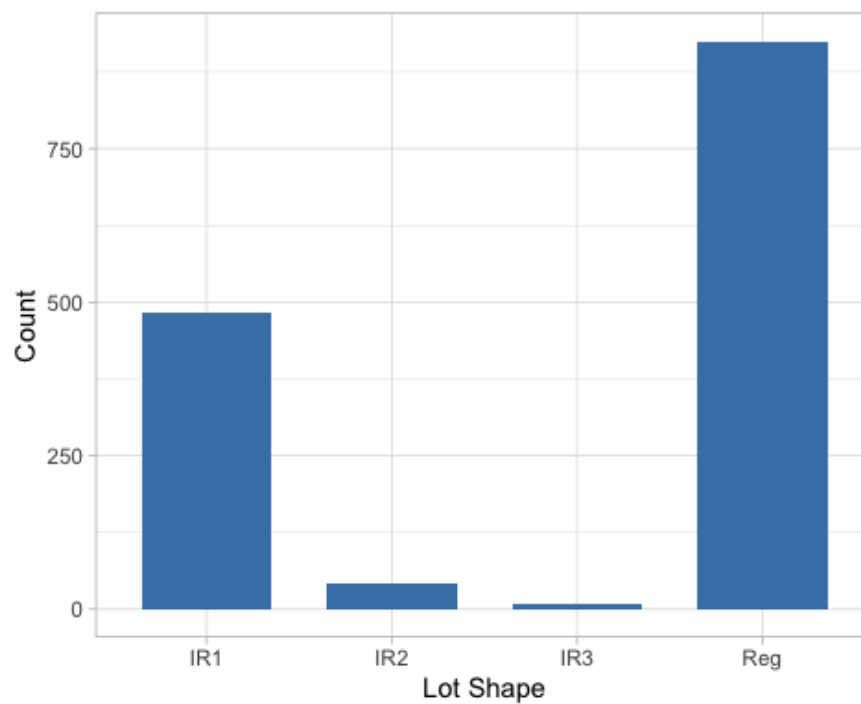
```
makeBarplots(Traindata, Traindata$Street, "Street", "Count")
```



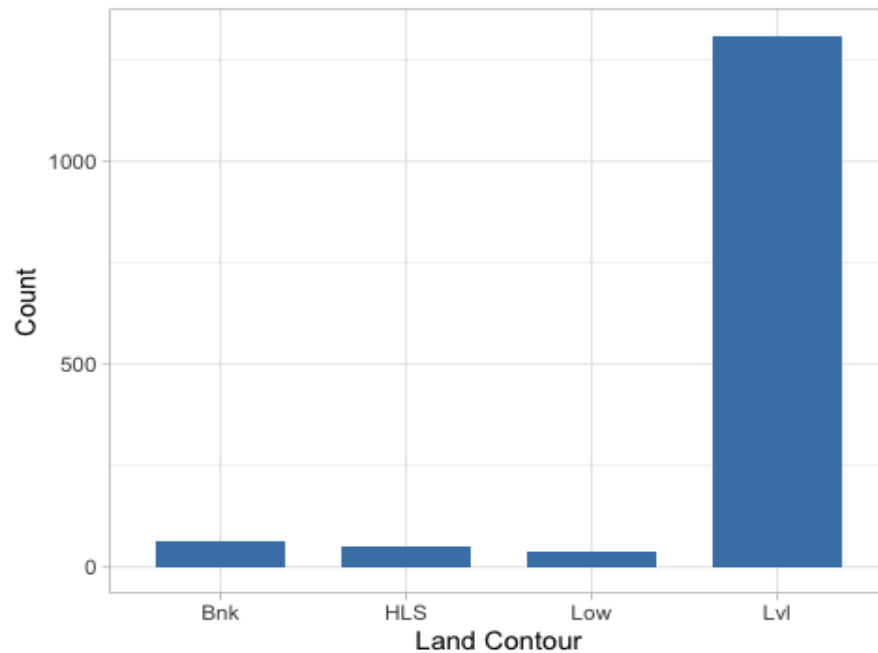
```
makeBarplots(Traindata, Traindata$Alley, "Alley", "Count")
```



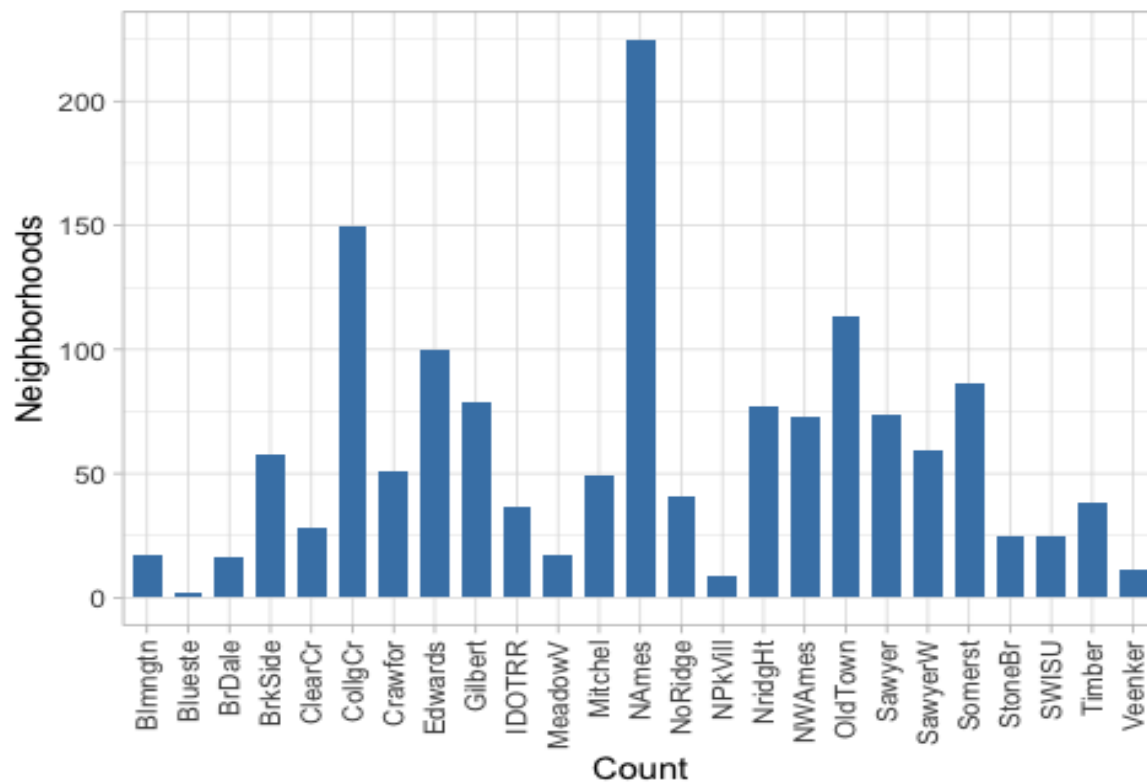
```
makeBarplots(Traindata, Traindata$LotShape, "Lot Shape", "Count")
```



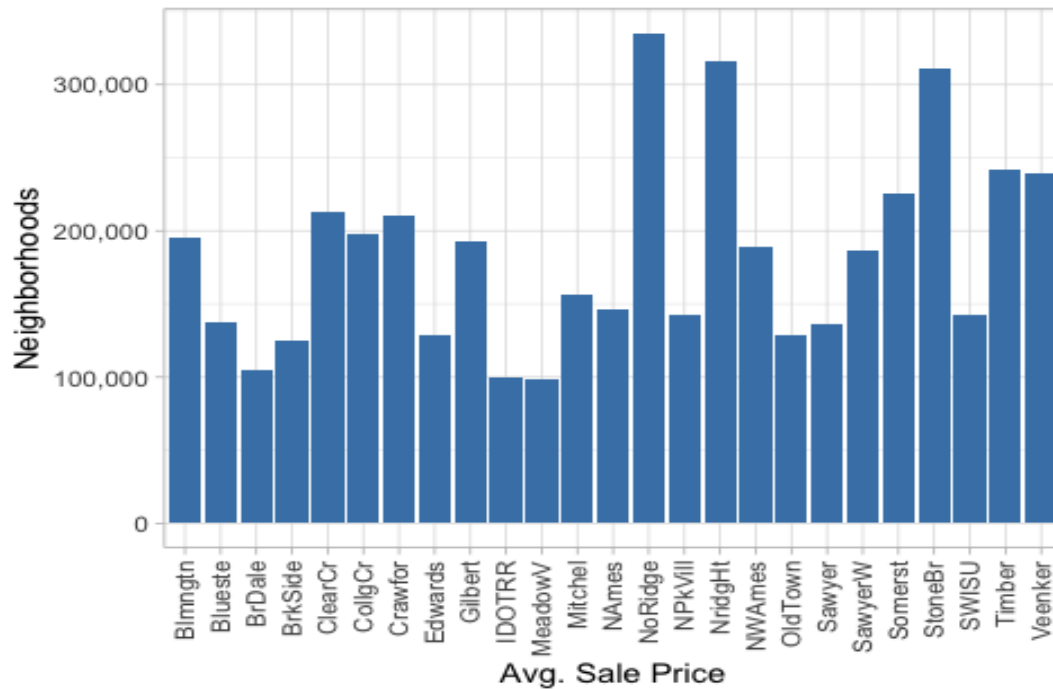
```
makeBarplots(Traindata, Traindata$LandContour, "Land Contour", "Count")
```



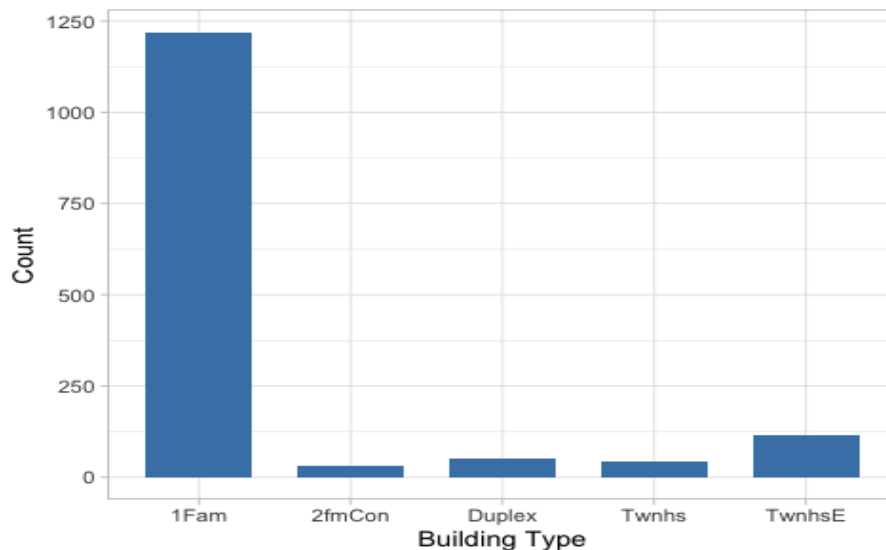
```
ggplot(Traindata, aes(x=factor(Neighborhood))) +
  geom_bar(stat = "count", width=0.7, fill="steelblue") +
  ylab("Neighborhoods") + xlab("Count") +
  theme_light() +
  theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))
```



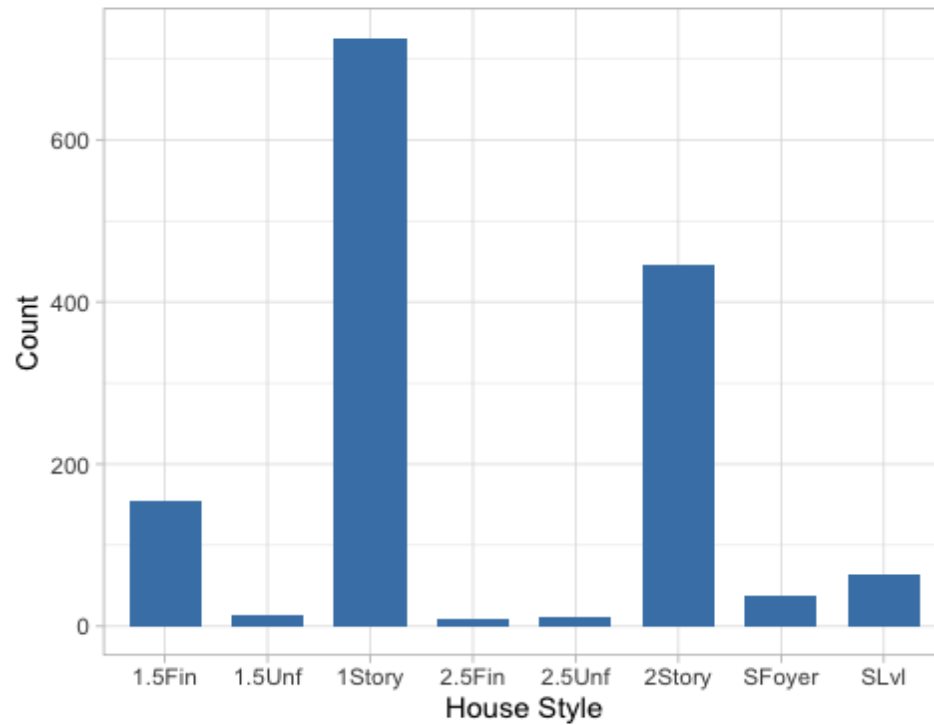
```
ggplot(Traindata, aes(x=factor(Neighborhood),y=SalePrice)) +
  stat_summary(fun.y="mean", geom="bar", fill="steelblue") +
  ylab("Neighborhoods") +
  xlab("Avg. Sale Price") +
  theme_light() +
  theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5)) +
  scale_y_continuous(labels = comma)
```



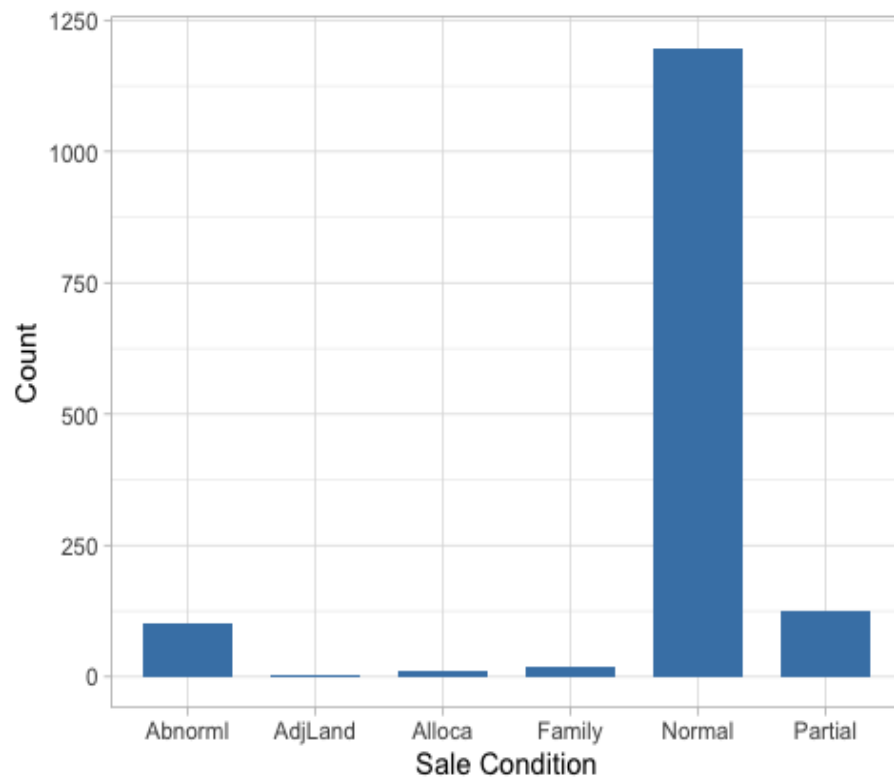
Area around the Iowa University, neighborhoods of College Creek and just N. Ames, just north of the university have high concentration of homes.
makeBarplots(Traindata, Traindata\$BldgType, "Building Type", "Count")



```
makeBarplots(Traindata, Traindata$HouseStyle, "House Style", "Count")
```



```
makeBarplots(Traindata, Traindata$SaleCondition, "Sale Condition", "Count")
```

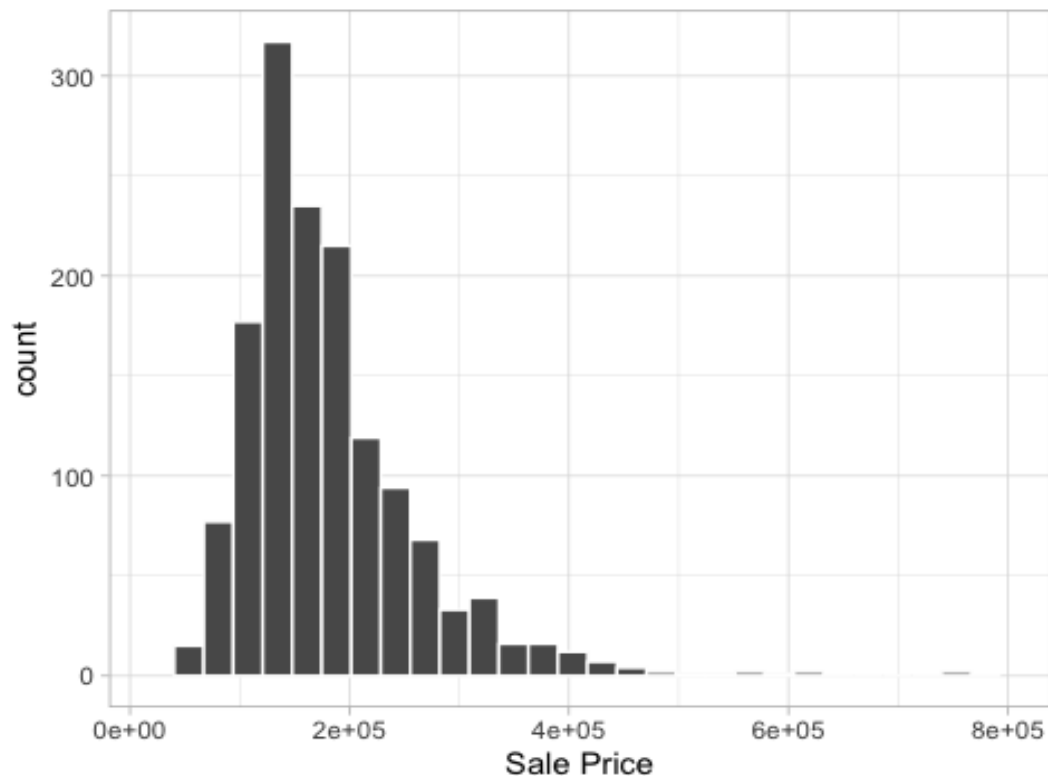


Let's plot our dependent variable sales price

```
summary(Traindata$SalePrice)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 34900 129975 163000 180921 214000 755000
```

```
ggplot(data=Traindata, aes(SalePrice)) +  
  geom_histogram(col = "white") +  
  theme_light() +  
  xlim(20000, 800000) +  
  xlab("Sale Price")
```

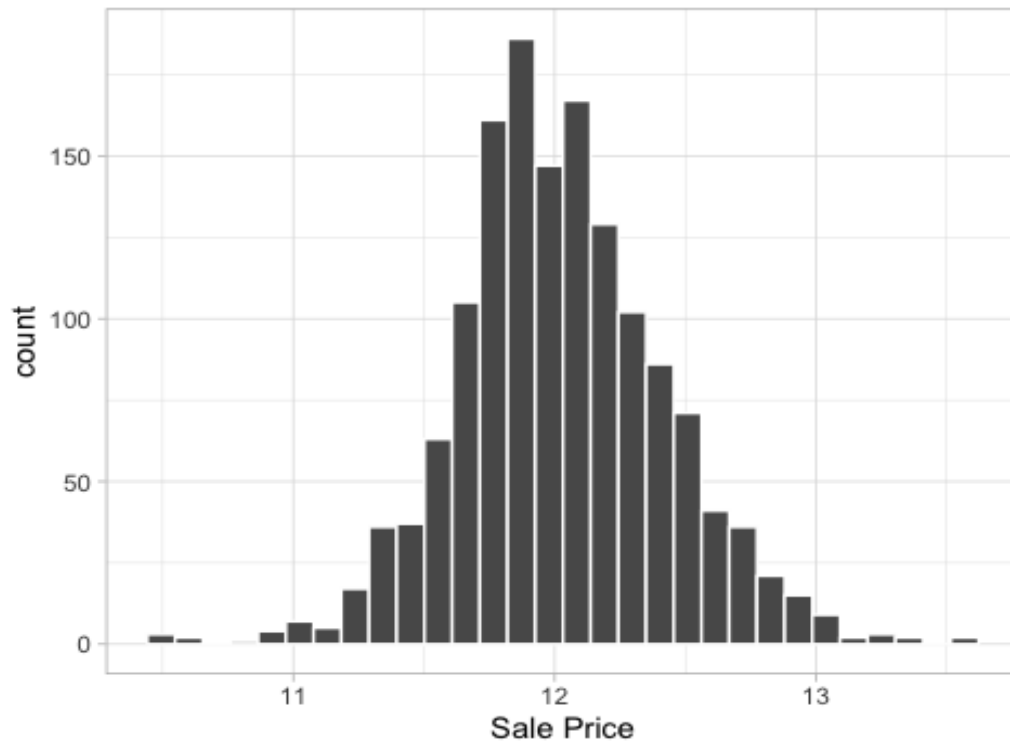


Sale Price appears to be heavily skewed.

#We will log transform the variable to obtain a normal distribution of our dependent variable.

This is to maintain positivity of the sale price variable, in all likelihood, sale price of a home will never be a negative value.

```
ggplot(data=Traindata, aes(log(SalePrice))) +  
  geom_histogram(col = "white") +  
  theme_light() +  
  xlab("Sale Price")
```

Linear Regression

model <-

```
lm(SalePrice~OverallQual+GrLivArea+GarageArea+GarageCars+TotalBsmtSF+X1stFlrS
F, data = Traindata)
```

summary(model)

##

Call:

```
## lm(formula = SalePrice ~ OverallQual + GrLivArea + GarageArea +
##     GarageCars + TotalBsmtSF + X1stFlrSF, data = Traindata)
```

##

Residuals:

```
##      Min       1Q   Median       3Q      Max
## -473373 -19732   -1080    16922  288035
```

##

Coefficients:

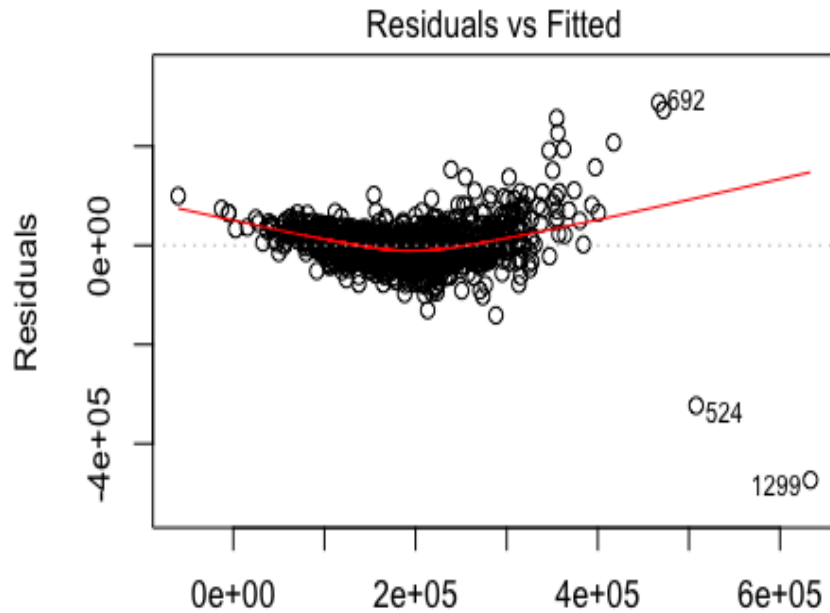
```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.027e+05  4.904e+03 -20.932  < 2e-16 ***
## OverallQual  2.400e+04  1.083e+03  22.150  < 2e-16 ***
## GrLivArea    4.312e+01  2.679e+00  16.095  < 2e-16 ***
## GarageArea   1.566e+01  1.047e+01   1.495   0.1350
## GarageCars   1.452e+04  3.019e+03   4.809  1.68e-06 ***
## TotalBsmtSF  2.439e+01  4.318e+00   5.649  1.94e-08 ***
## X1stFlrSF    1.119e+01  5.032e+00   2.223   0.0264 *
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

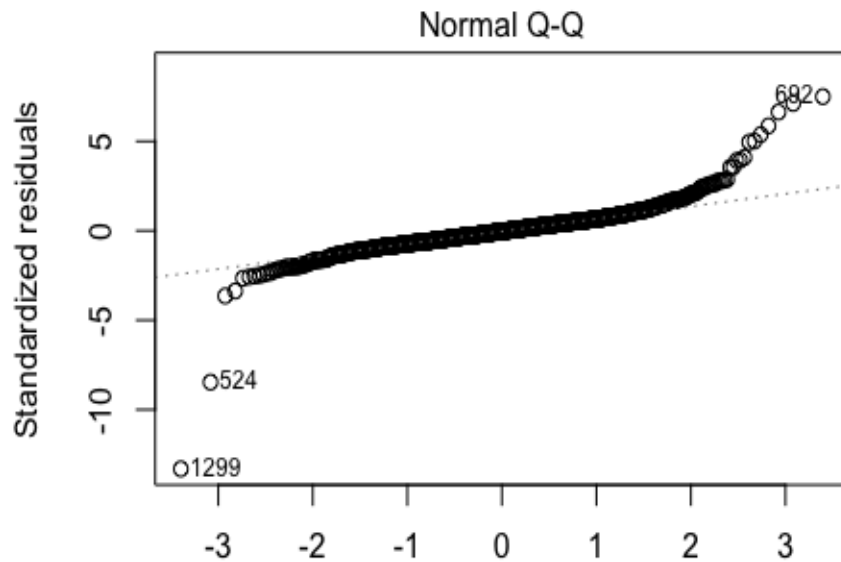
##

```
## Residual standard error: 38840 on 1453 degrees of freedom
## Multiple R-squared:  0.7619, Adjusted R-squared:  0.7609
## F-statistic: 775 on 6 and 1453 DF,  p-value: < 2.2e-16
```

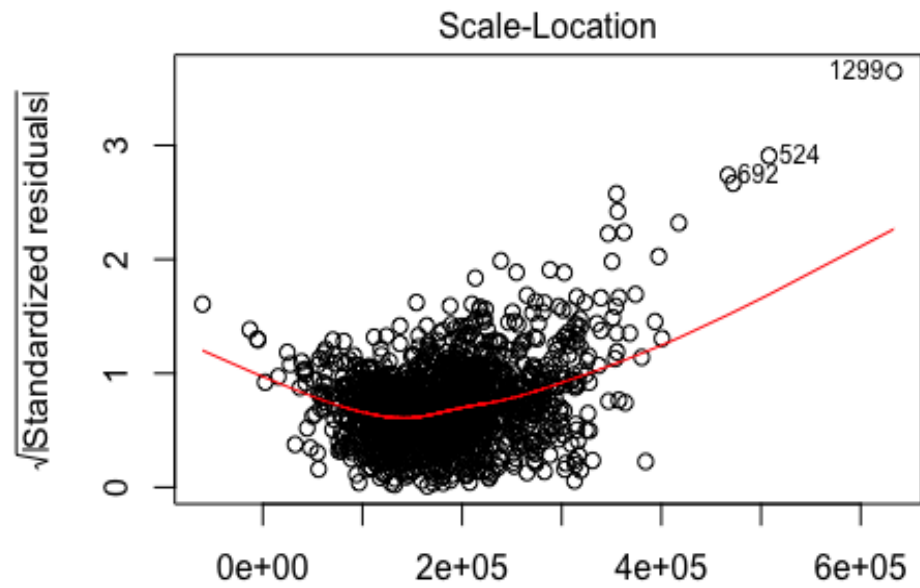
```
plot(model)
```



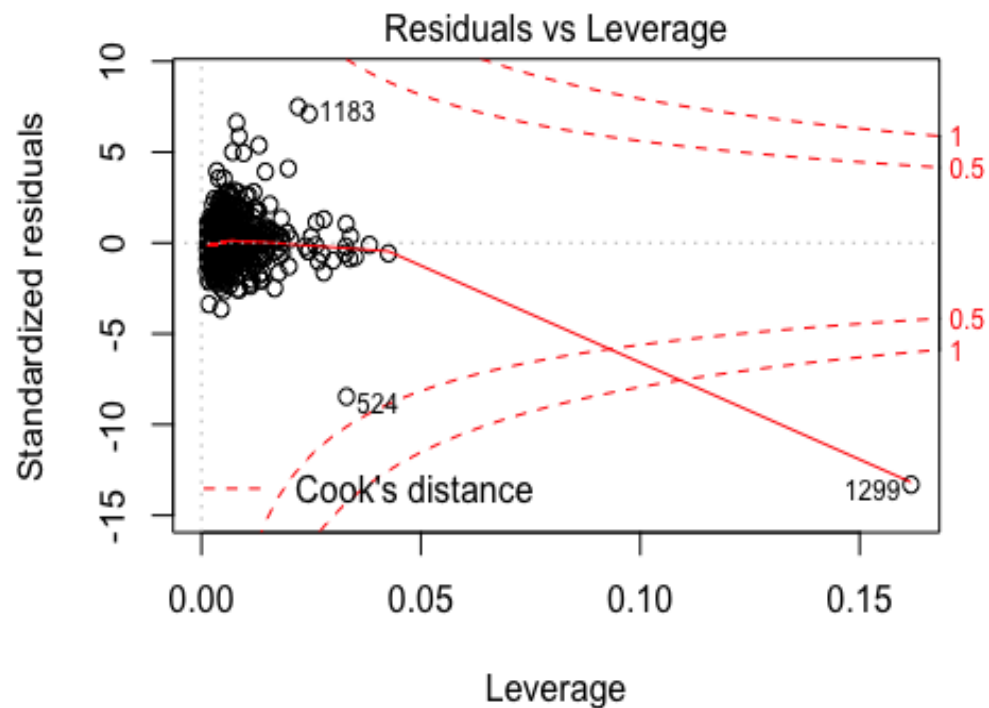
Fitted values
 $\text{ilePrice} \sim \text{OverallQual} + \text{GrLivArea} + \text{GarageArea} + \text{GarageCars} + \text{T}$



Theoretical Quantiles
 $\text{ilePrice} \sim \text{OverallQual} + \text{GrLivArea} + \text{GarageArea} + \text{GarageCars} + \text{T}$



Fitted values
 $\text{ilePrice} \sim \text{OverallQual} + \text{GrLivArea} + \text{GarageArea} + \text{GarageCars} + \text{T}$



Leverage
 $\text{ilePrice} \sim \text{OverallQual} + \text{GrLivArea} + \text{GarageArea} + \text{GarageCars} + \text{T}$