Final Project IS605

Sharon Morris
5/06/2017

Table of Contents

Kaggle.com Home Price Completion

1. You are to register for Kaggle.com (free) and compete in the House Prices: Advanced Regression Techniques competition.

House Prices: Advanced Regression Techniques competition 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 playground competition's 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 competition challenges you to predict the final price of each home.

Load Library

```
library(formattable)
library(stats)
require(dplyr)
library(MASS)

devtools::install_github("rubenarslan/formr")
```

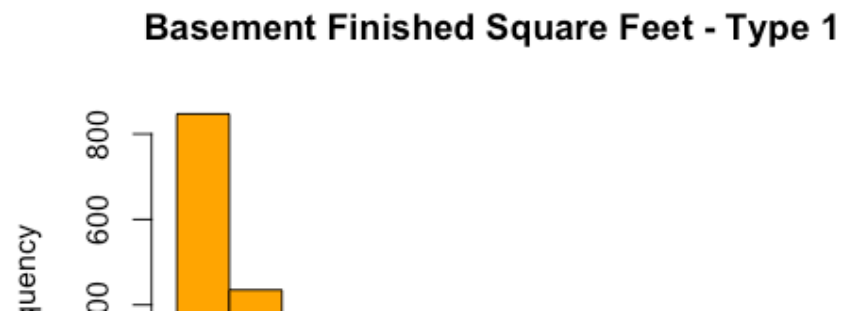
1. Define Variables

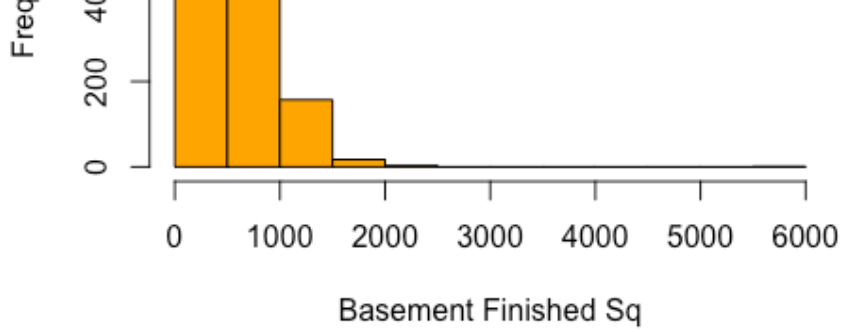
Pick one of the quantitative independent variables from the training data set (train.csv) , and define that variable as X. Pick SalePrice as the dependent variable, and define it as Y for the next analysis.

```
train <-
read.csv(paste0("https://raw.githubusercontent.com/indianspice/IS605/master/Final%20Project/train.csv"), stringsAsFactors = F)

#Quantitative independent variable BsmtFinSF1: Type 1 finished square feet
X <- train$BsmtFinSF1
Y <- train$SalePrice

#Summary of independent and dependent variables
summary(X)
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0   383.5   443.6   712.2  5644.0
summary(Y)
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   34900  130000  163000  180900  214000  755000
hist(train$BsmtFinSF1,
main = "Basement Finished Square Feet - Type 1",
xlab = "Basement Finished Sq",
col = "orange")
```






```
hist(train$SalePrice,
main = "Sale Price",
xlab = "Sale Price",
col = "gold")
```

2. Probability

Calculate as a minimum the below probabilities a through c. Assume the small letter $"x"$ is estimated as the 4th quartile of the variable, and the small letter $"y"$ is estimated as the 2d quartile of the variable. Interpret the meaning of all probabilities. a.

```
quantile(X) #Independent variable
##     0%    25%    50%    75%   100%
##   0.00   0.00 383.50 712.25 5644.00
quantile(Y)
##     0%    25%    50%    75%   100%
##  34900 129975 163000 214000 755000
```

The probability of a house at the second quartile has an overall condition rating at the third quartile is 13.5%

```
(sum(Y > 163000 & X > 6) / length(X)) / (sum(Y > 163000) / length(Y))
## [1] 0.7046703
```

```
sum(X > 6 & Y > 163000) / length(X)
## [1] 0.3513699
```

```
(sum(Y > 163000 & X < 6) / length(X)) / (sum(Y > 163000) / length(Y))
## [1] 0.2953297
```

Count Table

```
#Sum rows with sales prices > 163000
(YQ2 <- sum(Y > 163000))
## [1] 728
```

```
#Count of all rows sales prices
(YTotal <- length(Y))
## [1] 1460
#Count of all rows of overall condition
(XTotal <- length(X))
## [1] 1460
#Rows greater than 163,000 and greater than 6
(Ygt <- sum(Y > 163000 & X > 6))
## [1] 513
#Rows greater than 163,000 and less than 6
(Ylt <- sum(Y > 163000 & X < 6))
## [1] 215
sum(train['BsmtFinSF1'] <= 6 & train['SalePrice'] > 163000)
## [1] 215
sum(train['BsmtFinSF1'] > 6 & train['SalePrice'] <= 163000)
## [1] 479
sum(train['BsmtFinSF1'] > 6 & train['SalePrice'] > 163000)
## [1] 513
```

X/Y	<=2nd Quartile	>2nd Quartile	Total
<= 3rd Quartile	728	215	943
> 3rd Quartile	479	513	992
Total	1,207	788	1,995

Does splitting the training data in this fashion make them independent? observations above the 3rd quartile of observations above the 2nd quartile of

indicates splitting the data in this fashion did not make them independent.

```
#Assign to variables
PA <- 299/1657
PB <- 728/1657

#P(A) * P(B)
(AXB <- PA*PB)
## [1] 0.07927889
#Convert to a percent
percent(AXB)
## [1] 7.93%
#P(A|B)
PAl <- 98/1657
PBl <- 1657/728

(PABl <- PAl*PBl)
## [1] 0.1346154
percent(PABl)
## [1] 13.46%
```

Evaluate by running a Chi Square test for association.

The chi-square value is 17.895. Since the p-value is less than the significance level of 0.05, reject the null hypothesis and conclude that the two variables are in fact dependent.

```
#Pearson's Chi Square test with Yates -- 2x2 table
chisq.test(matrix(c(728, 630, 201, 98), ncol=2))
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  matrix(c(728, 630, 201, 98), ncol = 2)
## X-squared = 17.895, df = 1, p-value = 2.335e-05
```

3. Descriptive and Inferential Statistics

Transform both variables simultaneously using Box-Cox transformations. You might have to research this. Using the transformed variables, run a correlation analysis and interpret. Test the hypothesis that the correlation between these variables is 0 and provide a 99% confidence interval. Discuss the meaning of your analysis.

```
#Examine variable names
names(train)
##  [1] "Id"           "MSSubClass"   "MSZoning"     "LotFrontage"
##  [5] "LotArea"      "Street"       "Alley"        "LotShape"
##  [9] "LandContour"  "Utilities"    "LotConfig"    "LandSlope"
## [13] "Neighborhood" "Condition1"   "Condition2"   "BldgType"
## [17] "HouseStyle"   "OverallQual"  "OverallCond"  "YearBuilt"
```

```
## [21] "YearRemodAdd"  "RoofStyle"    "RoofMatl"     "Exterior1st"
## [25] "Exterior2nd"   "MasVnrType"   "MasVnrArea"   "ExterQual"
## [29] "ExterCond"     "Foundation"   "BsmtQual"     "BsmtCond"
## [33] "BsmtExposure"  "BsmtFinType1" "BsmtFinSF1"   "BsmtFinType2"
## [37] "BsmtFinSF2"    "BsmtUnfSF"    "TotalBsmtSF"  "Heating"
## [41] "HeatingQC"     "CentralAir"   "Electrical"   "X1stFlrSF"
## [45] "X2ndFlrSF"     "LowQualFinSF" "GrLivArea"    "BsmtFullBath"
## [49] "BsmtHalfBath"  "FullBath"     "HalfBath"     "BedroomAbvGr"
## [53] "KitchenAbvGr"  "KitchenQual"  "TotRmsAbvGrd" "Functional"
## [57] "Fireplaces"    "FireplaceQu"  "GarageType"   "GarageYrBlt"
## [61] "GarageFinish"  "GarageCars"   "GarageArea"   "GarageQual"
## [65] "GarageCond"    "PavedDrive"   "WoodDeckSF"   "OpenPorchSF"
## [69] "EnclosedPorch" "X3SsnPorch"   "ScreenPorch"  "PoolArea"
## [73] "PoolQC"        "Fence"        "MiscFeature"  "MiscVal"
## [77] "MoSold"        "YrSold"       "SaleType"     "SaleCondition"
## [81] "SalePrice"
detach("package:dplyr", character.only = TRUE)
library("dplyr", character.only = TRUE)
```

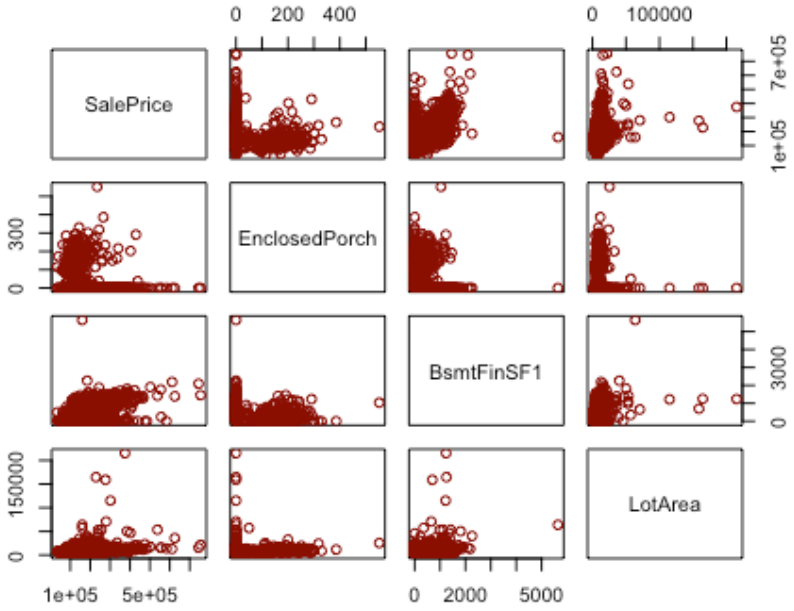
```
#Selecting 4 numeric variables
numericV <- select(train, SalePrice, EnclosedPorch, BsmtFinSF1, LotArea)
numericV <- na.omit(numericV) #Remove nas
glimpse(numericV)
## Observations: 1,460
## Variables: 4
## $ SalePrice     <int> 208500, 181500, 223500, 140000, 250000, 143000, ...
## $ EnclosedPorch <int> 0, 0, 0, 272, 0, 0, 0, 228, 205, 0, 0, 0, 0, 0, ...
## $ BsmtFinSF1    <int> 706, 978, 486, 216, 655, 732, 1369, 859, 0, 851,...
## $ LotArea       <int> 8450, 9600, 11250, 9550, 14260, 14115, 10084, 10...
```

Provide univariate descriptive statistics

```
#Mean,median,25th and 75th quartiles,min,max
summary(numericV)
##    SalePrice      EnclosedPorch     BsmtFinSF1       LotArea
##  Min.   : 34900   Min.   :  0.00   Min.   :   0.0   Min.   :  1300
##  1st Qu.:129975   1st Qu.:  0.00   1st Qu.:   0.0   1st Qu.:  7554
##  Median :163000   Median :  0.00   Median : 383.5   Median :  9478
##  Mean   :180921   Mean   : 21.95   Mean   : 443.6   Mean   : 10517
##  3rd Qu.:214000   3rd Qu.:  0.00   3rd Qu.: 712.2   3rd Qu.: 11602
##  Max.   :755000   Max.   :552.00   Max.   :5644.0   Max.   :215245
```

Provide a scatterplot of X and Y.

```
#Look at all the relationships in selected variables
plot(numericV, col="darkred")
```



Test the hypothesis that the correlation between these variables is 0 and provide a 99% confidence interval.

It appeas that OverallCond and SalesPrice have a negative correlation

```
var <- data.frame(train$BsmtFinSF1, train$SalePrice)
cor(var)
##                  train.BsmtFinSF1 train.SalePrice
## train.BsmtFinSF1        1.0000000       0.3864198
## train.SalePrice         0.3864198       1.0000000
cor.test(train$BsmtFinSF1, train$SalePrice, conf.level = 0.99)
##
##  Pearson's product-moment correlation
##
## data:  train$BsmtFinSF1 and train$SalePrice
## t = 15.998, df = 1458, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 99 percent confidence interval:
##  0.3275690 0.4422838
## sample estimates:
##       cor
## 0.3864198
```

4. Linear Algebra and Correlation

Invert your correlation matrix. (This is known as the precision matrix and contains variance inflation factors on the diagonal.) Multiply the correlation matrix by the precision matrix, and then multiply the precision matrix by the correlation matrix.

The table below show correlations coefficients between the possible pairs of variables

```
d <- select(train, BsmtFinSF1, SalePrice)

(corMatrix <- cor(d))  #Correlation matrix
##             BsmtFinSF1 SalePrice
## BsmtFinSF1   1.0000000 0.3864198
## SalePrice    0.3864198 1.0000000
round(corMatrix, 3)
##             BsmtFinSF1 SalePrice
## BsmtFinSF1       1.000     0.386
## SalePrice        0.386     1.000
#The inverse of the correlation matrix
(preMatrix <- solve(corMatrix))
##             BsmtFinSF1  SalePrice
## BsmtFinSF1   1.1755305 -0.4542483
## SalePrice   -0.4542483  1.1755305
round(preMatrix, 3)
##             BsmtFinSF1 SalePrice
## BsmtFinSF1       1.176    -0.454
## SalePrice       -0.454     1.176
```

Multiply the correlation matrix by the precision matrix, and the precision matrix by the correlation matrix

```
corMatrix %*% preMatrix #Correlation * precision
##              BsmtFinSF1 SalePrice
## BsmtFinSF1 1.000000e+00         0
## SalePrice  5.551115e-17         1
preMatrix %*% corMatrix #Precision * correlation
##             BsmtFinSF1    SalePrice
## BsmtFinSF1           1 5.551115e-17
## SalePrice            0 1.000000e+00
```

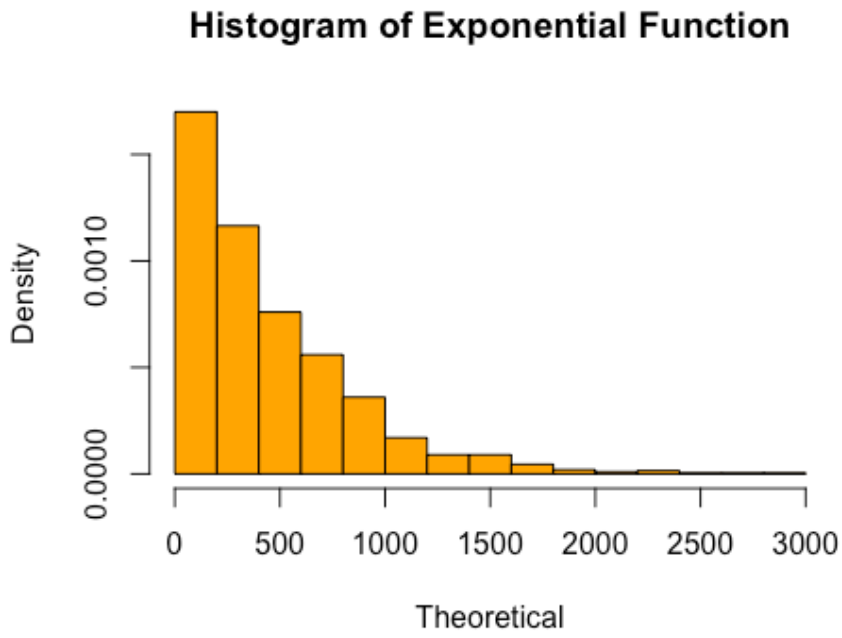
5. Calculus-Based Probability & Statistics

For your non-transformed independent variable, location shift it so that the minimum value is above zero. Then load the MASS package and run fitdistr to fit a density function of your choice. (See $\url{https://stat.ethz.ch/R-manual/R-devel/library/MASS/html/fitdistr.html$}). Find the optimal value of the parameters for this distribution, and then take 1000 samples from this distribution (e.g.,) for an exponential). Plot a histogram and compare it with a histogram of your non-transformed original variable.

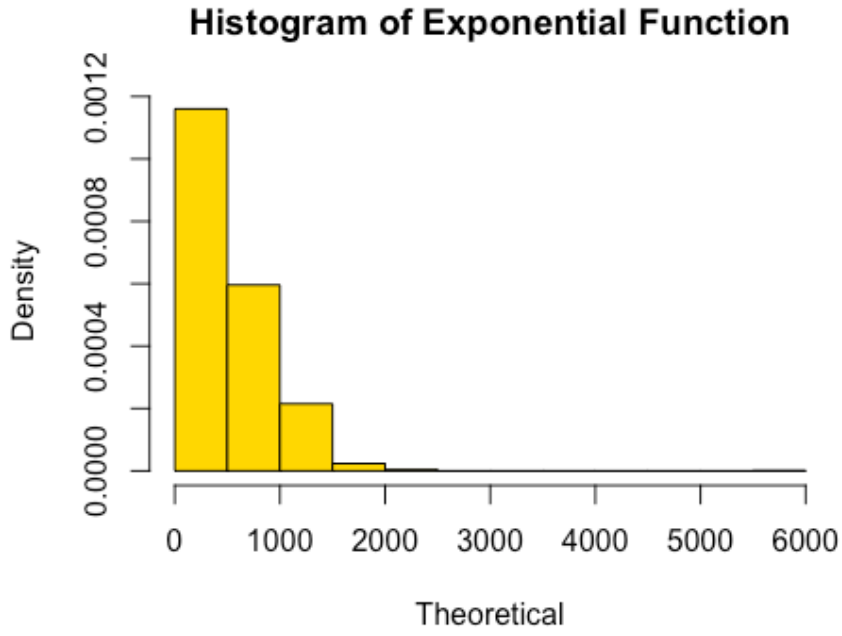
```
#Fits the exponential distribution to the data
fitDist <- fitdistr(X,"exponential")
(lam <- fitDist$estimate)  #Lambda
##        rate
```

```
## 0.002254081
#Get values from the exponential distribution
n <- rexp(1000, lam)

#Histogram of the exponential
hist(n, prob=TRUE, col="orange", main = "Histogram of Exponential Function",
xlab="Theoretical")
```




```
#Hitogram of the OverCond
hist(train$BsmtFinSF1, prob=TRUE, col="gold", main = "Histogram of Exponential Function",
xlab="Theoretical")
```



6. Modeling

Build some type of regression model and submit your model to the competition board. Provide your complete model summary and results with analysis.

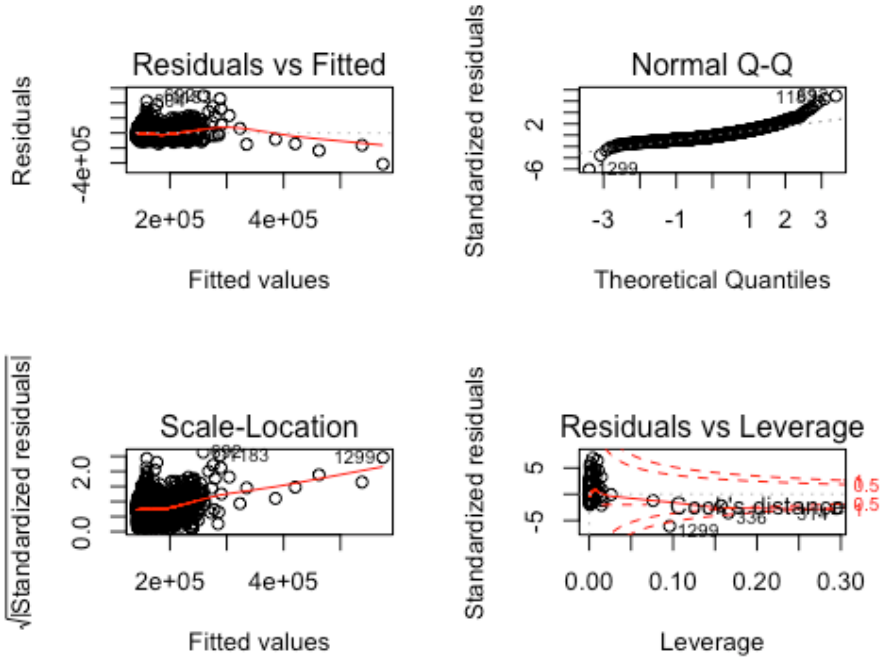
= Sale Price - Response variable = BasementFinSF1 - First regression coefficient = Lot Area - Second regression coefficient

```
#Correlation
cor(na.omit(cbind(numericV, Y)))
```

```
##                SalePrice EnclosedPorch BsmtFinSF1     LotArea          Y
## SalePrice      1.0000000   -0.12857796  0.3864198  0.26384335  1.0000000
## EnclosedPorch -0.1285780    1.00000000 -0.1023033 -0.01833973 -0.1285780
## BsmtFinSF1     0.3864198   -0.10230331  1.0000000  0.21410313  0.3864198
## LotArea        0.2638434   -0.01833973  0.2141031  1.00000000  0.2638434
## Y              1.0000000   -0.12857796  0.3864198  0.26384335  1.0000000
```

R-squared = 7.54% explains the variability in caused by the variables used in this model. The P-value < .001 indicates the regression coefficient variables are statistically significant

```
fit <- lm(SalePrice ~ BsmtFinSF1 + LotArea, data = train)
(sum <- summary(fit)); par(mfrow = c(2, 2)); plot(fit)
##
## Call:
## lm(formula = SalePrice ~ BsmtFinSF1 + LotArea, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -414754  -47826  -15561   33074  496521
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.383e+05  3.087e+03  44.810  < 2e-16 ***
## BsmtFinSF1  6.023e+01  4.221e+00  14.270  < 2e-16 ***
## LotArea     1.511e+00  1.929e-01   7.833 9.11e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 71830 on 1457 degrees of freedom
## Multiple R-squared:  0.1837, Adjusted R-squared:  0.1826
## F-statistic: 163.9 on 2 and 1457 DF,  p-value: < 2.2e-16
```




```
test <-
read.csv(paste0("https://raw.githubusercontent.com/indianspice/IS605/master/Final%20Project/test.csv"), stringsAsFactors = F)

predictor1 <- test$BsmtFinSF1
predictor1 <- na.omit(predictor1) #Remove missing values
predictor2 <- test$LotArea
predictor2 <- na.omit(predictor2)

rFit <- as.data.frame(cbind("Id" = test[,"Id"],
"SalePrice" = fit$coefficients[1] +
                      fit$coefficients[2] * predictor1 +
                      fit$coefficients[3] * predictor2))
## Warning in fit$coefficients[1] + fit$coefficients[2] * predictor1 + fit
## $coefficients[3] * : longer object length is not a multiple of shorter
## object length
write.csv(rFit, "KaggleSubmission.csv", row.names = F)
dim(rFit)
```

```
dim(rFit)
## [1] 1459    2
```

Kaggle Information

Refernces http://stackoverflow.com/questions/24202120/dplyrselect-function-clashes-with-massselect http://stackoverflow.com/questions/11995832/inverse-of-matrix-in-r http://www.theanalysisfactor.com/interpreting-regression-coefficients/ https://rstudio-pubs-static.s3.amazonaws.com/234598_13317c1c83534900b9c36eda2f205a87.html#modeling

Report your Kaggle.com