House Price Precictions R Notebook

# load libraries

library(knitr)  
library(ggplot2)  
library(plyr)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:plyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(corrplot)

## corrplot 0.84 loaded

library(gridExtra)

##   
## Attaching package: 'gridExtra'

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

library(scales)  
library(Rmisc)

## Loading required package: lattice

library(ggrepel)  
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:gridExtra':  
##   
## combine

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

## The following object is masked from 'package:ggplot2':  
##   
## margin

library(psych)

##   
## Attaching package: 'psych'

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

## The following objects are masked from 'package:scales':  
##   
## alpha, rescale

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

library(xgboost)

##   
## Attaching package: 'xgboost'

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

# Read Train and Test data

train <- read.csv("train.csv", stringsAsFactors = F)  
test <- read.csv("test.csv", stringsAsFactors = F)

dim(train)

## [1] 1460 81

str(train[,c(1:10, 81)])

## 'data.frame': 1460 obs. of 11 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 : chr "RL" "RL" "RL" "RL" ...  
## $ 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 : chr "Pave" "Pave" "Pave" "Pave" ...  
## $ Alley : chr NA NA NA NA ...  
## $ LotShape : chr "Reg" "Reg" "IR1" "IR1" ...  
## $ LandContour: chr "Lvl" "Lvl" "Lvl" "Lvl" ...  
## $ Utilities : chr "AllPub" "AllPub" "AllPub" "AllPub" ...  
## $ SalePrice : int 208500 181500 223500 140000 250000 143000 307000 200000 129900 118000 ...

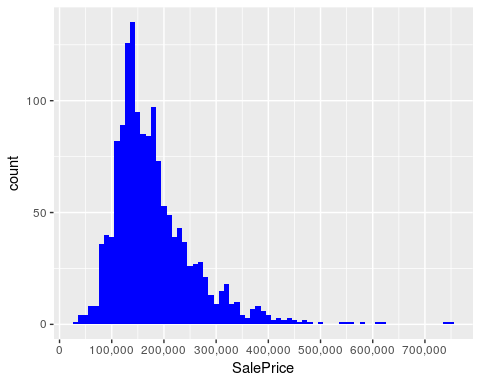
# #Getting rid of the IDs but keeping the test IDs in a vector. These are needed to compose the submission file  
test\_labels <- test$Id  
test$Id <- NULL  
train$Id <- NULL

test$SalePrice <- NA  
all <- rbind(train, test)  
dim(all)

## [1] 2919 80

## Exploring some of the most important variables

ggplot(data = all[!is.na(all$SalePrice),], aes(x = SalePrice))+  
 geom\_histogram(fill = "blue", binwidth = 10000)+  
 scale\_x\_continuous(breaks = seq(0, 800000, by = 100000), labels = comma)



summary(all$SalePrice)

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

### The character variables need some work before I can use them. To get a feel for the dataset, I decided to first see which numeric variables have a high correlation with the SalePrice.

### Correlations with SalePrice

#index vector numeric variables  
numericVars <- which(sapply(all, is.numeric))  
#saving names vector for use later on  
numericNames <- names(numericVars)  
## There are 37 numeric variables

all\_numVar <- all[, numericVars]  
#correlations of all numeric variables  
cor\_numVar <- cor(all\_numVar, use = "pairwise.complete.obs")  
#sort on decreasing correlations with SalePrice  
cor\_sorted <- as.matrix(sort(cor\_numVar[, 'SalePrice'], decreasing = TRUE))  
#select only high corelations  
CorHigh <- names(which(apply(cor\_sorted, 1, function(x) abs(x) > 0.5)))  
cor\_numVar <- cor\_numVar[CorHigh, CorHigh]  
  
corrplot.mixed(cor\_numVar, tl.col = "black", tl.pos = "lt")

