DATA 605 : Final Exam : Problem3

Ramnivas Singh

12/14/2021

Problem 3 - House Prices: Advanced Regression Techniques competition

You are to compete in the House Prices: Advanced Regression Techniques competition https://www.kaggle.com/c/house-prices-advanced-regression-techniques (https://www.kaggle.com/c/house-prices-advanced-regression-techniques) . I want you to do the following.

1. Descriptive and Inferential Statistics. Provide univariate descriptive statistics and appropriate plots for the training data set. Provide a scatterplot matrix for at least two of the independent variables and the dependent variable. Derive a correlation matrix for any three quantitative variables in the dataset. Test the hypotheses that the correlations between each pairwise set of variables is 0 and provide an 80% confidence interval. Discuss the meaning of your analysis. Would you be worried about familywise error? Why or why not?

2. Linear Algebra and Correlation. Invert your correlation matrix from above. (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. Conduct LU decomposition on the matrix.

3. Calculus-Based Probability & Statistics. Many times, it makes sense to fit a closed form distribution to data. Select a variable in the Kaggle.com training dataset that is skewed to the right, shift it so that the minimum value is absolutely above zero if necessary. Then load the MASS package and run fitdistr to fit an exponential probability density function. (See https://stat.ethz.ch/R-manual/Rdevel/library/MASS/html/fitdistr.html (https://stat.ethz.ch/R-manual/Rdevel/library/MASS/html/fitdistr.html)). Find the optimal value of λ for this distribution, and then take 1000 samples from this exponential distribution using this value (e.g., rexp(1000, λ)). Plot a histogram and compare it with a your original variable. Using the exponential pdf, find the 5th and 95th percentiles using the cumulative distribution function (CDF). Also generate a 95% confidence interval from the empirical data, assuming normality. Finally, provide the empirical 5th percentile and 95th percentile of the data. Discuss.

4. Modeling. Build some type of multiple regression model and submit your model to the competition board. Provide your complete model summary and results with analysis. Report your Kaggle.com user name and score.

Load Dataset

```
house_prices.train<-read.table("https://raw.githubusercontent.com/rnivas2028/MSDS/Data605/Final/
        train.csv"
                      ,sep=",",header=T,stringsAsFactors = T)
house_prices.test<-read.table("https://raw.githubusercontent.com/rnivas2028/MSDS/Data605/Final/t
        est.csv"
                      ,sep=",",header=T,stringsAsFactors = T)
#Print top 5 records from training and test data
kable(head(house_prices.train[,1:5],5))
```

Id	MSSubClass	MSZoning	LotFrontage	LotArea
1	60	RL	65	8450
2	20	RL	80	9600
3	60	RL	68	11250
4	70	RL	60	9550
5	60	RL	84	14260

```
kable(head(house_prices.test[,11:15],5))
```

LotConfig	LandSlope	Neighborhood	Condition1	Condition2
Inside	Gtl	NAmes	Feedr	Norm
Corner	Gtl	NAmes	Norm	Norm
Inside	Gtl	Gilbert	Norm	Norm
Inside	Gtl	Gilbert	Norm	Norm
Inside	Gtl	StoneBr	Norm	Norm

1. Descriptive and Inferential Statistics

```
# Lets pick fields for working data set
house_prices_working_data <- house_prices.train[c('LotArea','GrLivArea','SalePrice')]
kable(round(describe(house_prices.train)[c(2,3,4,7,8,9,10,11,12,13)],2))
```

	n	mean	sd	mad	min	max	range	skew	kurtosis	se
Id	1460	730.50	421.61	541.15	1	1460	1459	0.00	-1.20	11.03
MSSubClass	1460	56.90	42.30	44.48	20	190	170	1.40	1.56	1.11
MSZoning*	1460	4.03	0.63	0.00	1	5	4	-1.73	6.25	0.02
LotFrontage	1201	70.05	24.28	16.31	21	313	292	2.16	17.34	0.70
LotArea	1460	10516.83	9981.26	2962.23	1300	215245	213945	12.18	202.26	261.22
Street*	1460	2.00	0.06	0.00	1	2	1	-15.49	238.01	0.00
Alley*	91	1.45	0.50	0.00	1	2	1	0.20	-1.98	0.05
LotShape*	1460	2.94	1.41	0.00	1	4	3	-0.61	-1.60	0.04
LandContour*	1460	3.78	0.71	0.00	1	4	3	-3.16	8.65	0.02
Utilities*	1460	1.00	0.03	0.00	1	2	1	38.13	1453.00	0.00

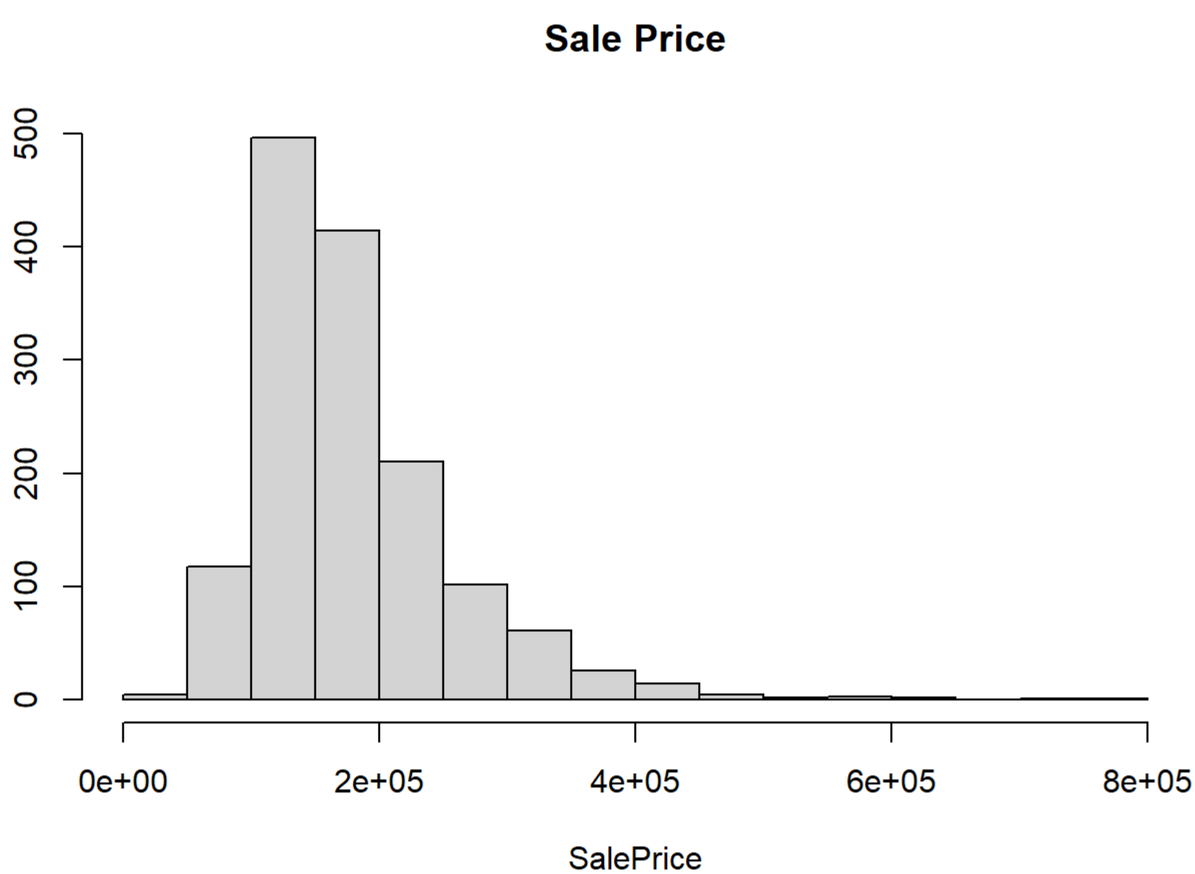
	n	mean	sd	mad	min	max	range	skew	kurtosis	se
LotConfig*	1460	4.02	1.62	0.00	1	5	4	-1.13	-0.59	0.04
LandSlope*	1460	1.06	0.28	0.00	1	3	2	4.80	24.47	0.01
Neighborhood*	1460	13.15	5.89	7.41	1	25	24	0.02	-1.06	0.15
Condition1*	1460	3.03	0.87	0.00	1	9	8	3.01	16.34	0.02
Condition2*	1460	3.01	0.26	0.00	1	8	7	13.14	247.54	0.01
BldgType*	1460	1.49	1.20	0.00	1	5	4	2.24	3.41	0.03
HouseStyle*	1460	4.04	1.91	1.48	1	8	7	0.31	-0.96	0.05
OverallQual	1460	6.10	1.38	1.48	1	10	9	0.22	0.09	0.04
OverallCond	1460	5.58	1.11	0.00	1	9	8	0.69	1.09	0.03
YearBuilt	1460	1971.27	30.20	37.06	1872	2010	138	-0.61	-0.45	0.79
YearRemodAdd	1460	1984.87	20.65	19.27	1950	2010	60	-0.50	-1.27	0.54
RoofStyle*	1460	2.41	0.83	0.00	1	6	5	1.47	0.61	0.02
RoofMatl*	1460	2.08	0.60	0.00	1	8	7	8.09	66.28	0.02
Exterior1st*	1460	10.62	3.20	1.48	1	15	14	-0.72	-0.37	0.08
Exterior2nd*	1460	11.34	3.54	2.97	1	16	15	-0.69	-0.52	0.09
MasVnrType*	1452	2.76	0.62	0.00	1	4	3	-0.07	-0.13	0.02
MasVnrArea	1452	103.69	181.07	0.00	0	1600	1600	2.66	10.03	4.75
ExterQual*	1460	3.54	0.69	0.00	1	4	3	-1.83	3.86	0.02
ExterCond*	1460	4.73	0.73	0.00	1	5	4	-2.56	5.29	0.02
Foundation*	1460	2.40	0.72	1.48	1	6	5	0.09	1.02	0.02
BsmtQual*	1423	3.26	0.87	1.48	1	4	3	-1.31	1.27	0.02
BsmtCond*	1423	3.81	0.66	0.00	1	4	3	-3.39	10.14	0.02
BsmtExposure*	1422	3.27	1.15	0.00	1	4	3	-1.15	-0.39	0.03
BsmtFinType1*	1423	3.73	1.83	2.97	1	6	5	-0.02	-1.39	0.05
BsmtFinSF1	1460	443.64	456.10	568.58	0	5644	5644	1.68	11.06	11.94
BsmtFinType2*	1422	5.71	0.94	0.00	1	6	5	-3.56	12.32	0.02
BsmtFinSF2	1460	46.55	161.32	0.00	0	1474	1474	4.25	20.01	4.22
BsmtUnfSF	1460	567.24	441.87	426.99	0	2336	2336	0.92	0.46	11.56
TotalBsmtSF	1460	1057.43	438.71	347.67	0	6110	6110	1.52	13.18	11.48
Heating*	1460	2.04	0.30	0.00	1	6	5	9.83	110.98	0.01

	n	mean	sd	mad	min	max	range	skew	kurtosis	se
HeatingQC*	1460	2.54	1.74	0.00	1	5	4	0.48	-1.51	0.05
CentralAir*	1460	1.93	0.25	0.00	1	2	1	-3.52	10.42	0.01
Electrical*	1459	4.68	1.05	0.00	1	5	4	-3.06	7.49	0.03
X1stFlrSF	1460	1162.63	386.59	347.67	334	4692	4358	1.37	5.71	10.12
X2ndFlrSF	1460	346.99	436.53	0.00	0	2065	2065	0.81	-0.56	11.42
LowQualFinSF	1460	5.84	48.62	0.00	0	572	572	8.99	82.83	1.27
GrLivArea	1460	1515.46	525.48	483.33	334	5642	5308	1.36	4.86	13.75
BsmtFullBath	1460	0.43	0.52	0.00	0	3	3	0.59	-0.84	0.01
BsmtHalfBath	1460	0.06	0.24	0.00	0	2	2	4.09	16.31	0.01
FullBath	1460	1.57	0.55	0.00	0	3	3	0.04	-0.86	0.01
HalfBath	1460	0.38	0.50	0.00	0	2	2	0.67	-1.08	0.01
BedroomAbvGr	1460	2.87	0.82	0.00	0	8	8	0.21	2.21	0.02
KitchenAbvGr	1460	1.05	0.22	0.00	0	3	3	4.48	21.42	0.01
KitchenQual*	1460	3.34	0.83	0.00	1	4	3	-1.42	1.72	0.02
TotRmsAbvGrd	1460	6.52	1.63	1.48	2	14	12	0.67	0.87	0.04
Functional*	1460	6.75	0.98	0.00	1	7	6	-4.08	16.37	0.03
Fireplaces	1460	0.61	0.64	1.48	0	3	3	0.65	-0.22	0.02
FireplaceQu*	770	3.73	1.13	1.48	1	5	4	-0.16	-0.98	0.04
GarageType*	1379	3.28	1.79	0.00	1	6	5	0.76	-1.30	0.05
GarageYrBlt	1379	1978.51	24.69	31.13	1900	2010	110	-0.65	-0.42	0.66
GarageFinish*	1379	2.18	0.81	1.48	1	3	2	-0.35	-1.41	0.02
GarageCars	1460	1.77	0.75	0.00	0	4	4	-0.34	0.21	0.02
GarageArea	1460	472.98	213.80	177.91	0	1418	1418	0.18	0.90	5.60
GarageQual*	1379	4.86	0.61	0.00	1	5	4	-4.43	18.25	0.02
GarageCond*	1379	4.90	0.52	0.00	1	5	4	-5.28	26.77	0.01
PavedDrive*	1460	2.86	0.50	0.00	1	3	2	-3.30	9.22	0.01
WoodDeckSF	1460	94.24	125.34	0.00	0	857	857	1.54	2.97	3.28
OpenPorchSF	1460	46.66	66.26	37.06	0	547	547	2.36	8.44	1.73
EnclosedPorch	1460	21.95	61.12	0.00	0	552	552	3.08	10.37	1.60
X3SsnPorch	1460	3.41	29.32	0.00	0	508	508	10.28	123.06	0.77

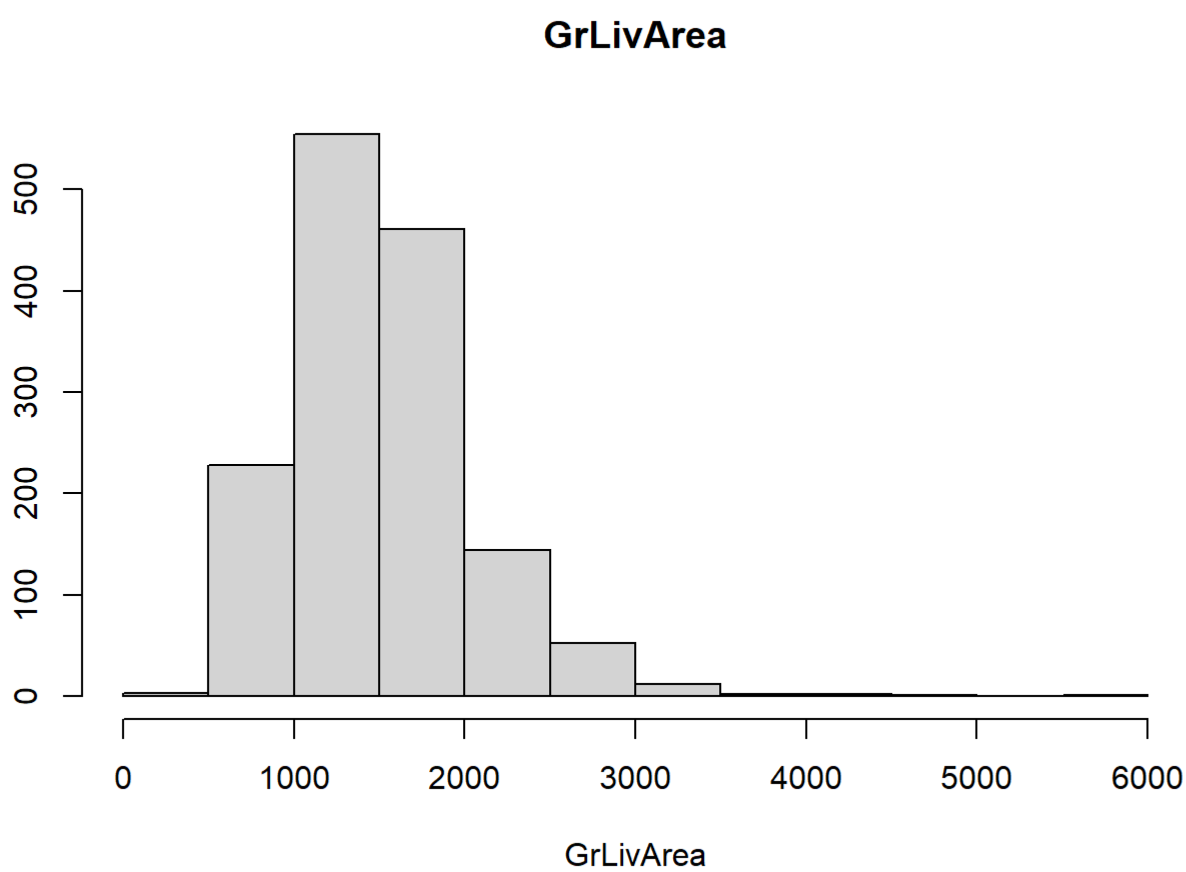
	n	mean	sd	mad	min	max	range	skew	kurtosis	se
ScreenPorch	1460	15.06	55.76	0.00	0	480	480	4.11	18.34	1.46
PoolArea	1460	2.76	40.18	0.00	0	738	738	14.80	222.19	1.05
PoolQC*	7	2.14	0.90	1.48	1	3	2	-0.22	-1.90	0.34
Fence*	281	2.43	0.86	0.00	1	4	3	-0.57	-0.88	0.05
MiscFeature*	54	2.91	0.45	0.00	1	4	3	-2.93	10.71	0.06
MiscVal	1460	43.49	496.12	0.00	0	15500	15500	24.43	697.64	12.98
MoSold	1460	6.32	2.70	2.97	1	12	11	0.21	-0.41	0.07
YrSold	1460	2007.82	1.33	1.48	2006	2010	4	0.10	-1.19	0.03
SaleType*	1460	8.51	1.56	0.00	1	9	8	-3.83	14.57	0.04
SaleCondition*	1460	4.77	1.10	0.00	1	6	5	-2.74	6.82	0.03
SalePrice	1460	180921.20	79442.50	56338.80	34900	755000	720100	1.88	6.50	2079.11

To understand columns from the dataset, lets plot histogram

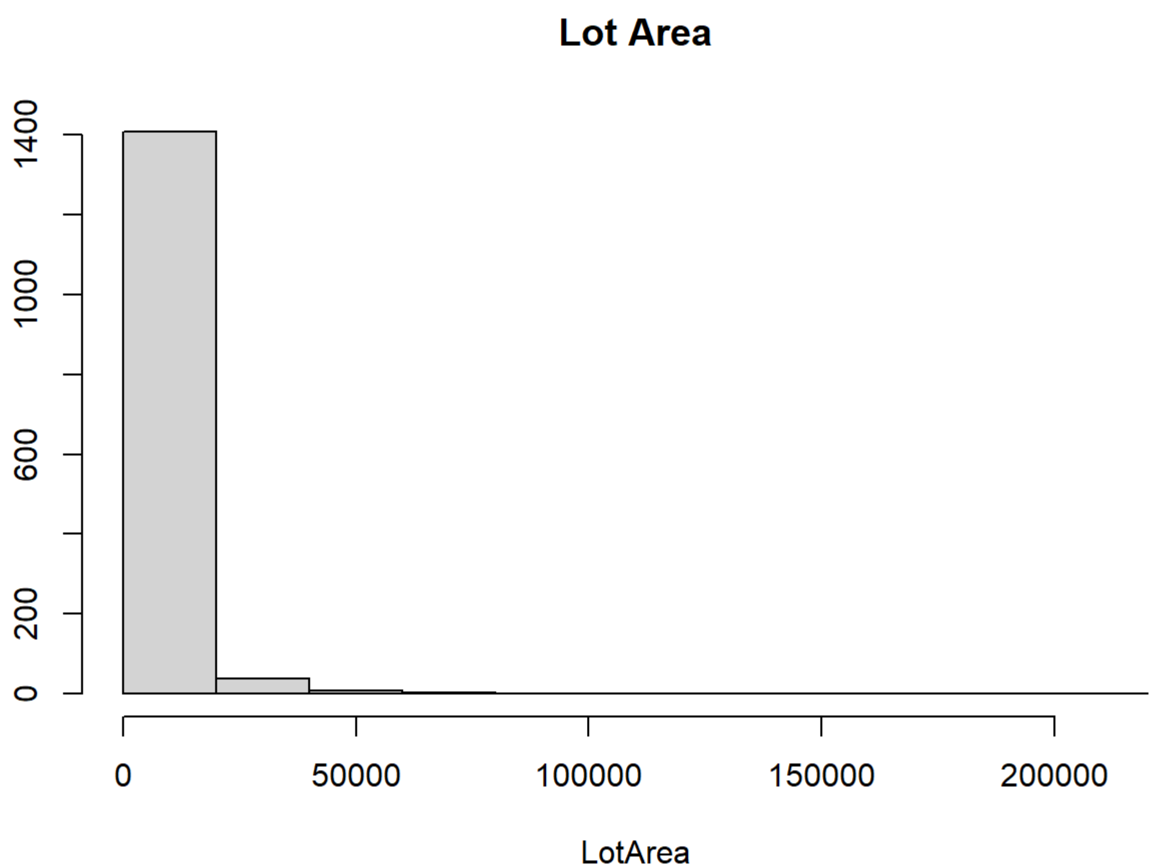
```
hist(house_prices_working_data$SalePrice, main="Sale Price",xlab="SalePrice",ylab="")
```



```r
hist(house_prices_working_data$GrLivArea, main="GrLivArea",xlab="GrLivArea",ylab="")
```

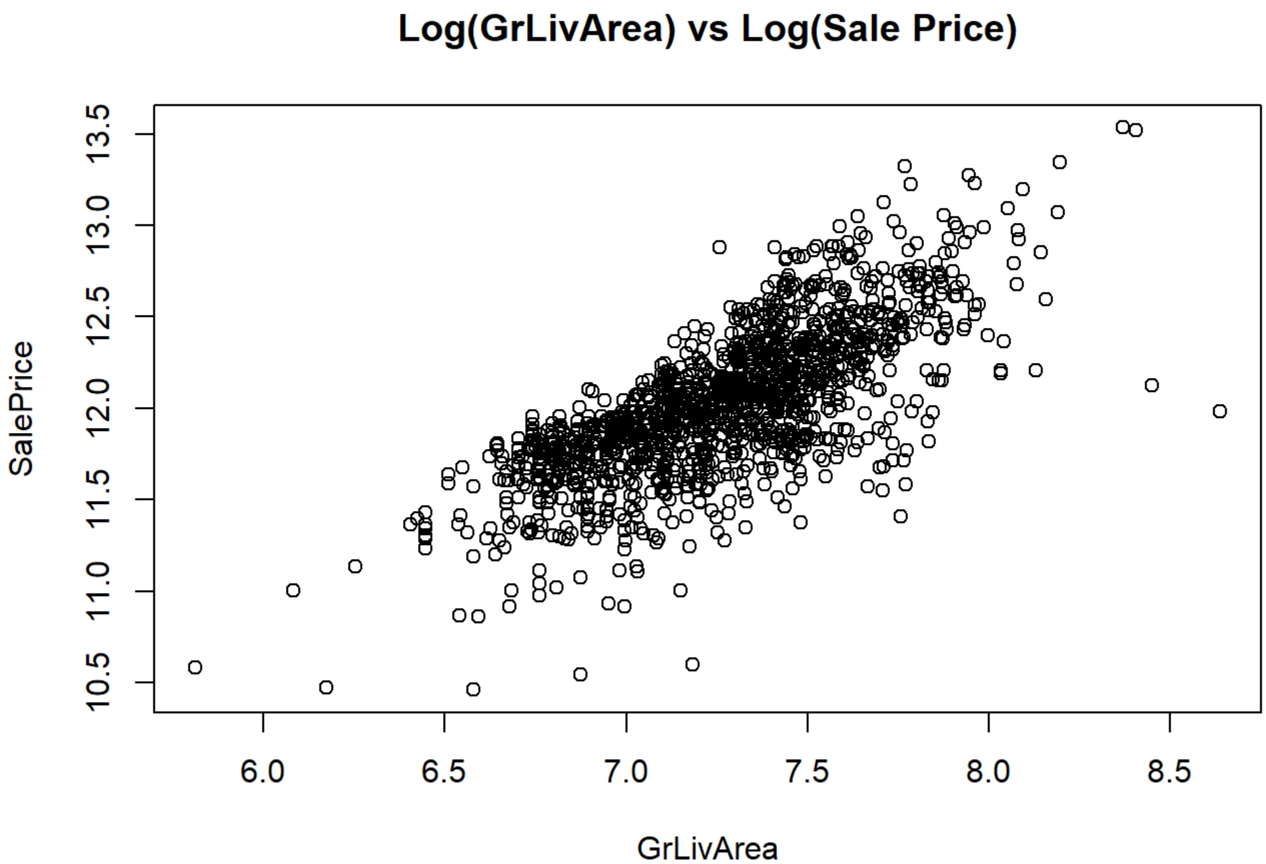



```r
hist(house_prices_working_data$LotArea, main="Lot Area",xlab="LotArea",ylab="")
```

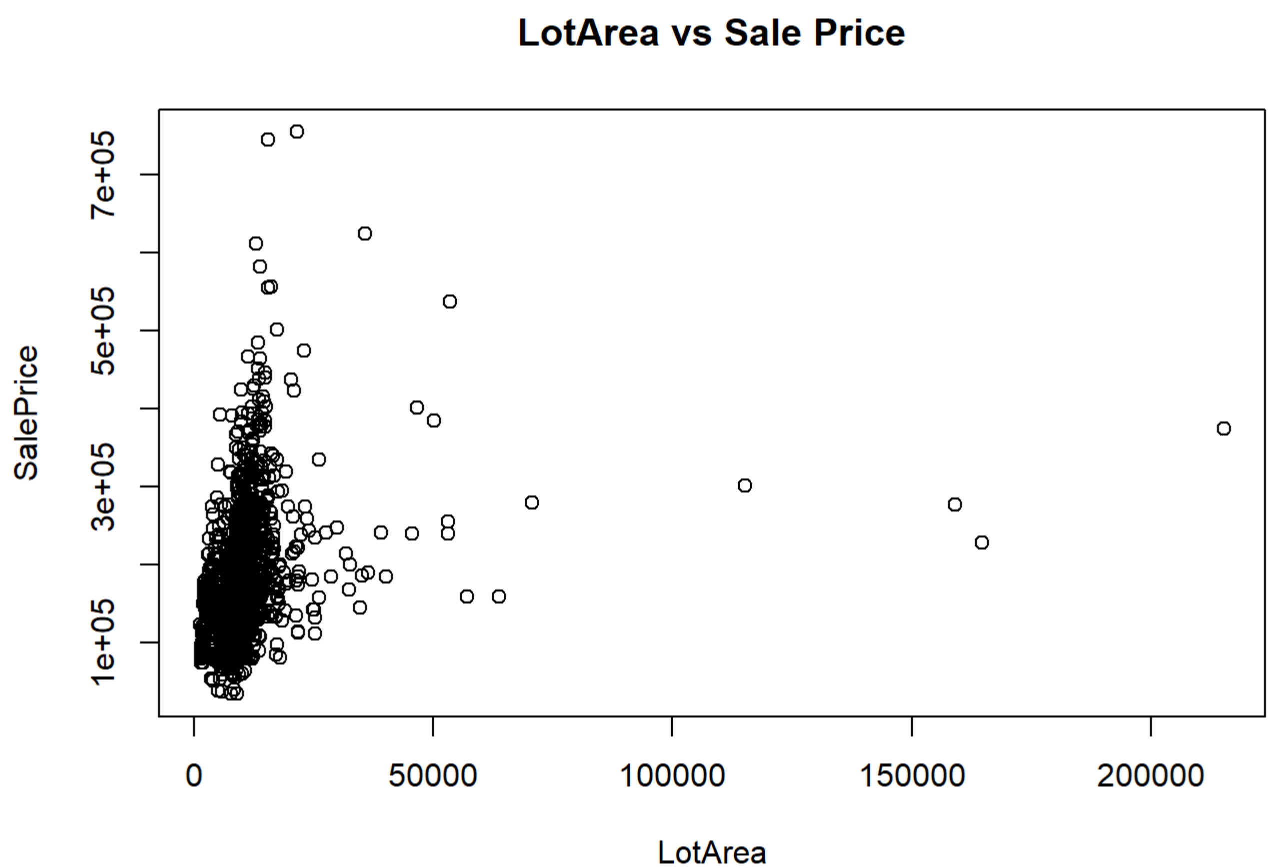


Based on above histogram lot of skew in these few quantitative variables. It's not perfect, but it may be a better way to look at data to be used in a linear regression model.

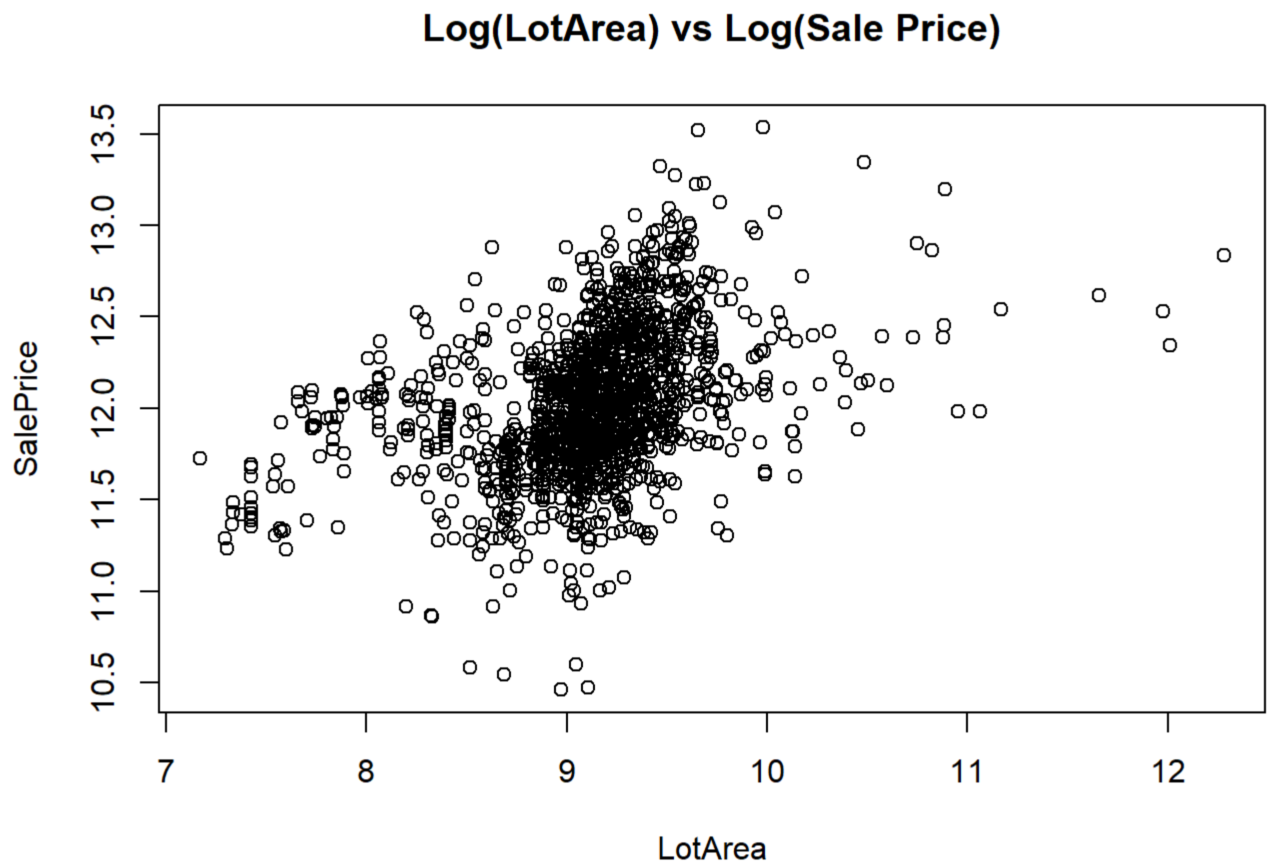
```
plot(log(house_prices_working_data[c("GrLivArea","SalePrice")]),main="Log(GrLivArea) vs Log(Sale
        Price)")
```




```r
plot(house_prices_working_data[c("LotArea","SalePrice")],main="LotArea vs Sale Price")
```

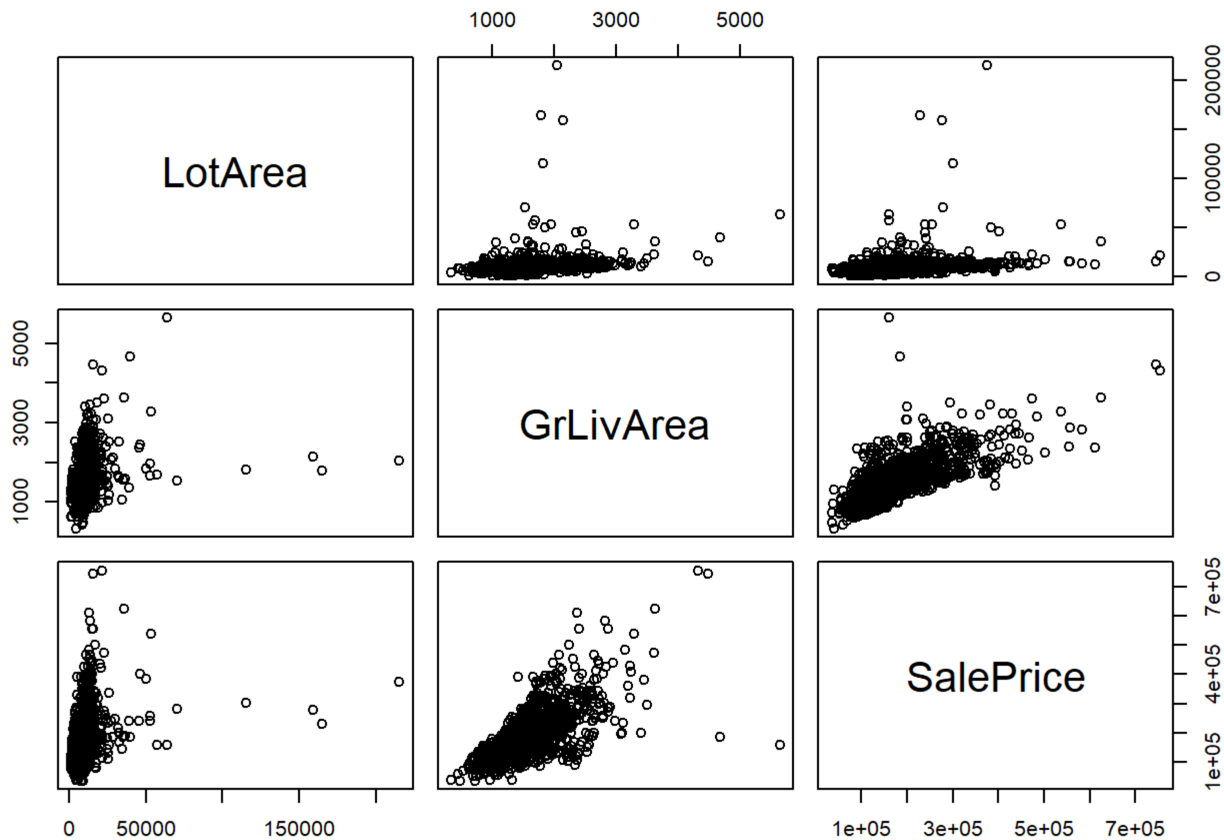



```
plot(log(house_prices_working_data[c("LotArea","SalePrice")]),main="Log(LotArea) vs Log(Sale Pri
        ce)")
```

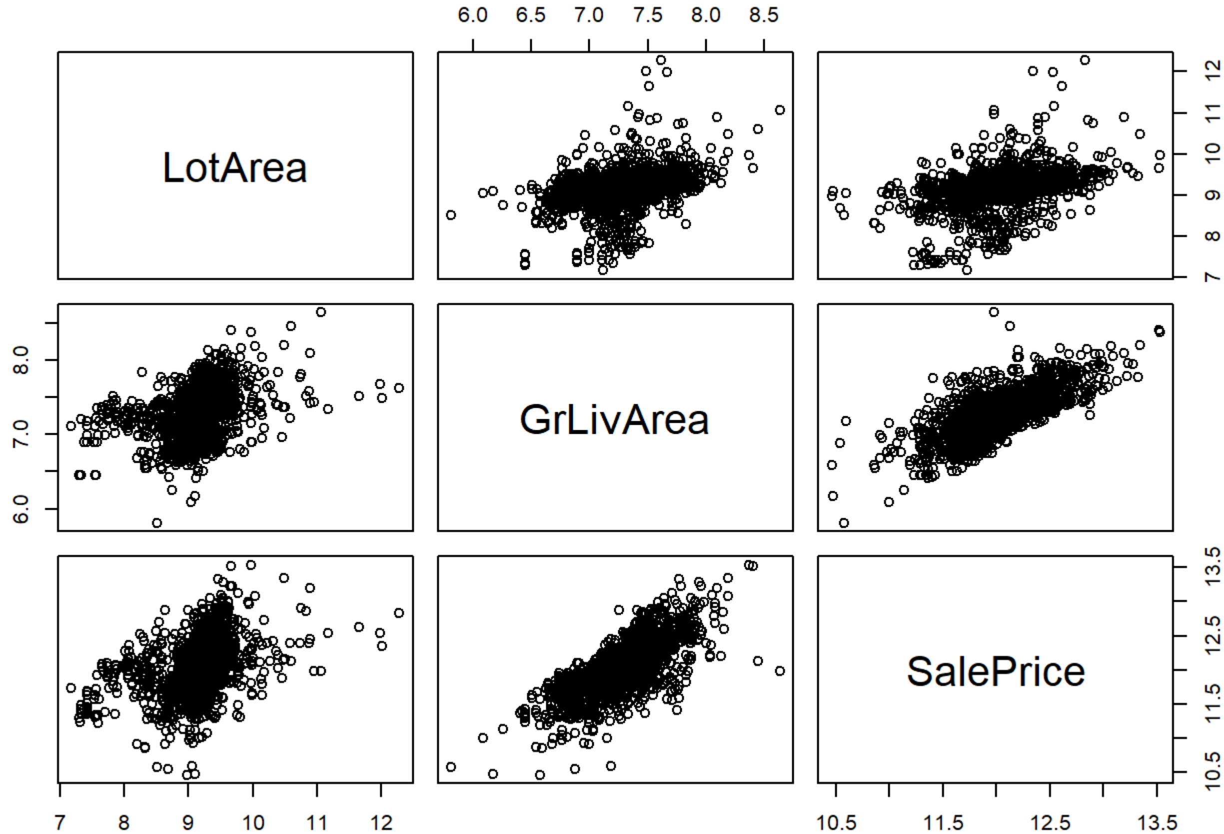


Provide a scatter-plot matrix

```
pairs(house_prices_working_data)
```



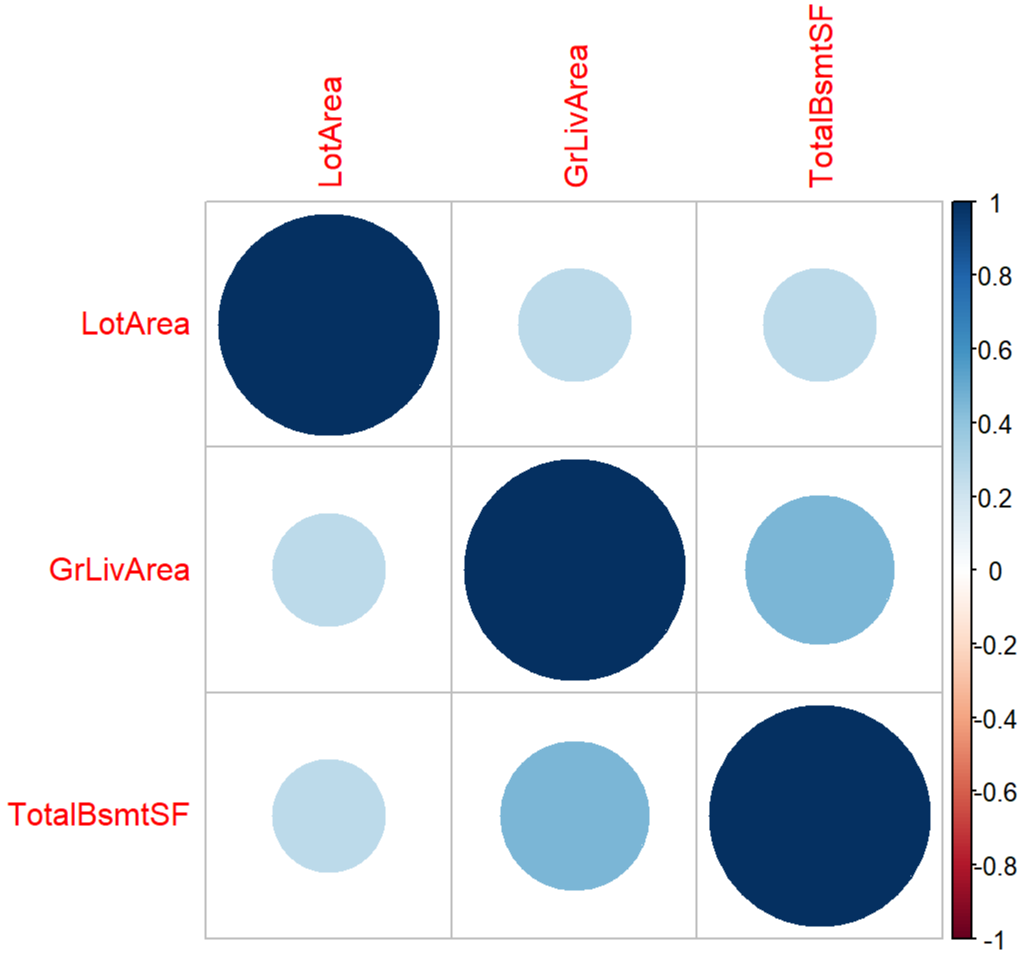

```
pairs(log(house_prices_working_data))
```



The relationship between LotArea and the other variables appears to be slightly positive, but not strong.The scatter matrix shows that there is a strong positive linear relationship between Above Grade Living Aread (“GrLivArea”) and Sale price.

Compute the Correlation matrix

```
house_prices.corrData <- house_prices.train[c('LotArea','GrLivArea','TotalBsmtSF')]
cor.mat <- cor(house_prices.corrData)
corrplot(cor.mat)
```




```
cor.test(house_prices.corrData$LotArea, house_prices.corrData$GrLivArea, method = "pearson" , co
         nf.level = 0.8)
```

```
##
##  Pearson's product-moment correlation
##
## data:  house_prices.corrData$LotArea and house_prices.corrData$GrLivArea
## t = 10.414, df = 1458, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 80 percent confidence interval:
##  0.2315997 0.2940809
## sample estimates:
##       cor
## 0.2631162
```

```
cor.test(house_prices.corrData$LotArea, house_prices.corrData$TotalBsmtSF, method = "pearson" ,
         conf.level = 0.8)
```

```
##
##  Pearson's product-moment correlation
##
## data:  house_prices.corrData$LotArea and house_prices.corrData$TotalBsmtSF
## t = 10.317, df = 1458, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 80 percent confidence interval:
##  0.2292786 0.2918400
## sample estimates:
##       cor
## 0.2608331
```

```
cor.test(house_prices.corrData$TotalBsmtSF, house_prices.corrData$GrLivArea, method = "pearson"
         , conf.level = 0.8)
```

```
##
##  Pearson's product-moment correlation
##
## data:  house_prices.corrData$TotalBsmtSF and house_prices.corrData$GrLivArea
## t = 19.503, df = 1458, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 80 percent confidence interval:
##  0.4278380 0.4810855
## sample estimates:
##       cor
## 0.4548682
```

Above we test each pariwise correlation to see if it is significantly different from zero with each case having the formulation: \[H_0 : r=0 \\ H_a : r \ne 0 \]

And we see that in all 3 cases, the correlation is significantly different from zero at 80% confidence. This makes sense given that they are all house-area-related variables and that bigger houses likely have bigger living-space, bigger basements etc. I believe that based on the family-wise error rate formula (FWER) shown below, we should be concerned here as the probabilty is nearly 50% (0.488) that we have at least 1 false conclusion:

\(FWER \leq 1 - (1-\alpha)^c\) where \(\alpha\) is the significance of the tests and \(c\) is the number of comparisons performed. In this case, we get a value of:

```
FWER <- 1-(1-0.2)^3
FWER
```

```
## [1] 0.488
```

2. Linear Algebra and Correlation

Here we invert the correlation matrix to create a precision matrix and multiply them to see whether we get the same answer independent of order:

```r
precision.mat <- solve(cor.mat)
kable(precision.mat)
```

	LotArea	GrLivArea	TotalBsmtSF
LotArea	1.1041806	-0.2011394	-0.1965150
GrLivArea	-0.2011394	1.2975230	-0.5377381
TotalBsmtSF	-0.1965150	-0.5377381	1.2958576

```r
pXc <-cor.mat %*% precision.mat
cXp <- precision.mat %*% cor.mat
pXc
```

```
##                   LotArea    GrLivArea TotalBsmtSF
## LotArea      1.000000e+00 0.000000e+00           0
## GrLivArea   -1.387779e-17 1.000000e+00           0
## TotalBsmtSF  2.775558e-17 1.110223e-16           1
```

```r
cXp
```

```
##                  LotArea     GrLivArea   TotalBsmtSF
## LotArea     1.000000e+00 -4.163336e-17 -2.775558e-17
## GrLivArea   0.000000e+00  1.000000e+00  1.110223e-16
## TotalBsmtSF 5.551115e-17  0.000000e+00  1.000000e+00
```

```r
identical(pXc,cXp)
```

```
## [1] FALSE
```

LU Decomposition

For LU Decomposition following function will run it on all three!

```
LU <- function(U){
  colnames(U) <- NULL
  rownames(U) <- NULL

  L = diag(x = 1, ncol = ncol(U), nrow = nrow(U))
  for (row in 1:dim(U)[1]){
    col = 1
    while (col< row) {
      L[row,col] <- U[row,col] / U[col,col]
      U[row,] <- -1 * U[row,col]/U[col,col] * U[col,] + U[row,]
      col = col+1
    }
  }
  return(list('L' = L, 'U' = U))
}
corMat.LU <- LU(data.matrix(cor.mat))
cXp.LU <- LU(data.matrix(cXp))
pXc.LU<- LU(data.matrix(pXc))
```

And we display the 3 outputs:

Correlation Mat U&L

```
kable(as.data.frame(corMat.LU$U))
```

V1	V2	V3
1	0.2631162	0.2608331
0	0.9307699	0.3862388
0	0.0000000	0.7716897

```
kable(as.data.frame(corMat.LU$L))
```

V1	V2	V3
1.0000000	0.000000	0
0.2631162	1.000000	0
0.2608331	0.414967	1

C X P Mat U&L

```
kable(as.data.frame(cXp.LU$U))
```

V1	V2	V3
1	0	0

V1	V2	V3
0	1	0
0	0	1

```
kable(as.data.frame(cXp.LU$L))
```

V1	V2	V3
1	0	0
0	1	0
0	0	1

P X C Mat U&L

```
kable(as.data.frame(pXc.LU$U))
```

V1	V2	V3
1	0	0
0	1	0
0	0	1

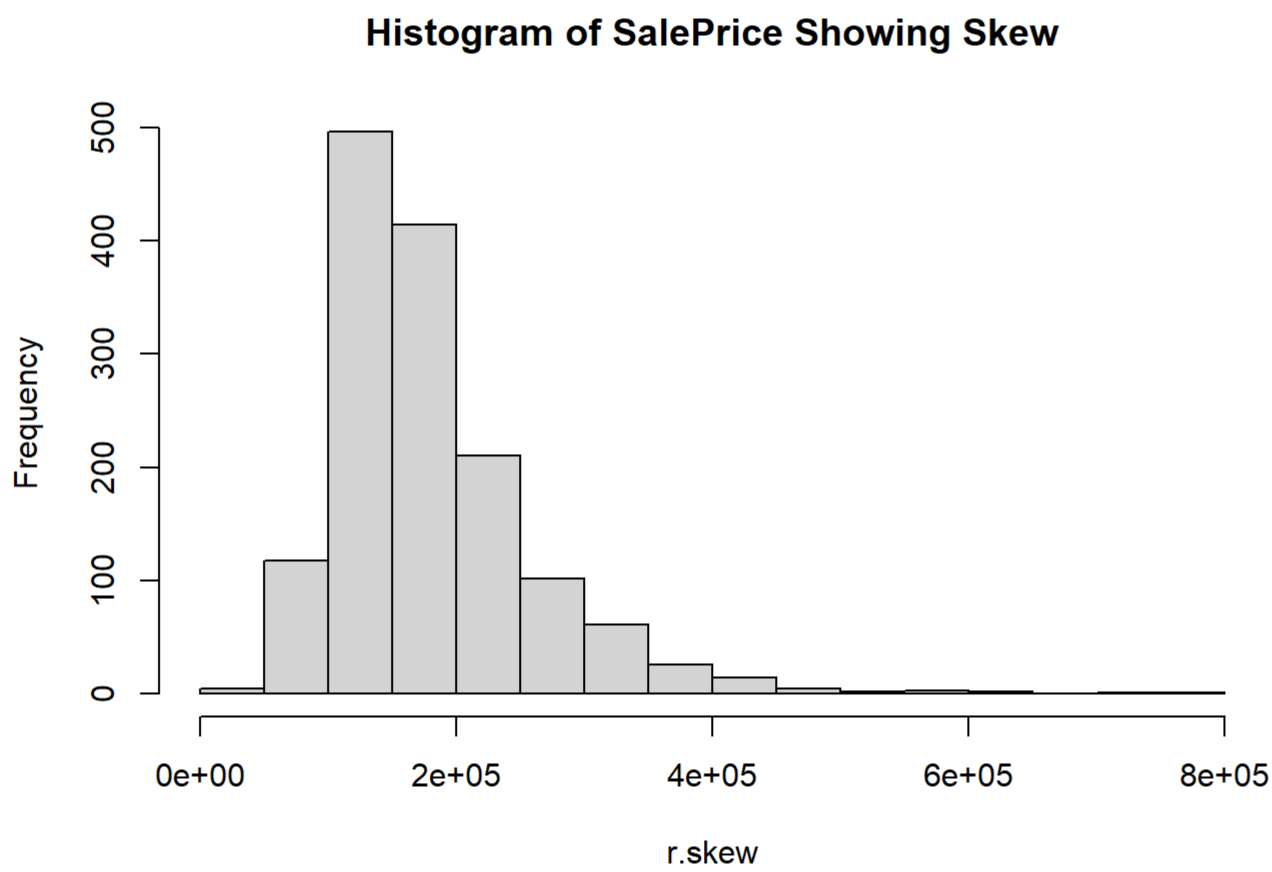
```
kable(as.data.frame(pXc.LU$L))
```

V1	V2	V3
1	0	0
0	1	0
0	0	1

3. Calculus-Based Probability & Statistics

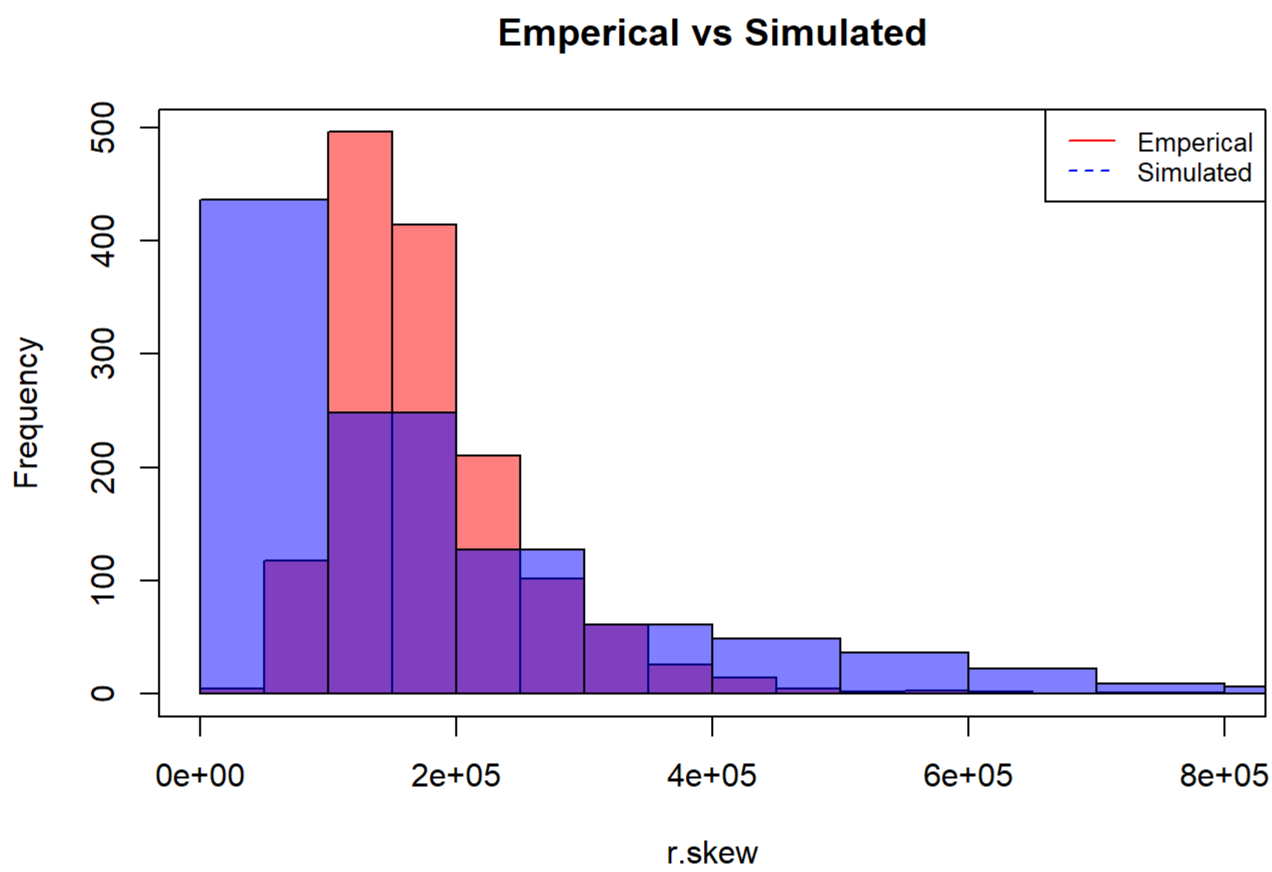
Pick a Variable with Right-Skew. Lets pick sale price, which has a reasonable skew

```
r.skew <- (house_prices.train$SalePrice)
hist(r.skew,main="Histogram of SalePrice Showing Skew")
```



Fit an Exponential Distribution & Create Histogram

```
#fit and generate 100 samples
dist.fit <- fitdistr(r.skew,densfun = "exponential")
samples <- rexp(1000,dist.fit$estimate)
#Compare histograms
hist(r.skew, col=rgb(1,0,0,0.5), breaks = 15, main="Emperical vs Simulated")
hist(samples, col=rgb(0,0,1,0.5),breaks = 15, add=T)
box()
legend("topright", legend=c("Emperical", "Simulated"),
       col=c("red", "blue"), lty=1:2, cex=0.8)
```



Compute percentiles and 95% CI

In addition to the required metrics, I have also shows summary stats as I think they are helpful here:

```
#plot
ci(r.skew,confidence = 0.95)
```

```
##    Estimate    CI lower    CI upper Std. Error
## 180921.196 176842.841 184999.551   2079.105
```

```
describe(r.skew)
```

```
##    vars    n     mean       sd median  trimmed     mad   min    max  range skew
## X1    1 1460 180921.2 79442.5 163000 170783.3 56338.8 34900 755000 720100 1.88
##    kurtosis      se
## X1      6.5 2079.11
```

```
describe(samples)
```

```
##    vars    n     mean     sd   median  trimmed      mad min     max   range
## X1    1 1000 183750.4 185929 121894.1 151078.8 124306.7 1.1 1415127 1415126
##    skew kurtosis      se
## X1 1.82     4.22 5879.59
```

Above we see a the 95% confidence interval which indicated the range in which we can be 95% confident that the mean of the population will be found. Given the nature of the distribution, I would suggest staying away from the assumption of normality in this case and using something like bootstrapping instead. If we compare the data for the simulation agains the emperical data, we see that we *do not* get an ideal fit. The empirical data has a similar mean to the simulated data, but a much higher standard deviation and median. Looking at the histograms, we can see a distinct difference in the shapes of the distributions

---

4. Modeling

First we'll run a regression on all the data and take a look at the result. We want to eliminate variables to the extent that we can. Drop anything that contains an NA and than train the model

```
df <- house_prices.train[ , apply(house_prices.train, 2, function(x) !any(is.na(x)))]
m1 <- lm(SalePrice ~ ., data = df)
```

```
summary(m1)
```

```
##
## Call:
## lm(formula = SalePrice ~ ., data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -174399  -10591      13    9618  174399
##
## Coefficients: (3 not defined because of singularities)
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -1.181e+06  1.062e+06  -1.112 0.266296
## Id                    6.688e-01  1.595e+00   0.419 0.675090
## MSSubClass           -8.564e+00  8.528e+01  -0.100 0.920028
## MSZoningFV            3.099e+04  1.219e+04   2.543 0.011112 *
## MSZoningRH            2.391e+04  1.226e+04   1.950 0.051452 .
## MSZoningRL            2.586e+04  1.044e+04   2.477 0.013396 *
## MSZoningRM            2.495e+04  9.771e+03   2.554 0.010770 *
## LotArea               7.040e-01  1.080e-01   6.521 1.00e-10 ***
## StreetPave            3.885e+04  1.226e+04   3.170 0.001562 **
## LotShapeIR2           4.532e+03  4.313e+03   1.051 0.293556
## LotShapeIR3           3.422e+03  8.796e+03   0.389 0.697299
## LotShapeReg           5.598e+02  1.661e+03   0.337 0.736120
## LandContourHLS        1.374e+04  5.276e+03   2.604 0.009323 **
## LandContourLow       -4.127e+03  6.507e+03  -0.634 0.526100
## LandContourLvl        7.273e+03  3.801e+03   1.914 0.055880 .
## UtilitiesNoSeWa      -2.945e+04  2.638e+04  -1.116 0.264602
## LotConfigCulDSac      7.631e+03  3.321e+03   2.298 0.021735 *
## LotConfigFR2         -5.860e+03  4.152e+03  -1.412 0.158314
## LotConfigFR3         -1.357e+04  1.306e+04  -1.039 0.298978
## LotConfigInside      -1.262e+03  1.803e+03  -0.700 0.483916
## LandSlopeMod          1.047e+04  4.030e+03   2.597 0.009506 **
## LandSlopeSev         -2.561e+04  1.107e+04  -2.315 0.020792 *
## NeighborhoodBlueste  -2.677e+03  1.932e+04  -0.139 0.889813
## NeighborhoodBrDale    8.298e+03  1.111e+04   0.747 0.455201
## NeighborhoodBrkSide  -1.896e+03  9.478e+03  -0.200 0.841466
## NeighborhoodClearCr  -1.285e+04  9.405e+03  -1.366 0.172043
## NeighborhoodCollgCr  -9.730e+03  7.324e+03  -1.329 0.184239
## NeighborhoodCrawfor   9.549e+03  8.656e+03   1.103 0.270190
## NeighborhoodEdwards  -1.678e+04  8.065e+03  -2.080 0.037700 *
## NeighborhoodGilbert  -1.384e+04  7.839e+03  -1.766 0.077617 .
## NeighborhoodIDOTRR   -7.324e+03  1.082e+04  -0.677 0.498597
## NeighborhoodMeadowV  -1.542e+03  1.138e+04  -0.135 0.892294
## NeighborhoodMitchel  -2.041e+04  8.266e+03  -2.470 0.013651 *
## NeighborhoodNAmes    -1.452e+04  7.890e+03  -1.840 0.066022 .
## NeighborhoodNoRidge   2.876e+04  8.389e+03   3.428 0.000628 ***
## NeighborhoodNPkVill   8.088e+03  1.431e+04   0.565 0.572103
## NeighborhoodNridgHt   2.463e+04  7.367e+03   3.343 0.000853 ***
## NeighborhoodNWAmes   -2.063e+04  8.138e+03  -2.535 0.011360 *
## NeighborhoodOldTown  -1.298e+04  9.658e+03  -1.344 0.179227
## NeighborhoodSawyer   -1.021e+04  8.217e+03  -1.243 0.214194
## NeighborhoodSawyerW  -6.179e+03  7.842e+03  -0.788 0.430921
## NeighborhoodSomerst   1.716e+01  8.966e+03   0.002 0.998473
```

```
## NeighborhoodStoneBr   3.896e+04  8.372e+03   4.653 3.61e-06 ***
## NeighborhoodSWISU    -9.568e+03  9.813e+03  -0.975 0.329727
## NeighborhoodTimber   -6.004e+03  8.359e+03  -0.718 0.472682
## NeighborhoodVeenker   3.093e+03  1.072e+04   0.289 0.772968
## Condition1Feedr       2.991e+03  5.091e+03   0.587 0.556973
## Condition1Norm        1.220e+04  4.196e+03   2.907 0.003716 **
## Condition1PosA        7.495e+03  1.028e+04   0.729 0.466139
## Condition1PosN        8.064e+03  7.601e+03   1.061 0.288931
## Condition1RRAe       -1.693e+04  9.358e+03  -1.809 0.070679 .
## Condition1RRAn        6.448e+03  7.001e+03   0.921 0.357203
## Condition1RRNe       -7.142e+03  1.835e+04  -0.389 0.697190
## Condition1RRNn        3.887e+03  1.308e+04   0.297 0.766416
## Condition2Feedr      -9.306e+03  2.302e+04  -0.404 0.686081
## Condition2Norm       -7.438e+03  1.959e+04  -0.380 0.704237
## Condition2PosA        2.017e+04  3.796e+04   0.532 0.595162
## Condition2PosN       -2.303e+05  2.757e+04  -8.352  < 2e-16 ***
## Condition2RRAe       -1.289e+05  4.673e+04  -2.757 0.005908 **
## Condition2RRAn       -1.206e+04  3.189e+04  -0.378 0.705223
## Condition2RRNn       -8.647e+03  2.708e+04  -0.319 0.749501
## BldgType2fmCon       -6.377e+03  1.285e+04  -0.496 0.619902
## BldgTypeDuplex       -1.146e+03  7.443e+03  -0.154 0.877696
## BldgTypeTwnhs        -2.552e+04  1.014e+04  -2.517 0.011966 *
## BldgTypeTwnhsE       -2.334e+04  9.188e+03  -2.540 0.011201 *
## HouseStyle1.5Unf      1.116e+04  7.907e+03   1.412 0.158309
## HouseStyle1Story      8.920e+03  4.326e+03   2.062 0.039437 *
## HouseStyle2.5Fin     -1.695e+04  1.228e+04  -1.380 0.167683
## HouseStyle2.5Unf     -1.158e+04  9.366e+03  -1.236 0.216527
## HouseStyle2Story     -6.378e+03  3.543e+03  -1.800 0.072066 .
## HouseStyleSFoyer      7.694e+03  6.183e+03   1.244 0.213590
## HouseStyleSLvl        7.340e+03  5.450e+03   1.347 0.178279
## OverallQual           8.014e+03  1.018e+03   7.874 7.29e-15 ***
## OverallCond           5.378e+03  8.692e+02   6.187 8.25e-10 ***
## YearBuilt             3.275e+02  7.360e+01   4.450 9.34e-06 ***
## YearRemodAdd          1.054e+02  5.523e+01   1.908 0.056590 .
## RoofStyleGable        1.177e+03  1.871e+04   0.063 0.949840
## RoofStyleGambrel      4.091e+03  2.047e+04   0.200 0.841666
## RoofStyleHip          2.769e+03  1.876e+04   0.148 0.882711
## RoofStyleMansard      1.712e+04  2.181e+04   0.785 0.432753
## RoofStyleShed         8.750e+04  3.547e+04   2.467 0.013772 *
## RoofMatlCompShg       6.493e+05  3.292e+04  19.724  < 2e-16 ***
## RoofMatlMembran       7.363e+05  4.765e+04  15.453  < 2e-16 ***
## RoofMatlMetal         6.963e+05  4.707e+04  14.793  < 2e-16 ***
## RoofMatlRoll          6.501e+05  4.150e+04  15.666  < 2e-16 ***
## RoofMatlTar&Grv       6.545e+05  3.784e+04  17.297  < 2e-16 ***
## RoofMatlWdShake       6.296e+05  3.666e+04  17.177  < 2e-16 ***
## RoofMatlWdShngl       7.269e+05  3.416e+04  21.275  < 2e-16 ***
## Exterior1stAsphShn   -1.097e+04  3.386e+04  -0.324 0.745942
## Exterior1stBrkComm   -1.132e+04  2.832e+04  -0.400 0.689358
## Exterior1stBrkFace    6.822e+03  1.251e+04   0.545 0.585724
## Exterior1stCBlock    -2.809e+04  2.754e+04  -1.020 0.307801
## Exterior1stCemntBd   -1.361e+04  1.917e+04  -0.710 0.477808
## Exterior1stHdBoard   -1.243e+04  1.258e+04  -0.988 0.323327
```

```
## Exterior1stImStucc   -6.780e+04  2.843e+04  -2.385 0.017224 *
## Exterior1stMetalSd   -1.990e+03  1.448e+04  -0.137 0.890733
## Exterior1stPlywood   -1.650e+04  1.245e+04  -1.325 0.185244
## Exterior1stStone     -1.381e+04  2.414e+04  -0.572 0.567551
## Exterior1stStucco    -3.653e+03  1.378e+04  -0.265 0.790949
## Exterior1stVinylSd   -1.652e+04  1.318e+04  -1.253 0.210394
## Exterior1stWd Sdng   -1.243e+04  1.204e+04  -1.032 0.302146
## Exterior1stWdShing   -4.975e+03  1.304e+04  -0.381 0.702980
## Exterior2ndAsphShn    7.085e+03  2.264e+04   0.313 0.754349
## Exterior2ndBrk Cmn    1.407e+04  2.060e+04   0.683 0.494652
## Exterior2ndBrkFace   -1.492e+03  1.319e+04  -0.113 0.909921
## Exterior2ndCBlock            NA         NA      NA       NA
## Exterior2ndCmentBd    1.237e+04  1.909e+04   0.648 0.517359
## Exterior2ndHdBoard    7.251e+03  1.234e+04   0.588 0.556776
## Exterior2ndImStucc    3.289e+04  1.431e+04   2.298 0.021710 *
## Exterior2ndMetalSd    2.294e+03  1.431e+04   0.160 0.872703
## Exterior2ndOther     -6.655e+03  2.813e+04  -0.237 0.813060
## Exterior2ndPlywood    8.272e+03  1.198e+04   0.691 0.489868
## Exterior2ndStone     -1.104e+04  1.720e+04  -0.642 0.520910
## Exterior2ndStucco     1.867e+03  1.356e+04   0.138 0.890471
## Exterior2ndVinylSd    1.596e+04  1.291e+04   1.236 0.216661
## Exterior2ndWd Sdng    9.997e+03  1.184e+04   0.845 0.398542
## Exterior2ndWd Shng    2.671e+03  1.237e+04   0.216 0.829035
## ExterQualFa          -8.475e+03  1.085e+04  -0.781 0.434775
## ExterQualGd          -3.089e+04  4.783e+03  -6.459 1.50e-10 ***
## ExterQualTA          -3.078e+04  5.353e+03  -5.750 1.12e-08 ***
## ExterCondFa          -2.694e+03  1.882e+04  -0.143 0.886199
## ExterCondGd          -7.991e+03  1.799e+04  -0.444 0.656932
## ExterCondPo           1.198e+04  3.276e+04   0.366 0.714683
## ExterCondTA          -5.285e+03  1.795e+04  -0.294 0.768536
## FoundationCBlock      1.834e+03  3.182e+03   0.576 0.564477
## FoundationPConc       4.853e+03  3.498e+03   1.387 0.165582
## FoundationSlab        8.600e+03  7.834e+03   1.098 0.272465
## FoundationStone       9.706e+02  1.087e+04   0.089 0.928841
## FoundationWood       -3.335e+04  1.510e+04  -2.209 0.027325 *
## BsmtFinSF1            3.711e+01  4.409e+00   8.416  < 2e-16 ***
## BsmtFinSF2            2.455e+01  5.787e+00   4.242 2.38e-05 ***
## BsmtUnfSF             1.498e+01  4.060e+00   3.689 0.000234 ***
## TotalBsmtSF                  NA         NA      NA       NA
## HeatingGasA          -6.580e+03  2.537e+04  -0.259 0.795359
## HeatingGasW          -1.513e+04  2.618e+04  -0.578 0.563344
## HeatingGrav          -1.433e+04  2.747e+04  -0.522 0.601921
## HeatingOthW          -4.540e+04  3.166e+04  -1.434 0.151839
## HeatingWall           9.296e+03  2.922e+04   0.318 0.750423
## HeatingQCFa          -1.496e+03  4.798e+03  -0.312 0.755158
## HeatingQCGd          -3.653e+03  2.139e+03  -1.708 0.087903 .
## HeatingQCPo           7.920e+03  2.743e+04   0.289 0.772847
## HeatingQCTA          -4.355e+03  2.116e+03  -2.058 0.039822 *
## CentralAirY          -3.452e+03  3.867e+03  -0.893 0.372264
## X1stFlrSF             5.513e+01  5.318e+00  10.367  < 2e-16 ***
## X2ndFlrSF             7.013e+01  5.243e+00  13.375  < 2e-16 ***
## LowQualFinSF          2.487e+01  1.859e+01   1.338 0.181237
```

```
## GrLivArea                     NA         NA      NA       NA
## BsmtFullBath           1.541e+03  1.964e+03   0.785 0.432883
## BsmtHalfBath           4.014e+02  3.110e+03   0.129 0.897325
## FullBath               2.522e+03  2.237e+03   1.127 0.259762
## HalfBath              -1.701e+02  2.133e+03  -0.080 0.936444
## BedroomAbvGr          -5.506e+03  1.378e+03  -3.994 6.86e-05 ***
## KitchenAbvGr          -1.580e+04  5.723e+03  -2.760 0.005862 **
## KitchenQualFa         -2.103e+04  6.345e+03  -3.315 0.000943 ***
## KitchenQualGd         -2.776e+04  3.482e+03  -7.972 3.45e-15 ***
## KitchenQualTA         -2.528e+04  3.990e+03  -6.337 3.24e-10 ***
## TotRmsAbvGrd           1.348e+03  9.738e+02   1.384 0.166551
## FunctionalMaj2         2.507e+02  1.359e+04   0.018 0.985287
## FunctionalMin1         4.244e+03  8.617e+03   0.493 0.622430
## FunctionalMin2         8.527e+03  8.558e+03   0.996 0.319263
## FunctionalMod         -7.214e+03  1.055e+04  -0.684 0.494057
## FunctionalSev         -6.024e+04  2.753e+04  -2.188 0.028829 *
## FunctionalTyp          1.963e+04  7.389e+03   2.656 0.007999 **
## Fireplaces             2.774e+03  1.370e+03   2.025 0.043076 *
## GarageCars             4.294e+03  2.216e+03   1.937 0.052937 .
## GarageArea             1.304e+01  7.617e+00   1.712 0.087196 .
## PavedDriveP           -3.287e+03  5.560e+03  -0.591 0.554567
## PavedDriveY           -2.047e+03  3.436e+03  -0.596 0.551468
## WoodDeckSF             1.367e+01  5.947e+00   2.298 0.021728 *
## OpenPorchSF            1.215e+01  1.180e+01   1.029 0.303446
## EnclosedPorch          5.447e+00  1.277e+01   0.426 0.669822
## X3SsnPorch             2.446e+01  2.311e+01   1.058 0.290148
## ScreenPorch            3.706e+01  1.257e+01   2.948 0.003259 **
## PoolArea               7.097e+01  1.831e+01   3.876 0.000112 ***
## MiscVal               -3.138e-01  1.466e+00  -0.214 0.830587
## MoSold                -6.327e+02  2.531e+02  -2.499 0.012566 *
## YrSold                -1.807e+02  5.236e+02  -0.345 0.730049
## SaleTypeCon            3.536e+04  1.835e+04   1.927 0.054204 .
## SaleTypeConLD          1.673e+04  9.998e+03   1.673 0.094580 .
## SaleTypeConLI          9.914e+03  1.190e+04   0.833 0.405035
## SaleTypeConLw         -2.240e+03  1.239e+04  -0.181 0.856546
## SaleTypeCWD            2.307e+04  1.334e+04   1.730 0.083921 .
## SaleTypeNew            3.453e+04  1.602e+04   2.155 0.031316 *
## SaleTypeOth            1.831e+04  1.501e+04   1.220 0.222807
## SaleTypeWD             5.104e+02  4.330e+03   0.118 0.906193
## SaleConditionAdjLand  8.466e+03  1.448e+04   0.585 0.558816
## SaleConditionAlloca    5.075e+03  8.757e+03   0.580 0.562322
## SaleConditionFamily  -1.410e+03  6.308e+03  -0.224 0.823156
## SaleConditionNormal   6.517e+03  2.971e+03   2.194 0.028439 *
## SaleConditionPartial -9.350e+03  1.544e+04  -0.606 0.544843
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23970 on 1273 degrees of freedom
## Multiple R-squared:  0.9206, Adjusted R-squared:  0.909
## F-statistic: 79.31 on 186 and 1273 DF,  p-value: < 2.2e-16
```

From this we can see that it appears as though there are a few variables that are more important than others. We'll collect those and work with them directly. It's also worth noting that this model appears to predict a high amount of variability in the target variable (\(r^2\) > 0.9) but given the number of variables, we can be reasonably confident that there's some overfitting going on here. It appears as though we should be able to cut about 75% of the original 80 variables

Lets transform few quantitative variables, including the target variable, using the log() function as a linear model should perform better post-transformation.

```
df.reduced <-df[c("LotArea","Street","LandContour","LotConfig","LandSlope",
                  "Neighborhood","Condition2","OverallQual","OverallCond",
                  "YearBuilt","RoofMatl","ExterQual","BsmtFinSF1","BsmtFinSF2",
                  "BsmtUnfSF","X1stFlrSF","X2ndFlrSF","KitchenAbvGr","KitchenQual",
                  "ScreenPorch","PoolArea","SalePrice")]
df.reduced$LotArea <- log(df.reduced$LotArea)
df.reduced$SalePrice <- log(df.reduced$SalePrice)
m2 <- lm(SalePrice ~ ., data = df.reduced)
```

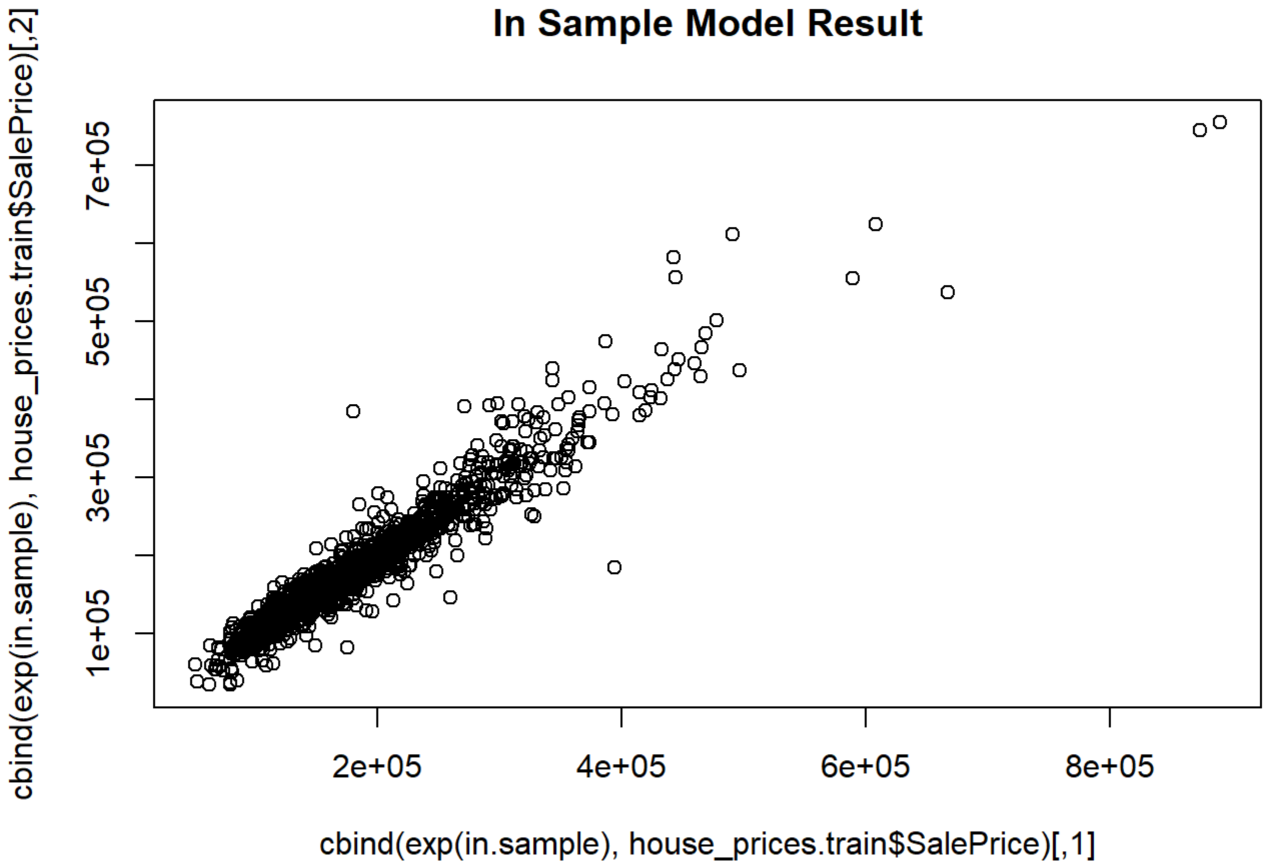
```
summary(m2)
```

```
##
## Call:
## lm(formula = SalePrice ~ ., data = df.reduced)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82329 -0.05441  0.00415  0.06361  0.75764
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          1.514e-02  5.917e-01   0.026 0.979592
## LotArea              1.170e-01  1.041e-02  11.232  < 2e-16 ***
## StreetPave           1.161e-01  5.605e-02   2.072 0.038459 *
## LandContourHLS       5.497e-02  2.531e-02   2.171 0.030076 *
## LandContourLow       2.781e-03  3.070e-02   0.091 0.927827
## LandContourLvl       2.467e-02  1.800e-02   1.370 0.170875
## LotConfigCulDSac     6.854e-03  1.565e-02   0.438 0.661432
## LotConfigFR2        -3.483e-02  1.995e-02  -1.745 0.081136 .
## LotConfigFR3        -8.824e-02  6.394e-02  -1.380 0.167807
## LotConfigInside     -8.415e-03  8.728e-03  -0.964 0.335163
## LandSlopeMod         1.944e-02  1.939e-02   1.003 0.316153
## LandSlopeSev        -6.073e-02  4.425e-02  -1.372 0.170149
## NeighborhoodBlueste -3.823e-02  9.359e-02  -0.409 0.682963
## NeighborhoodBrDale  -1.708e-01  4.558e-02  -3.748 0.000185 ***
## NeighborhoodBrkSide -7.933e-02  3.935e-02  -2.016 0.043958 *
## NeighborhoodClearCr -7.083e-02  4.371e-02  -1.620 0.105379
## NeighborhoodCollgCr -9.477e-02  3.366e-02  -2.816 0.004933 **
## NeighborhoodCrawfor  1.095e-02  3.963e-02   0.276 0.782329
## NeighborhoodEdwards -1.769e-01  3.644e-02  -4.854 1.35e-06 ***
## NeighborhoodGilbert -9.292e-02  3.625e-02  -2.563 0.010478 *
## NeighborhoodIDOTRR  -2.169e-01  4.193e-02  -5.172 2.66e-07 ***
## NeighborhoodMeadowV -2.073e-01  4.504e-02  -4.603 4.55e-06 ***
## NeighborhoodMitchel -1.555e-01  3.822e-02  -4.068 5.00e-05 ***
## NeighborhoodNAmes   -1.326e-01  3.526e-02  -3.761 0.000176 ***
## NeighborhoodNoRidge -6.513e-02  3.873e-02  -1.681 0.092898 .
## NeighborhoodNPkVill -5.624e-02  5.237e-02  -1.074 0.282969
## NeighborhoodNridgHt  2.035e-02  3.552e-02   0.573 0.566748
## NeighborhoodNWAmes  -1.577e-01  3.680e-02  -4.284 1.96e-05 ***
## NeighborhoodOldTown -1.441e-01  3.840e-02  -3.752 0.000183 ***
## NeighborhoodSawyer  -1.506e-01  3.742e-02  -4.025 6.01e-05 ***
## NeighborhoodSawyerW -1.343e-01  3.632e-02  -3.699 0.000225 ***
## NeighborhoodSomerst -5.725e-03  3.392e-02  -0.169 0.865978
## NeighborhoodStoneBr  2.249e-02  4.051e-02   0.555 0.578884
## NeighborhoodSWISU   -5.824e-02  4.443e-02  -1.311 0.190146
## NeighborhoodTimber  -8.420e-02  3.937e-02  -2.139 0.032623 *
## NeighborhoodVeenker -5.845e-02  5.083e-02  -1.150 0.250334
## Condition2Feedr      3.827e-02  1.038e-01   0.369 0.712360
## Condition2Norm       2.286e-02  9.007e-02   0.254 0.799717
## Condition2PosA       1.769e-01  1.563e-01   1.132 0.257893
## Condition2PosN      -8.904e-01  1.278e-01  -6.968 4.94e-12 ***
## Condition2RRAe      -7.202e-02  1.529e-01  -0.471 0.637699
## Condition2RRAn      -1.033e-01  1.524e-01  -0.678 0.498001
```

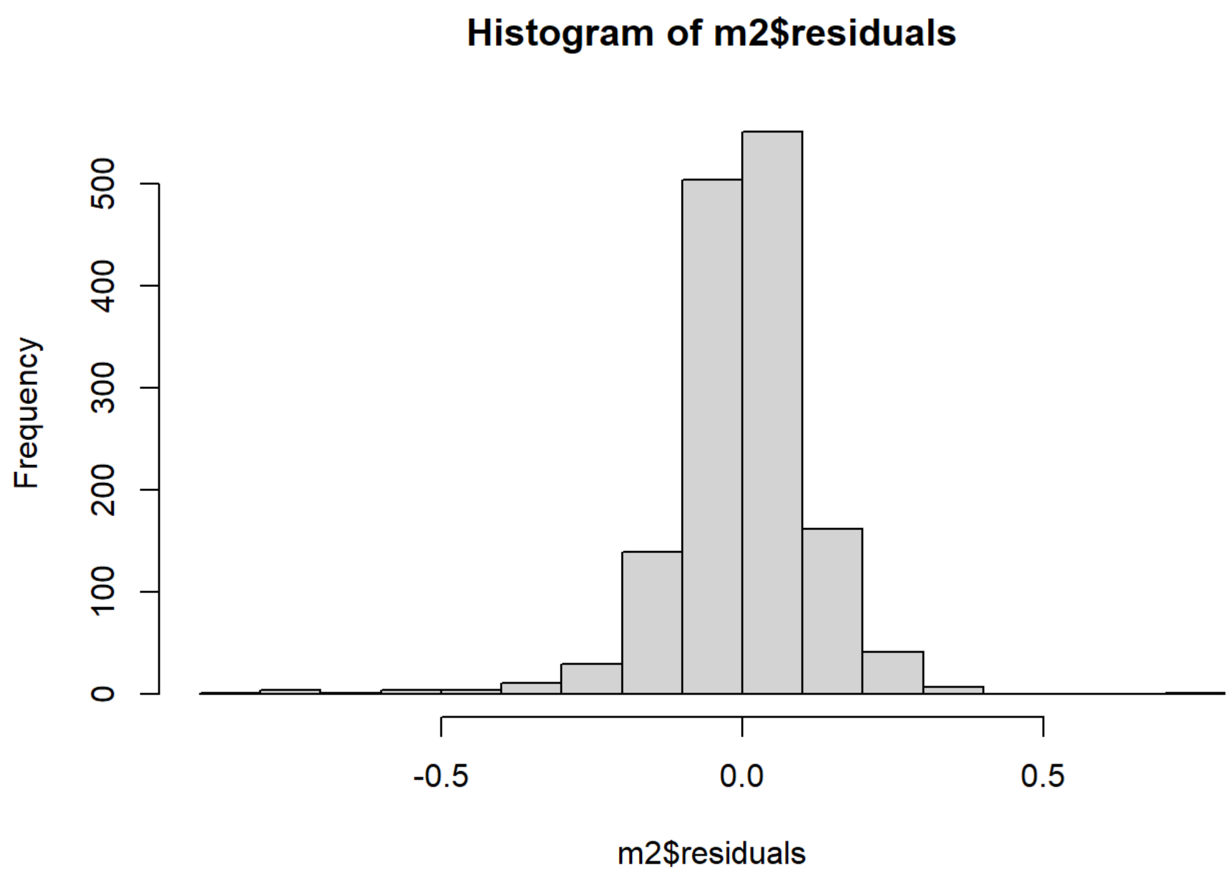
```
## Condition2RRNn      -5.669e-02  1.264e-01  -0.449 0.653781
## OverallQual          6.284e-02  4.673e-03  13.447  < 2e-16 ***
## OverallCond          5.243e-02  3.492e-03  15.015  < 2e-16 ***
## YearBuilt            3.542e-03  2.638e-04  13.425  < 2e-16 ***
## RoofMatlCompShg      2.819e+00  1.455e-01  19.381  < 2e-16 ***
## RoofMatlMembran      2.983e+00  1.984e-01  15.033  < 2e-16 ***
## RoofMatlMetal        3.000e+00  1.970e-01  15.227  < 2e-16 ***
## RoofMatlRoll         2.809e+00  1.910e-01  14.703  < 2e-16 ***
## RoofMatlTar&Grv      2.754e+00  1.495e-01  18.421  < 2e-16 ***
## RoofMatlWdShake      2.761e+00  1.566e-01  17.632  < 2e-16 ***
## RoofMatlWdShngl      2.894e+00  1.527e-01  18.953  < 2e-16 ***
## ExterQualFa         -6.737e-02  4.715e-02  -1.429 0.153250
## ExterQualGd         -1.984e-02  2.305e-02  -0.861 0.389401
## ExterQualTA         -2.466e-02  2.564e-02  -0.962 0.336431
## BsmtFinSF1           1.897e-04  1.590e-05  11.933  < 2e-16 ***
## BsmtFinSF2           1.361e-04  2.490e-05   5.466 5.46e-08 ***
## BsmtUnfSF            8.806e-05  1.544e-05   5.702 1.44e-08 ***
## X1stFlrSF            2.887e-04  1.787e-05  16.163  < 2e-16 ***
## X2ndFlrSF            2.765e-04  1.048e-05  26.384  < 2e-16 ***
## KitchenAbvGr        -6.725e-02  1.707e-02  -3.940 8.57e-05 ***
## KitchenQualFa       -1.220e-01  2.933e-02  -4.160 3.38e-05 ***
## KitchenQualGd       -5.369e-02  1.696e-02  -3.165 0.001585 **
## KitchenQualTA       -8.287e-02  1.900e-02  -4.361 1.39e-05 ***
## ScreenPorch          2.469e-04  5.993e-05   4.119 4.03e-05 ***
## PoolArea             9.656e-05  8.689e-05   1.111 0.266630
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1228 on 1393 degrees of freedom
## Multiple R-squared:  0.9098, Adjusted R-squared:  0.9055
## F-statistic: 212.9 on 66 and 1393 DF,  p-value: < 2.2e-16
```

We see a slight reduction in model performance, but it is likely worth it given the reduction in parameters. Next we'll use look at a few visualizations of the residuals.

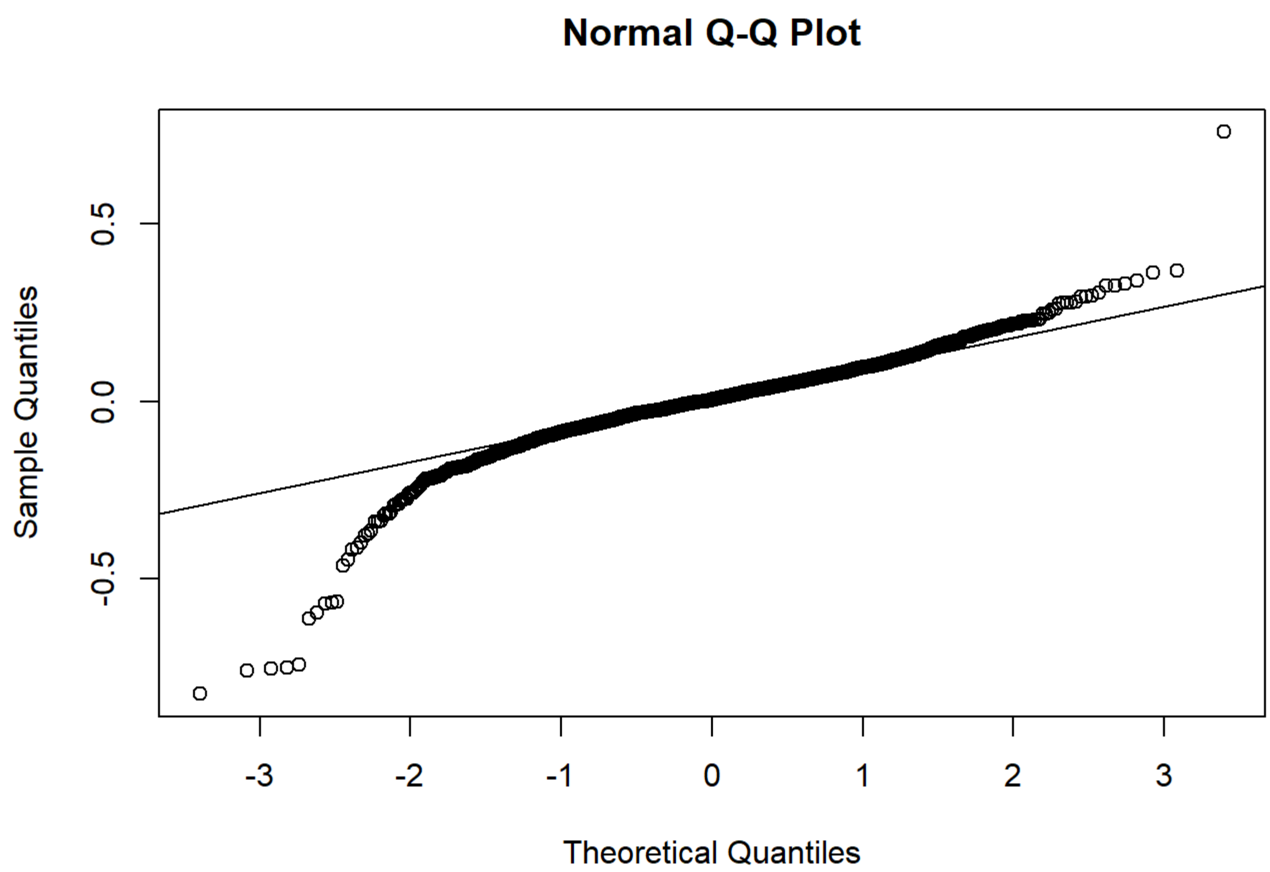
```
in.sample <- predict(m2,data=house_prices.train)
plot(cbind(exp(in.sample),house_prices.train$SalePrice), main = "In Sample Model Result")
```




```
hist(m2$residuals)
```




```r
qqnorm(m2$residuals)
qqline(m2$residuals)
```



Based on my Kaggle performance, the model is much improved, however, using log-transformed data vs. using the raw data as is. This indicates that the model likely doesn't meet the assumptions for linear regression and as such, we don't expect it to perform exceptionally well on kaggle.

Now lets run model on the test dataset and create an output file which can be loaded to kaggle.

```
df.test <-house_prices.test[c("LotArea","Street","LandContour","LotConfig","LandSlope",
                  "Neighborhood","Condition2","OverallQual","OverallCond",
                  "YearBuilt","RoofMatl","ExterQual","BsmtFinSF1","BsmtFinSF2",
                  "BsmtUnfSF","X1stFlrSF","X2ndFlrSF","KitchenAbvGr","KitchenQual",
                  "ScreenPorch","PoolArea")]
df.test$LotArea <- log(df.test$LotArea)
prediction <- exp(predict(m2, newdata = df.test) )
prediction[is.na(prediction)] <- mean(prediction, na.rm = TRUE)
prediction.df  <- as.data.frame(cbind(house_prices.test$Id,prediction))

colnames(prediction.df) <- c("Id","SalePrice")
write.csv(prediction.df, "data_605_result.csv",row.names=F)
```

The score for the model is ~0.146 which looks like model doesn't perfectly meet the assumptions of linear regression. Model results are available under https://www.kaggle.com/ramnivassingh (https://www.kaggle.com/ramnivassingh)