Project: Individual Kaggle Notebook

Ali Ladha

2024-11-08

Introduction & Project Goal

The goal of this project is to create a model for house Sale prices that has an R squared of at least .75. The project requires using many predictors to achieve this goal for house prices in the Kaggle competition. The 5 predictors that I have chosen are:

Predictors

a. LotArea - Lot size in square feet

b. Neighborhood - Physical location within Ames city limits

c. HouseStyle - Style of the dwelling

d. OverallQual- Overall Material and Finish quality

e. YearRemodAdd - Remodel date. (I chose this variable instead of Year built because it is the same as the construction date if there is no remodeling or additions)

Loading & Viewing Datasets

```
library(tidyverse)
test <- read_csv("test.csv")
train <- read_csv("train.csv")
```

Viewing Train Dataset

```
head(train)
```

```
## # A tibble: 6 × 81
##      Id MSSubClass MSZoning LotFrontage LotArea Street Alley LotShape
##   <dbl>      <dbl> <chr>          <dbl>   <dbl> <chr>  <chr> <chr>
## 1     1         60 RL                65    8450 Pave   <NA>  Reg
## 2     2         20 RL                80    9600 Pave   <NA>  Reg
## 3     3         60 RL                68   11250 Pave   <NA>  IR1
## 4     4         70 RL                60    9550 Pave   <NA>  IR1
## 5     5         60 RL                84   14260 Pave   <NA>  IR1
## 6     6         50 RL                85   14115 Pave   <NA>  IR1
## # ℹ 73 more variables: LandContour <chr>, Utilities <chr>, LotConfig <chr>,
## #   LandSlope <chr>, Neighborhood <chr>, Condition1 <chr>, Condition2 <chr>,
## #   BldgType <chr>, HouseStyle <chr>, OverallQual <dbl>, OverallCond <dbl>,
## #   YearBuilt <dbl>, YearRemodAdd <dbl>, RoofStyle <chr>, RoofMatl <chr>,
## #   Exterior1st <chr>, Exterior2nd <chr>, MasVnrType <chr>, MasVnrArea <dbl>,
## #   ExterQual <chr>, ExterCond <chr>, Foundation <chr>, BsmtQual <chr>,
## #   BsmtCond <chr>, BsmtExposure <chr>, BsmtFinType1 <chr>, BsmtFinSF1 <dbl>, …
```

Viewing Test dataset

```
head(test)
```

```
## # A tibble: 6 × 80
##      Id MSSubClass MSZoning LotFrontage LotArea Street Alley LotShape
##   <dbl>      <dbl> <chr>          <dbl>   <dbl> <chr>  <chr> <chr>
## 1  1461         20 RH                80   11622 Pave   <NA>  Reg
## 2  1462         20 RL                81   14267 Pave   <NA>  IR1
## 3  1463         60 RL                74   13830 Pave   <NA>  IR1
## 4  1464         60 RL                78    9978 Pave   <NA>  IR1
## 5  1465        120 RL                43    5005 Pave   <NA>  IR1
## 6  1466         60 RL                75   10000 Pave   <NA>  IR1
## # ℹ 72 more variables: LandContour <chr>, Utilities <chr>, LotConfig <chr>,
## #   LandSlope <chr>, Neighborhood <chr>, Condition1 <chr>, Condition2 <chr>,
## #   BldgType <chr>, HouseStyle <chr>, OverallQual <dbl>, OverallCond <dbl>,
## #   YearBuilt <dbl>, YearRemodAdd <dbl>, RoofStyle <chr>, RoofMatl <chr>,
## #   Exterior1st <chr>, Exterior2nd <chr>, MasVnrType <chr>, MasVnrArea <dbl>,
## #   ExterQual <chr>, ExterCond <chr>, Foundation <chr>, BsmtQual <chr>,
## #   BsmtCond <chr>, BsmtExposure <chr>, BsmtFinType1 <chr>, BsmtFinSF1 <dbl>, …
```

Finding Missing data - Train Data set

```
count_missings <- function(x) sum(is.na(x))

train %>%
  summarize_all(count_missings)
```

```
## # A tibble: 1 × 81
##      Id MSSubClass MSZoning LotFrontage LotArea Street Alley LotShape
##   <int>      <int>    <int>       <int>   <int>  <int> <int>    <int>
## 1     0          0        0         259       0      0  1369        0
## # ℹ 73 more variables: LandContour <int>, Utilities <int>, LotConfig <int>,
## #   LandSlope <int>, Neighborhood <int>, Condition1 <int>, Condition2 <int>,
## #   BldgType <int>, HouseStyle <int>, OverallQual <int>, OverallCond <int>,
## #   YearBuilt <int>, YearRemodAdd <int>, RoofStyle <int>, RoofMatl <int>,
## #   Exterior1st <int>, Exterior2nd <int>, MasVnrType <int>, MasVnrArea <int>,
## #   ExterQual <int>, ExterCond <int>, Foundation <int>, BsmtQual <int>,
## #   BsmtCond <int>, BsmtExposure <int>, BsmtFinType1 <int>, BsmtFinSF1 <int>, …
```

```
train %>%
  summarize_all(count_missings) %>% select(LotArea, Neighborhood, HouseStyle, OverallQua
l,
YearRemodAdd)
```

```
## # A tibble: 1 × 5
##   LotArea Neighborhood HouseStyle OverallQual YearRemodAdd
##     <int>        <int>      <int>       <int>        <int>
## 1       0            0          0           0            0
```

Based on the function, there is no missing data for the 5 predictors that were chosen using the Train data set. Therefore we do not have to drop, recode or impute the data.

Finding Missing data - Test Data set

```
test %>%
  summarize_all(count_missings)
```

```
## # A tibble: 1 × 80
##      Id MSSubClass MSZoning LotFrontage LotArea Street Alley LotShape
##   <int>      <int>    <int>       <int>   <int>  <int> <int>    <int>
## 1     0          0        4         227       0      0  1352        0
## # ℹ 72 more variables: LandContour <int>, Utilities <int>, LotConfig <int>,
## #   LandSlope <int>, Neighborhood <int>, Condition1 <int>, Condition2 <int>,
## #   BldgType <int>, HouseStyle <int>, OverallQual <int>, OverallCond <int>,
## #   YearBuilt <int>, YearRemodAdd <int>, RoofStyle <int>, RoofMatl <int>,
## #   Exterior1st <int>, Exterior2nd <int>, MasVnrType <int>, MasVnrArea <int>,
## #   ExterQual <int>, ExterCond <int>, Foundation <int>, BsmtQual <int>,
## #   BsmtCond <int>, BsmtExposure <int>, BsmtFinType1 <int>, BsmtFinSF1 <int>, …
```

```
test %>%
  summarize_all(count_missings) %>% select(LotArea, Neighborhood, HouseStyle, OverallQua
l,
YearRemodAdd)
```

```
## # A tibble: 1 × 5
##   LotArea Neighborhood HouseStyle OverallQual YearRemodAdd
##     <int>        <int>      <int>       <int>        <int>
## 1       0            0          0           0            0
```

Based on the function, there is no missing data for the 5 predictors that were chosen using the Test data set. Therefore we do not have to drop, recode or impute the data.

Outliers

Outlier for Train Dataset

```
# a) LotArea - Lot size in square feet


#Max value of Square footage
max(train$LotArea) # 215,245 Sqft
```

```
## [1] 215245
```

```
third_quartile <- (train %>% summarize(quantile(LotArea, probs = 0.75)))

outlier <- third_quartile * 1.5
outlier # 17402.25 square foot and above
```

```
##   quantile(LotArea, probs = 0.75)
## 1                        17402.25
```

```
#How many houses are above 17402 sqfeet?
(train %>% filter(LotArea >= 17402) %>% select(LotArea) %>% count()) #73 houses
```

```
## # A tibble: 1 × 1
##       n
##   <int>
## 1    73
```

```
#How many houses are below 17402 sqfeet?
(train %>% filter(LotArea <= 17402) %>% select(LotArea) %>% count()) #1387 houses
```

```
## # A tibble: 1 × 1
##       n
##   <int>
## 1  1387
```

```
(73/1387) # 0.052 * 100  = 5.2%
```

```
## [1] 0.05263158
```

```
#5.2% of houses are beyond 17402 squarefeet, so we can consider them as outliers.



#Therefore the Outlier for LotArea is 17402 sqft

#filtering out the outlier

train <- train %>% filter(LotArea <= 17402)

max(train$LotArea) #17,400 square feet.
```

```
## [1] 17400
```

```
train %>% filter(LotArea > 17402) %>% select(LotArea) %>% count()
```

```
## # A tibble: 1 × 1
##       n
##   <int>
## 1     0
```

```
#Evaluating the model with the removal of outliers

#1. Leaving variables as numerics (with removal of outliers):
lm(SalePrice ~ LotArea + Neighborhood + HouseStyle + OverallQual + YearRemodAdd, data =
train) %>% summary() #R-squared = 0.78
```

```
##
## Call:
## lm(formula = SalePrice ~ LotArea + Neighborhood + HouseStyle +
##     OverallQual + YearRemodAdd, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -122533  -19498   -1250   16078  345723
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -9.171e+05  1.234e+05  -7.434 1.87e-13 ***
## LotArea               6.193e+00  3.799e-01  16.302  < 2e-16 ***
## NeighborhoodBlueste -9.624e+03  2.650e+04  -0.363 0.716537
## NeighborhoodBrDale  -3.224e+04  1.276e+04  -2.527 0.011626 *
## NeighborhoodBrkSide -1.840e+04  1.050e+04  -1.751 0.080100 .
## NeighborhoodClearCr -2.328e+03  1.328e+04  -0.175 0.860840
## NeighborhoodCollgCr -2.010e+04  9.409e+03  -2.136 0.032870 *
## NeighborhoodCrawfor  1.420e+03  1.063e+04   0.133 0.893826
## NeighborhoodEdwards -3.080e+04  1.004e+04  -3.067 0.002202 **
## NeighborhoodGilbert -3.255e+04  1.011e+04  -3.221 0.001309 **
## NeighborhoodIDOTRR  -3.825e+04  1.120e+04  -3.416 0.000654 ***
## NeighborhoodMeadowV -4.353e+03  1.278e+04  -0.341 0.733487
## NeighborhoodMitchel -2.679e+04  1.063e+04  -2.520 0.011853 *
## NeighborhoodNAmes   -2.282e+04  9.631e+03  -2.369 0.017965 *
## NeighborhoodNoRidge  4.771e+04  1.099e+04   4.341 1.52e-05 ***
## NeighborhoodNPkVill -9.791e+03  1.472e+04  -0.665 0.505993
## NeighborhoodNridgHt  4.195e+04  9.872e+03   4.249 2.29e-05 ***
## NeighborhoodNWAmes  -2.478e+04  1.023e+04  -2.422 0.015578 *
## NeighborhoodOldTown -3.555e+04  9.898e+03  -3.592 0.000340 ***
## NeighborhoodSawyer  -2.765e+04  1.028e+04  -2.690 0.007223 **
## NeighborhoodSawyerW -2.068e+04  1.019e+04  -2.030 0.042540 *
## NeighborhoodSomerst -6.580e+03  9.589e+03  -0.686 0.492713
## NeighborhoodStoneBr  4.081e+04  1.146e+04   3.561 0.000382 ***
## NeighborhoodSWISU   -2.122e+04  1.202e+04  -1.765 0.077735 .
## NeighborhoodTimber  -1.191e+04  1.105e+04  -1.078 0.281383
## NeighborhoodVeenker -3.000e+02  1.494e+04  -0.020 0.983979
## HouseStyle1.5Unf    -2.052e+04  9.996e+03  -2.053 0.040283 *
## HouseStyle1Story    -1.698e+03  3.672e+03  -0.462 0.643944
## HouseStyle2.5Fin     4.052e+03  1.538e+04   0.264 0.792189
## HouseStyle2.5Unf    -1.277e+04  1.125e+04  -1.135 0.256537
## HouseStyle2Story     5.559e+03  3.904e+03   1.424 0.154742
## HouseStyleSFoyer    -3.746e+03  6.943e+03  -0.540 0.589623
## HouseStyleSLvl      -4.977e+03  5.769e+03  -0.863 0.388421
## OverallQual          2.766e+04  1.105e+03  25.039  < 2e-16 ***
## YearRemodAdd         4.458e+02  6.241e+01   7.143 1.49e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35180 on 1352 degrees of freedom
## Multiple R-squared:  0.7868, Adjusted R-squared:  0.7814
## F-statistic: 146.7 on 34 and 1352 DF,  p-value: < 2.2e-16
```

```
#2. Factoring variables: Neighborhood, HouseStyle, OverallQual & YearRemodAdd (with removal of outliers):

lm(SalePrice ~ LotArea + factor(Neighborhood) + factor(HouseStyle) + factor(OverallQual) + factor(YearRemodAdd), data = train) %>% summary() #R-Squared = 0.81
```

```
## 
## Call:
## lm(formula = SalePrice ~ LotArea + factor(Neighborhood) + factor(HouseStyle) + 
##     factor(OverallQual) + factor(YearRemodAdd), data = train)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -113958  -16446   -1601   14499  269354 
## 
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                  56309.692  25163.129   2.238 0.025406 *  
## LotArea                          5.038      0.370  13.616  < 2e-16 ***
## factor(Neighborhood)Blueste -19242.153  26771.154  -0.719 0.472417    
## factor(Neighborhood)BrDale  -39954.565  12993.616  -3.075 0.002150 ** 
## factor(Neighborhood)BrkSide -29937.205  10031.481  -2.984 0.002896 ** 
## factor(Neighborhood)ClearCr  -7995.070  12814.197  -0.624 0.532789    
## factor(Neighborhood)CollgCr -20166.829   8931.219  -2.258 0.024112 *  
## factor(Neighborhood)Crawfor    808.069  10082.287   0.080 0.936132    
## factor(Neighborhood)Edwards -42071.204   9579.249  -4.392 1.22e-05 ***
## factor(Neighborhood)Gilbert -30881.459   9644.344  -3.202 0.001398 ** 
## factor(Neighborhood)IDOTRR  -52737.022  10654.544  -4.950 8.42e-07 ***
## factor(Neighborhood)MeadowV -28854.755  12739.190  -2.265 0.023676 *  
## factor(Neighborhood)Mitchel -24490.810  10308.868  -2.376 0.017662 *  
## factor(Neighborhood)NAmes   -27195.396   9328.157  -2.915 0.003614 ** 
## factor(Neighborhood)NoRidge  34443.138  10752.129   3.203 0.001392 ** 
## factor(Neighborhood)NPkVill  -9105.376  14591.169  -0.624 0.532716    
## factor(Neighborhood)NridgHt  12388.013   9486.882   1.306 0.191853    
## factor(Neighborhood)NWAmes  -19114.980   9961.674  -1.919 0.055224 .  
## factor(Neighborhood)OldTown -45695.503   9467.126  -4.827 1.55e-06 ***
## factor(Neighborhood)Sawyer  -35473.878  10067.103  -3.524 0.000440 ***
## factor(Neighborhood)SawyerW -18182.465   9964.535  -1.825 0.068276 .  
## factor(Neighborhood)Somerst -16032.607   9052.317  -1.771 0.076780 .  
## factor(Neighborhood)StoneBr  20618.749  11079.349   1.861 0.062971 .  
## factor(Neighborhood)SWISU   -28302.039  11474.835  -2.466 0.013776 *  
## factor(Neighborhood)Timber  -22213.734  10537.599  -2.108 0.035220 *  
## factor(Neighborhood)Veenker    487.129  14370.412   0.034 0.972964    
## factor(HouseStyle)1.5Unf    -19998.819   9332.292  -2.143 0.032303 *  
## factor(HouseStyle)1Story     -7991.645   3590.008  -2.226 0.026182 *  
## factor(HouseStyle)2.5Fin       319.318  14423.247   0.022 0.982340    
## factor(HouseStyle)2.5Unf    -13214.219  10559.804  -1.251 0.211027    
## factor(HouseStyle)2Story      4266.683   3758.040   1.135 0.256441    
## factor(HouseStyle)SFoyer     -5946.307   6686.844  -0.889 0.374033    
## factor(HouseStyle)SLvl       -3484.941   5561.117  -0.627 0.530992    
## factor(OverallQual)2           945.643  29844.408   0.032 0.974728    
## factor(OverallQual)3         28698.971  24325.193   1.180 0.238296    
## factor(OverallQual)4         40395.654  23418.596   1.725 0.084779 .  
## factor(OverallQual)5         53031.410  23352.449   2.271 0.023317 *  
## factor(OverallQual)6         71286.333  23393.268   3.047 0.002356 ** 
## factor(OverallQual)7         96114.397  23519.072   4.087 4.65e-05 ***
## factor(OverallQual)8        141197.793  23678.055   5.963 3.19e-09 ***
## factor(OverallQual)9        210534.609  24275.061   8.673  < 2e-16 ***
## factor(OverallQual)10       275681.019  25543.995  10.792  < 2e-16 ***
## factor(YearRemodAdd)1951     16129.710  19123.722   0.843 0.399139    
## factor(YearRemodAdd)1952     10049.793  14921.759   0.673 0.500751    
## factor(YearRemodAdd)1953      3148.282  11357.092   0.277 0.781665    
## factor(YearRemodAdd)1954      7302.611   9606.549   0.760 0.447292    
## factor(YearRemodAdd)1955     14167.457  12014.024   1.179 0.238520    
## factor(YearRemodAdd)1956     11490.275  10884.292   1.056 0.291315    
## factor(YearRemodAdd)1957      9263.128  11458.098   0.808 0.418990    
## factor(YearRemodAdd)1958     -3118.425   9145.806  -0.341 0.733184    
## factor(YearRemodAdd)1959      5324.270   8795.020   0.605 0.545038    
## factor(YearRemodAdd)1960      8993.754  10909.068   0.824 0.409849    
## factor(YearRemodAdd)1961     17569.239  12178.153   1.443 0.149353    
## factor(YearRemodAdd)1962     15303.691  10214.419   1.498 0.134315    
## factor(YearRemodAdd)1963     14984.668  10556.589   1.419 0.156007    
## factor(YearRemodAdd)1964     15926.175  11139.237   1.430 0.153036    
## factor(YearRemodAdd)1965      7615.327   8695.340   0.876 0.381306    
## factor(YearRemodAdd)1966     13590.809   9524.007   1.427 0.153821    
## factor(YearRemodAdd)1967      7177.049  11058.190   0.649 0.516438    
## factor(YearRemodAdd)1968     14502.732   8807.118   1.647 0.099863 .  
## factor(YearRemodAdd)1969      6257.368   9754.331   0.641 0.521315    
## factor(YearRemodAdd)1970     12722.132   7529.766   1.690 0.091351 .  
## factor(YearRemodAdd)1971      2332.427   9103.582   0.256 0.797830    
## factor(YearRemodAdd)1972      6562.567   8320.160   0.789 0.430401    
## factor(YearRemodAdd)1973     15822.361  11126.188   1.422 0.155245    
## factor(YearRemodAdd)1974      9176.915  14118.710   0.650 0.515820    
## factor(YearRemodAdd)1975     15053.490  13003.159   1.158 0.247210    
## factor(YearRemodAdd)1976      9033.524   7677.945   1.177 0.239591    
## factor(YearRemodAdd)1977     11186.966   7806.671   1.433 0.152101    
## factor(YearRemodAdd)1978      6313.434   9232.283   0.684 0.494197    
## factor(YearRemodAdd)1979      9416.535  12474.438   0.755 0.450467    
## factor(YearRemodAdd)1980     16692.527  10989.667   1.519 0.129026    
## factor(YearRemodAdd)1981     12781.181  12153.887   1.052 0.293175    
## factor(YearRemodAdd)1982     12269.898  12894.090   0.952 0.341483    
## factor(YearRemodAdd)1983     10325.857  15184.993   0.680 0.496624    
## factor(YearRemodAdd)1984      1004.476  13110.078   0.077 0.938939    
## factor(YearRemodAdd)1985      4615.117  11713.270   0.394 0.693641    
## factor(YearRemodAdd)1986     17088.259  16966.762   1.007 0.314047    
## factor(YearRemodAdd)1987     22008.533  11391.566   1.932 0.053579 .  
## factor(YearRemodAdd)1988     23632.033  11540.255   2.048 0.040783 *  
## factor(YearRemodAdd)1989     27153.228  11111.936   2.444 0.014675 *  
## factor(YearRemodAdd)1990     27580.974   9608.927   2.870 0.004167 ** 
## factor(YearRemodAdd)1991     21438.305   9819.563   2.183 0.029200 *  
## factor(YearRemodAdd)1992      2682.551   8970.259   0.299 0.764951    
## factor(YearRemodAdd)1993     29287.758   8534.999   3.431 0.000619 ***
## factor(YearRemodAdd)1994     30021.584   8361.257   3.591 0.000342 ***
## factor(YearRemodAdd)1995     24399.215   6781.156   3.598 0.000333 ***
## factor(YearRemodAdd)1996     26235.419   6483.958   4.046 5.52e-05 ***
## factor(YearRemodAdd)1997     43507.553   7699.761   5.651 1.97e-08 ***
## factor(YearRemodAdd)1998     23649.259   6495.914   3.641 0.000283 ***
## factor(YearRemodAdd)1999     29266.557   7070.698   4.139 3.71e-05 ***
## factor(YearRemodAdd)2000     17842.874   5485.125   3.253 0.001172 ** 
## factor(YearRemodAdd)2001     27580.139   7883.311   3.499 0.000484 ***
## factor(YearRemodAdd)2002     26418.205   6054.546   4.363 1.38e-05 ***
## factor(YearRemodAdd)2003     22633.360   5858.975   3.863 0.000118 ***
## factor(YearRemodAdd)2004     21897.741   5465.018   4.007 6.51e-05 ***
## factor(YearRemodAdd)2005     20555.159   5343.283   3.847 0.000125 ***
## factor(YearRemodAdd)2006     26123.558   5032.476   5.191 2.43e-07 ***
## factor(YearRemodAdd)2007     28150.598   5337.623   5.274 1.56e-07 ***
## factor(YearRemodAdd)2008     39659.815   6677.658   5.939 3.68e-09 ***
## factor(YearRemodAdd)2009     54303.009   8156.648   6.658 4.12e-11 ***
## factor(YearRemodAdd)2010     82763.395  14128.364   5.858 5.95e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32450 on 1285 degrees of freedom
## Multiple R-squared:  0.8276, Adjusted R-squared:  0.814
## F-statistic: 61.06 on 101 and 1285 DF,  p-value: < 2.2e-16
```

```
train %>% count()
```

```
## # A tibble: 1 × 1
##       n
##   <int>
## 1  1387
```

```
#The total rows for the Train dataset is 1387 with the removal of Outliers.
#Kaggle only accepts a submission of data that has Atleast 1459 rows.
#Therefore, we will add the Outliers back to the analysis. This may result in a differen
t R squared than above.


train <- read_csv("train.csv")
```

```
## Rows: 1460 Columns: 81
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (43): MSZoning, Street, Alley, LotShape, LandContour, Utilities, LotConf...
## dbl (38): Id, MSSubClass, LotFrontage, LotArea, OverallQual, OverallCond, Ye...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

The Outliers for the Train dataset for LotArea was initially removed. However Kaggle only accepts a submission of data that has Atleast 1459 rows. Therefore, we we added the Outliers back to the analysis by restoring the data.

Outliers for Test Dataset

```
#We will filter Test by the same square footage outlier as it is reasonable


test <- test %>% filter(LotArea <= 17402)

max(test$LotArea) #17,360 square feet.
```

```
## [1] 17360
```

```
test %>% count()
```

```
## # A tibble: 1 × 1
##       n
##   <int>
## 1  1393
```

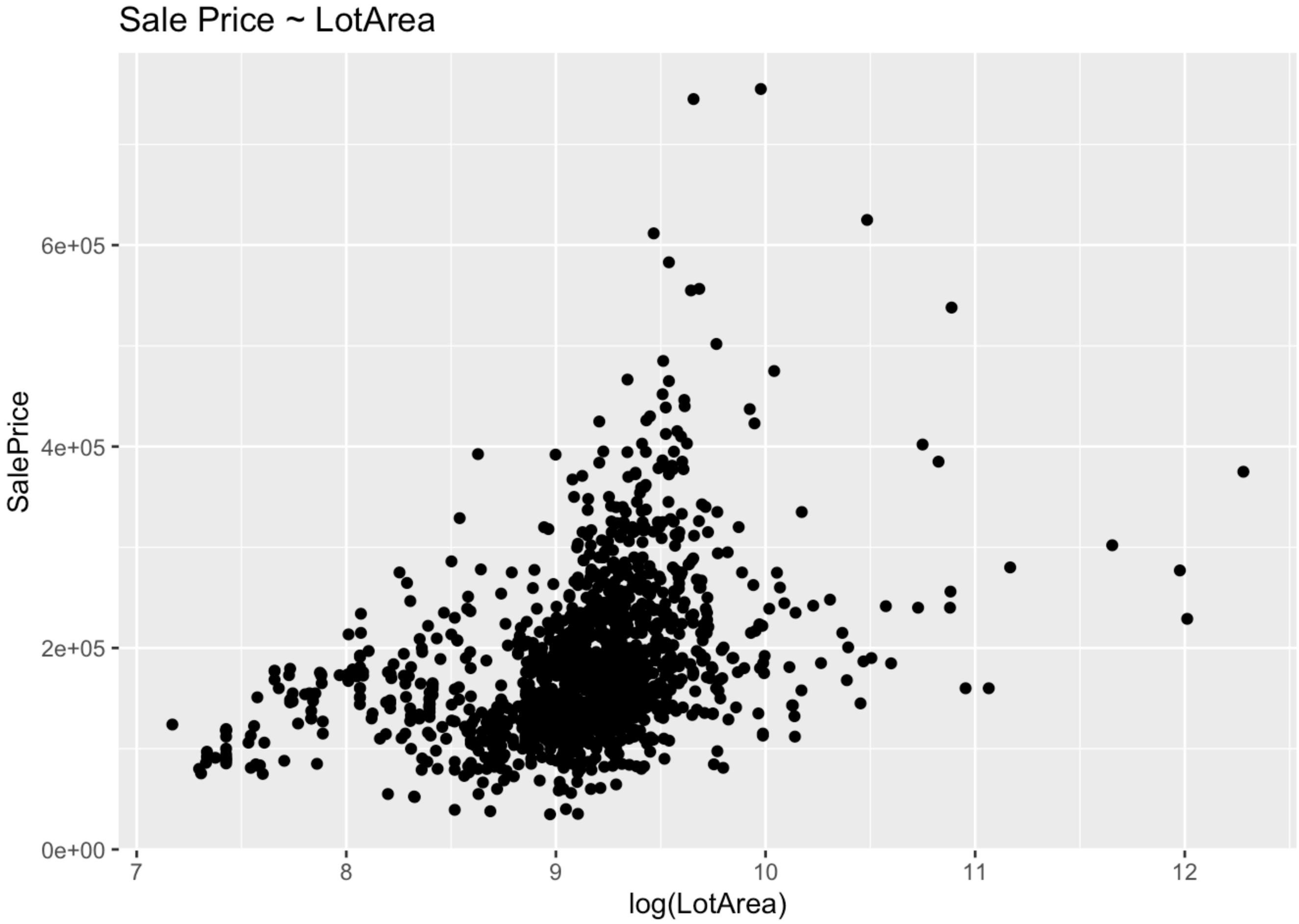
```
#The total rows for the Test dataset is 1393 with the removal of Outliers.
#Kaggle only accepts a submission of data that has Atleast 1459 rows.
#Therefore, we will add the Outliers back to the analysis.


test <- read_csv("test.csv")
```

```
## Rows: 1459 Columns: 80
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (43): MSZoning, Street, Alley, LotShape, LandContour, Utilities, LotConf...
## dbl (37): Id, MSSubClass, LotFrontage, LotArea, OverallQual, OverallCond, Ye...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

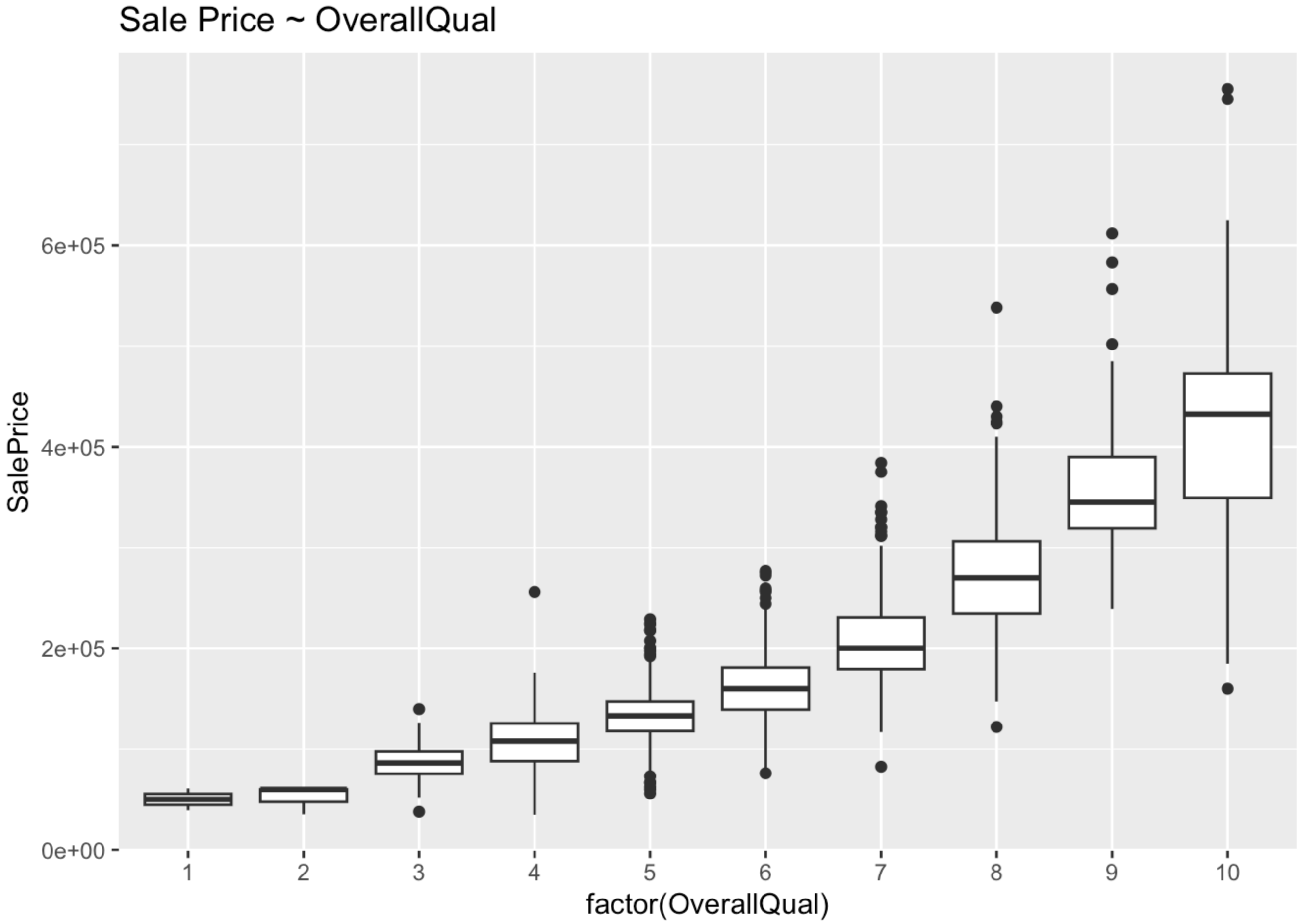
The Outliers for the Train dataset for LotArea was initially removed. However Kaggle only accepts a submission of data that has Atleast 1459 rows. Therefore, we we added the Outliers back to the analysis by restoring the data.

Data modeling

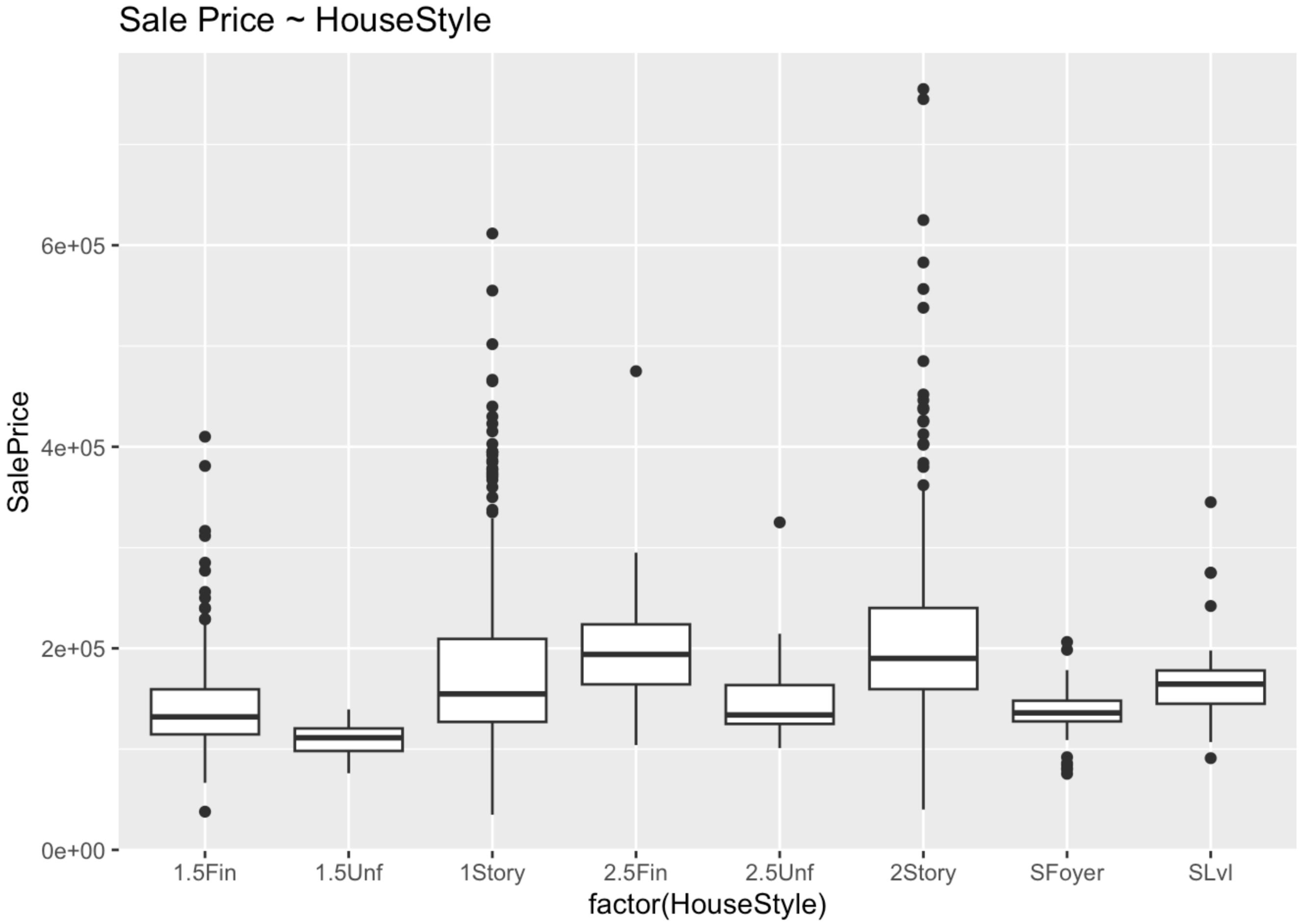
```
#Evaluating the Relationship between SalePrice and Lot Area

train %>% ggplot(aes(log(LotArea), SalePrice)) + geom_point() + labs(title = "Sale Price
~ LotArea")
```



```
#Evaluating the Relationship between SalePrice & OverallQual


train %>% ggplot(aes(factor(OverallQual), SalePrice)) + geom_boxplot() + labs(title = "S
ale Price ~ OverallQual")
```

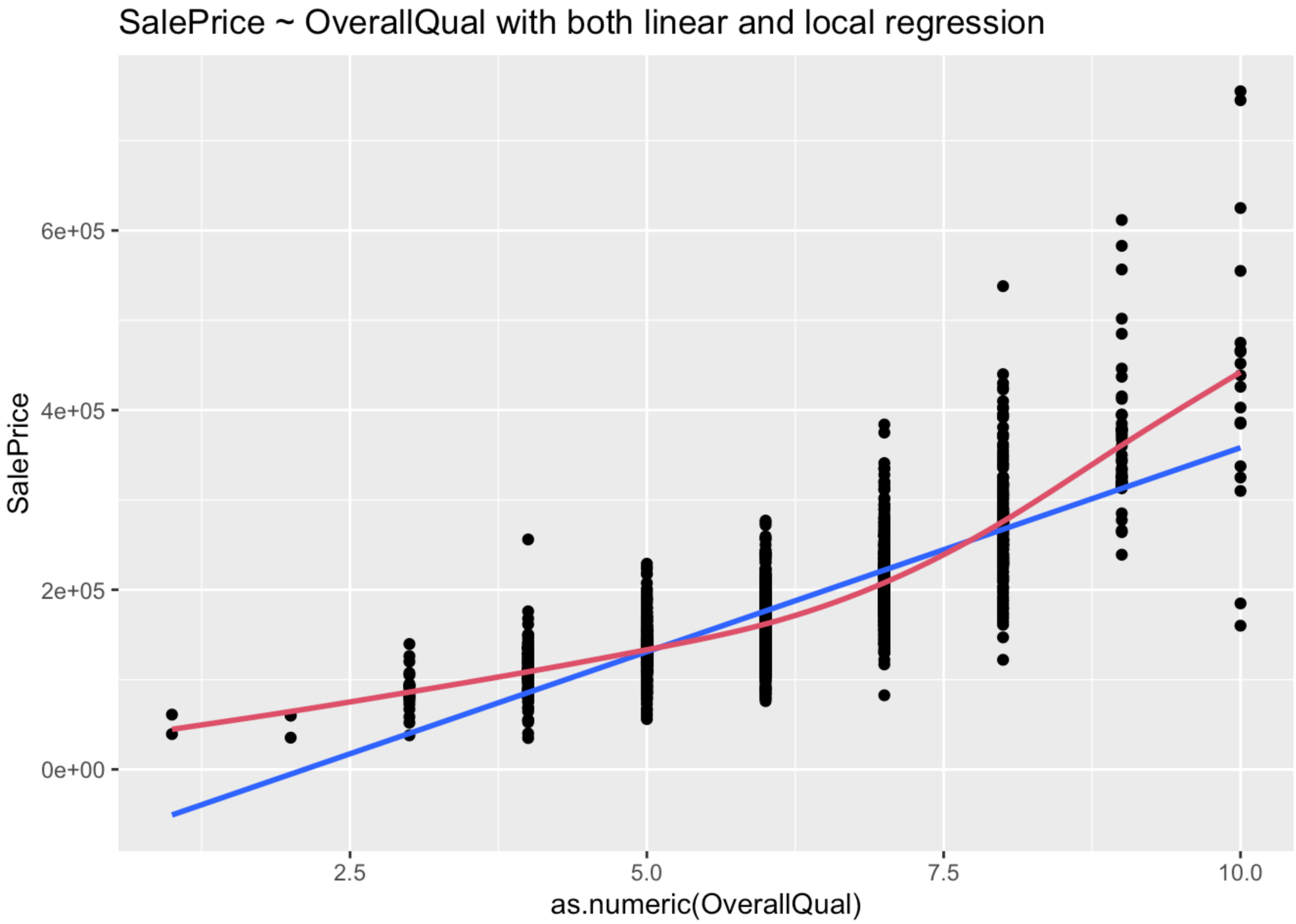


```
#Factoring predictors: Neighborhood, HouseStyle, OverallCond to improve data model

# Evaluating the Relationship between SalePrice and HouseStyle

train %>% ggplot(aes(factor(HouseStyle), SalePrice)) + geom_boxplot() + labs(title = "Sale Price ~ HouseStyle")
```



```
# Plotting a regression line with a non linear fit


ggplot(train, aes(as.numeric(OverallQual), SalePrice)) +
  geom_point() +
  geom_smooth(method = "lm", se = F) +
  geom_smooth(se = F, col = 2) + # Local regression named LOESS
  labs(title = "SalePrice ~ OverallQual with both linear and local regression")
```

```
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```



Comparing Rsquared for the models and determining to factor variables or not

```
#1. Leaving variables as numerics:
lm(SalePrice ~ LotArea + Neighborhood + HouseStyle + OverallQual + YearRemodAdd, data =
train) %>% summary() #R-squared = 0.74
```

```
##
## Call:
## lm(formula = SalePrice ~ LotArea + Neighborhood + HouseStyle +
##     OverallQual + YearRemodAdd, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -203959  -20953   -1728   16268  346228
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -7.849e+05  1.379e+05  -5.690 1.54e-08 ***
## LotArea              1.229e+00  1.187e-01  10.353  < 2e-16 ***
## NeighborhoodBlueste -1.583e+04  3.071e+04  -0.516  0.60622
## NeighborhoodBrDale  -3.703e+04  1.475e+04  -2.510  0.01218 *
## NeighborhoodBrkSide  6.316e+03  1.193e+04   0.530  0.59648
## NeighborhoodClearCr  2.983e+04  1.321e+04   2.258  0.02409 *
## NeighborhoodCollgCr  1.229e+04  1.055e+04   1.165  0.24426
## NeighborhoodCrawfor  4.025e+04  1.175e+04   3.426  0.00063 ***
## NeighborhoodEdwards  3.158e+02  1.119e+04   0.028  0.97749
## NeighborhoodGilbert  5.019e+03  1.124e+04   0.446  0.65534
## NeighborhoodIDOTRR  -1.006e+04  1.267e+04  -0.795  0.42696
## NeighborhoodMeadowV -1.674e+03  1.476e+04  -0.113  0.90975
## NeighborhoodMitchel  7.992e+03  1.177e+04   0.679  0.49738
## NeighborhoodNAmes    1.136e+04  1.070e+04   1.062  0.28840
## NeighborhoodNoRidge  1.021e+05  1.209e+04   8.445  < 2e-16 ***
## NeighborhoodNPkVill -8.384e+03  1.705e+04  -0.492  0.62290
## NeighborhoodNridgHt  7.644e+04  1.106e+04   6.913 7.14e-12 ***
## NeighborhoodNWAmes   1.669e+04  1.127e+04   1.481  0.13882
## NeighborhoodOldTown -8.170e+03  1.118e+04  -0.731  0.46519
## NeighborhoodSawyer   9.792e+03  1.140e+04   0.859  0.39040
## NeighborhoodSawyerW  1.389e+04  1.143e+04   1.215  0.22452
## NeighborhoodSomerst  1.651e+04  1.096e+04   1.507  0.13202
## NeighborhoodStoneBr  7.714e+04  1.293e+04   5.965 3.08e-09 ***
## NeighborhoodSWISU   -8.367e+02  1.366e+04  -0.061  0.95118
## NeighborhoodTimber   2.630e+04  1.221e+04   2.153  0.03147 *
## NeighborhoodVeenker  4.849e+04  1.593e+04   3.044  0.00238 **
## HouseStyle1.5Unf    -2.797e+04  1.155e+04  -2.420  0.01563 *
## HouseStyle1Story    -1.229e+03  4.102e+03  -0.300  0.76457
## HouseStyle2.5Fin     4.509e+04  1.537e+04   2.933  0.00341 **
## HouseStyle2.5Unf    -1.212e+04  1.301e+04  -0.932  0.35172
## HouseStyle2Story     5.372e+03  4.381e+03   1.226  0.22029
## HouseStyleSFoyer    -6.808e+03  7.966e+03  -0.855  0.39287
## HouseStyleSLvl      -3.096e+03  6.454e+03  -0.480  0.63151
## OverallQual          3.081e+04  1.192e+03  25.853  < 2e-16 ***
## YearRemodAdd         3.768e+02  6.984e+01   5.395 8.04e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 40790 on 1425 degrees of freedom
## Multiple R-squared:  0.7425, Adjusted R-squared:  0.7363
## F-statistic: 120.8 on 34 and 1425 DF,  p-value: < 2.2e-16
```

```
#2. Factoring variables: Neighborhood, HouseStyle, OverallQual & YearRemodAdd

lm(SalePrice ~ LotArea + factor(Neighborhood) + factor(HouseStyle) + factor(OverallQual)
+ factor(YearRemodAdd), data = train) %>% summary() #R-Squared = 0.78
```

```
## 
## Call:
## lm(formula = SalePrice ~ LotArea + factor(Neighborhood) + factor(HouseStyle) + 
##     factor(OverallQual) + factor(YearRemodAdd), data = train)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -274846  -18305   -1881   15388  263545 
## 
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                  6.670e+04  2.896e+04   2.304 0.021396 *  
## LotArea                      1.087e+00  1.138e-01   9.550  < 2e-16 ***
## factor(Neighborhood)Blueste -2.240e+04  3.088e+04  -0.726 0.468244    
## factor(Neighborhood)BrDale  -4.456e+04  1.493e+04  -2.986 0.002880 ** 
## factor(Neighborhood)BrkSide -1.045e+04  1.132e+04  -0.923 0.356406    
## factor(Neighborhood)ClearCr  2.218e+04  1.261e+04   1.758 0.078898 .  
## factor(Neighborhood)CollgCr  5.423e+03  9.943e+03   0.545 0.585527    
## factor(Neighborhood)Crawfor  3.370e+04  1.107e+04   3.044 0.002380 ** 
## factor(Neighborhood)Edwards -2.093e+04  1.065e+04  -1.966 0.049491 *  
## factor(Neighborhood)Gilbert  4.450e+02  1.060e+04   0.042 0.966539    
## factor(Neighborhood)IDOTRR  -3.256e+04  1.199e+04  -2.715 0.006702 ** 
## factor(Neighborhood)MeadowV -3.145e+04  1.461e+04  -2.152 0.031570 *  
## factor(Neighborhood)Mitchel  2.653e+02  1.133e+04   0.023 0.981327    
## factor(Neighborhood)NAmes   -1.644e+03  1.029e+04  -0.160 0.873106    
## factor(Neighborhood)NoRidge  7.813e+04  1.179e+04   6.626 4.97e-11 ***
## factor(Neighborhood)NPkVill -1.075e+04  1.675e+04  -0.642 0.521270    
## factor(Neighborhood)NridgHt  3.937e+04  1.062e+04   3.708 0.000217 ***
## factor(Neighborhood)NWAmes   1.349e+04  1.091e+04   1.237 0.216450    
## factor(Neighborhood)OldTown -2.505e+04  1.065e+04  -2.353 0.018778 *  
## factor(Neighborhood)Sawyer  -7.477e+03  1.106e+04  -0.676 0.499047    
## factor(Neighborhood)SawyerW  1.098e+04  1.106e+04   0.993 0.320749    
## factor(Neighborhood)Somerst  4.740e+03  1.029e+04   0.461 0.645013    
## factor(Neighborhood)StoneBr  5.175e+04  1.245e+04   4.155 3.45e-05 ***
## factor(Neighborhood)SWISU   -1.232e+04  1.295e+04  -0.951 0.341962    
## factor(Neighborhood)Timber   9.968e+03  1.162e+04   0.858 0.391036    
## factor(Neighborhood)Veenker  3.742e+04  1.522e+04   2.458 0.014102 *  
## factor(HouseStyle)1.5Unf    -2.578e+04  1.074e+04  -2.401 0.016483 *  
## factor(HouseStyle)1Story    -9.022e+03  4.002e+03  -2.254 0.024343 *  
## factor(HouseStyle)2.5Fin     2.956e+04  1.444e+04   2.047 0.040846 *  
## factor(HouseStyle)2.5Unf    -1.048e+04  1.215e+04  -0.863 0.388528    
## factor(HouseStyle)2Story     3.069e+03  4.201e+03   0.731 0.465144    
## factor(HouseStyle)SFoyer    -1.004e+04  7.628e+03  -1.316 0.188304    
## factor(HouseStyle)SLvl      -2.669e+03  6.229e+03  -0.428 0.668385    
## factor(OverallQual)2         4.013e+03  3.445e+04   0.117 0.907270    
## factor(OverallQual)3         2.800e+04  2.801e+04   0.999 0.317748    
## factor(OverallQual)4         4.123e+04  2.702e+04   1.526 0.127279    
## factor(OverallQual)5         5.366e+04  2.694e+04   1.992 0.046605 *  
## factor(OverallQual)6         7.144e+04  2.699e+04   2.647 0.008216 ** 
## factor(OverallQual)7         9.878e+04  2.711e+04   3.643 0.000279 ***
## factor(OverallQual)8         1.464e+05  2.728e+04   5.366 9.45e-08 ***
## factor(OverallQual)9         2.276e+05  2.788e+04   8.162 7.44e-16 ***
## factor(OverallQual)10        2.895e+05  2.878e+04  10.060  < 2e-16 ***
## factor(YearRemodAdd)1951     1.828e+04  1.925e+04   0.950 0.342464    
## factor(YearRemodAdd)1952     1.118e+04  1.720e+04   0.650 0.515673    
## factor(YearRemodAdd)1953     1.114e+04  1.243e+04   0.896 0.370204    
## factor(YearRemodAdd)1954     9.276e+03  1.070e+04   0.867 0.386194    
## factor(YearRemodAdd)1955     1.802e+04  1.310e+04   1.375 0.169304    
## factor(YearRemodAdd)1956     1.906e+04  1.252e+04   1.522 0.128164    
## factor(YearRemodAdd)1957     1.383e+04  1.319e+04   1.048 0.294618    
## factor(YearRemodAdd)1958     6.169e+02  1.049e+04   0.059 0.953128    
## factor(YearRemodAdd)1959     1.007e+04  9.818e+03   1.026 0.305050    
## factor(YearRemodAdd)1960     1.708e+04  1.161e+04   1.472 0.141257    
## factor(YearRemodAdd)1961     1.574e+04  1.401e+04   1.123 0.261574    
## factor(YearRemodAdd)1962     1.756e+04  1.099e+04   1.599 0.110106    
## factor(YearRemodAdd)1963     1.365e+04  1.124e+04   1.214 0.224776    
## factor(YearRemodAdd)1964     1.966e+04  1.224e+04   1.606 0.108505    
## factor(YearRemodAdd)1965     4.334e+03  9.743e+03   0.445 0.656510    
## factor(YearRemodAdd)1966     2.200e+04  1.062e+04   2.072 0.038452 *  
## factor(YearRemodAdd)1967     5.468e+03  1.172e+04   0.467 0.640877    
## factor(YearRemodAdd)1968     1.651e+04  1.010e+04   1.634 0.102476    
## factor(YearRemodAdd)1969     1.493e+04  1.082e+04   1.379 0.167975    
## factor(YearRemodAdd)1970     1.678e+04  8.627e+03   1.945 0.052007 .  
## factor(YearRemodAdd)1971     4.419e+03  1.045e+04   0.423 0.672355    
## factor(YearRemodAdd)1972     9.182e+03  9.556e+03   0.961 0.336827    
## factor(YearRemodAdd)1973     1.721e+04  1.280e+04   1.344 0.179078    
## factor(YearRemodAdd)1974     1.579e+04  1.511e+04   1.045 0.296157    
## factor(YearRemodAdd)1975     1.959e+04  1.271e+04   1.542 0.123344    
## factor(YearRemodAdd)1976     1.407e+04  8.577e+03   1.641 0.101120    
## factor(YearRemodAdd)1977     1.671e+04  8.749e+03   1.910 0.056385 .  
## factor(YearRemodAdd)1978     1.052e+04  1.058e+04   0.994 0.320389    
## factor(YearRemodAdd)1979     5.128e+03  1.285e+04   0.399 0.690007    
## factor(YearRemodAdd)1980     1.693e+04  1.265e+04   1.338 0.181025    
## factor(YearRemodAdd)1981     2.052e+04  1.397e+04   1.469 0.142055    
## factor(YearRemodAdd)1982     1.064e+04  1.484e+04   0.717 0.473365    
## factor(YearRemodAdd)1983     1.155e+04  1.748e+04   0.661 0.508974    
## factor(YearRemodAdd)1984    -6.788e+03  1.506e+04  -0.451 0.652212    
## factor(YearRemodAdd)1985     7.810e+02  1.345e+04   0.058 0.953715    
## factor(YearRemodAdd)1986     9.357e+02  1.743e+04   0.054 0.957183    
## factor(YearRemodAdd)1987     2.466e+04  1.254e+04   1.967 0.049341 *  
## factor(YearRemodAdd)1988     2.918e+04  1.326e+04   2.201 0.027938 *  
## factor(YearRemodAdd)1989     2.670e+04  1.212e+04   2.202 0.027831 *  
## factor(YearRemodAdd)1990     3.348e+04  1.041e+04   3.216 0.001329 ** 
## factor(YearRemodAdd)1991     2.350e+04  1.092e+04   2.153 0.031513 *  
## factor(YearRemodAdd)1992    -4.893e+02  1.004e+04  -0.049 0.961135    
## factor(YearRemodAdd)1993     2.874e+04  9.585e+03   2.999 0.002759 ** 
## factor(YearRemodAdd)1994     2.502e+04  9.038e+03   2.769 0.005702 ** 
## factor(YearRemodAdd)1995     3.376e+04  7.701e+03   4.384 1.25e-05 ***
## factor(YearRemodAdd)1996     2.710e+04  7.367e+03   3.678 0.000244 ***
## factor(YearRemodAdd)1997     4.145e+04  8.497e+03   4.878 1.20e-06 ***
## factor(YearRemodAdd)1998     2.295e+04  7.432e+03   3.088 0.002059 ** 
## factor(YearRemodAdd)1999     2.341e+04  7.996e+03   2.928 0.003467 ** 
## factor(YearRemodAdd)2000     1.324e+04  6.268e+03   2.113 0.034784 *  
## factor(YearRemodAdd)2001     2.563e+04  9.059e+03   2.830 0.004729 ** 
## factor(YearRemodAdd)2002     3.024e+04  6.764e+03   4.470 8.46e-06 ***
## factor(YearRemodAdd)2003     2.477e+04  6.515e+03   3.802 0.000150 ***
## factor(YearRemodAdd)2004     1.849e+04  6.185e+03   2.989 0.002853 **
## factor(YearRemodAdd)2005     1.913e+04  6.014e+03   3.180 0.001504 **
## factor(YearRemodAdd)2006     3.024e+04  5.626e+03   5.375 9.02e-08 ***
## factor(YearRemodAdd)2007     2.713e+04  5.987e+03   4.531 6.38e-06 ***
## factor(YearRemodAdd)2008     2.710e+04  7.545e+03   3.592 0.000340 ***
## factor(YearRemodAdd)2009     5.681e+04  9.192e+03   6.180 8.45e-10 ***
## factor(YearRemodAdd)2010     7.412e+04  1.625e+04   4.562 5.52e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37470 on 1358 degrees of freedom
## Multiple R-squared:  0.793,  Adjusted R-squared:  0.7776
## F-statistic:  51.5 on 101 and 1358 DF,  p-value: < 2.2e-16
```

The R squared was originally 0.74 without factoring. Upon factoring, the R squared improved to 0.78. Factoring Neighborhood, HouseStyle, OverallQual & YearRemodAdd predictors improved R squared by 0.04 compared to evaluating the predictors as numerics.

Overall The R-Squared with the predictors factored is .78 surpassing the benchmark goal of .75 The RMSE also went down upon factoring from: 40,790 (originally) to 37470 (upon factoring) which is an improvement.

Since the factored model R squared is higher than both the benchmark and the non factored models R squared, the 4 predictors will be factored.

Factoring for the variables in the Train Dataset

```
train <- train %>%
  mutate(Neighborhood = factor(Neighborhood),
         HouseStyle = factor(HouseStyle),
         OverallQual = factor(OverallQual),
         YearRemodAdd = factor(YearRemodAdd))
```

Cross Validation to determine overfitting

Creating an index of 70% of the rows randomly

```
set.seed(123)
index <- sample(x = 1:nrow(train),
                size = nrow(train) * 0.7,
                replace = F)



head(index)
```

```
## [1] 415 463 179 526 195 938
```

Creating a random 70/30 split of the data via index

```
train_fold <- train[index, ] #this is the 70% split
validation_fold <- train[-index, ] # this is the 30% split for testing
```

Out of Sample Performance

Fitting the model on the train fold to evaluate it on the validation fold

```
# Fitting the Example Model
model <- lm(SalePrice ~ LotArea + Neighborhood + HouseStyle + OverallQual + YearRemodAd
d, data = train_fold)



# Getting predictions on the validation fold

predictions <- predict(model, newdata = validation_fold)

rmse <- function(observed, predicted) sqrt(mean((observed - predicted)^2))

r_squared <- function(observed, predicted) {
  TSS <- sum((observed - mean(observed))^2)
  RSS <- sum((observed - predicted)^2)
  1- RSS/TSS
}




#Estimating the out of sample RMSE

rmse(validation_fold$SalePrice, predictions)
```

```
## [1] 34647.15
```

```
#Estimating the Rsquared

r_squared(validation_fold$SalePrice, predictions)
```

```
## [1] 0.8084348
```

After splitting the data, The RMSE is 34647.15 The Out of sample R squared of the model is .81 beating the Out of sample R squared benchmark of .75.

In sample Performance

Fitting the model with the train set in its entirely to determine its In Sample Performance

```
submission_model <- lm(SalePrice ~ LotArea + Neighborhood + HouseStyle + OverallQual + YearRemodAdd, data = train)

summary(submission_model)
```

```
##
## Call:
## lm(formula = SalePrice ~ LotArea + Neighborhood + HouseStyle +
##     OverallQual + YearRemodAdd, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -274846  -18305   -1881   15388  263545
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          6.670e+04  2.896e+04   2.304 0.021396 *
## LotArea              1.087e+00  1.138e-01   9.550  < 2e-16 ***
## NeighborhoodBlueste -2.240e+04  3.088e+04  -0.726 0.468244
## NeighborhoodBrDale  -4.456e+04  1.493e+04  -2.986 0.002880 **
## NeighborhoodBrkSide -1.045e+04  1.132e+04  -0.923 0.356406
## NeighborhoodClearCr  2.218e+04  1.261e+04   1.758 0.078898 .
## NeighborhoodCollgCr  5.423e+03  9.943e+03   0.545 0.585527
## NeighborhoodCrawfor  3.370e+04  1.107e+04   3.044 0.002380 **
## NeighborhoodEdwards -2.093e+04  1.065e+04  -1.966 0.049491 *
## NeighborhoodGilbert  4.450e+02  1.060e+04   0.042 0.966539
## NeighborhoodIDOTRR  -3.256e+04  1.199e+04  -2.715 0.006702 **
## NeighborhoodMeadowV -3.145e+04  1.461e+04  -2.152 0.031570 *
## NeighborhoodMitchel  2.653e+02  1.133e+04   0.023 0.981327
## NeighborhoodNAmes   -1.644e+03  1.029e+04  -0.160 0.873106
## NeighborhoodNoRidge  7.813e+04  1.179e+04   6.626 4.97e-11 ***
## NeighborhoodNPkVill -1.075e+04  1.675e+04  -0.642 0.521270
## NeighborhoodNridgHt  3.937e+04  1.062e+04   3.708 0.000217 ***
## NeighborhoodNWAmes   1.349e+04  1.091e+04   1.237 0.216450
## NeighborhoodOldTown -2.505e+04  1.065e+04  -2.353 0.018778 *
## NeighborhoodSawyer  -7.477e+03  1.106e+04  -0.676 0.499047
## NeighborhoodSawyerW  1.098e+04  1.106e+04   0.993 0.320749
## NeighborhoodSomerst  4.740e+03  1.029e+04   0.461 0.645013
## NeighborhoodStoneBr  5.175e+04  1.245e+04   4.155 3.45e-05 ***
## NeighborhoodSWISU   -1.232e+04  1.295e+04  -0.951 0.341962
## NeighborhoodTimber   9.968e+03  1.162e+04   0.858 0.391036
## NeighborhoodVeenker  3.742e+04  1.522e+04   2.458 0.014102 *
## HouseStyle1.5Unf    -2.578e+04  1.074e+04  -2.401 0.016483 *
## HouseStyle1Story    -9.022e+03  4.002e+03  -2.254 0.024343 *
## HouseStyle2.5Fin     2.956e+04  1.444e+04   2.047 0.040846 *
## HouseStyle2.5Unf    -1.048e+04  1.215e+04  -0.863 0.388528
## HouseStyle2Story     3.069e+03  4.201e+03   0.731 0.465144
## HouseStyleSFoyer    -1.004e+04  7.628e+03  -1.316 0.188304
## HouseStyleSLvl      -2.669e+03  6.229e+03  -0.428 0.668385
## OverallQual2         4.013e+03  3.445e+04   0.117 0.907270
## OverallQual3         2.800e+04  2.801e+04   0.999 0.317748
## OverallQual4         4.123e+04  2.702e+04   1.526 0.127279
## OverallQual5         5.366e+04  2.694e+04   1.992 0.046605 *
## OverallQual6         7.144e+04  2.699e+04   2.647 0.008216 **
## OverallQual7         9.878e+04  2.711e+04   3.643 0.000279 ***
## OverallQual8         1.464e+05  2.728e+04   5.366 9.45e-08 ***
## OverallQual9         2.276e+05  2.788e+04   8.162 7.44e-16 ***
## OverallQual10       2.895e+05  2.878e+04  10.060  < 2e-16 ***
## YearRemodAdd1951     1.828e+04  1.925e+04   0.950 0.342464
## YearRemodAdd1952     1.118e+04  1.720e+04   0.650 0.515673
## YearRemodAdd1953     1.114e+04  1.243e+04   0.896 0.370204
## YearRemodAdd1954     9.276e+03  1.070e+04   0.867 0.386194
## YearRemodAdd1955     1.802e+04  1.310e+04   1.375 0.169304
## YearRemodAdd1956     1.906e+04  1.252e+04   1.522 0.128164
## YearRemodAdd1957     1.383e+04  1.319e+04   1.048 0.294618
## YearRemodAdd1958     6.169e+02  1.049e+04   0.059 0.953128
## YearRemodAdd1959     1.007e+04  9.818e+03   1.026 0.305050
## YearRemodAdd1960     1.708e+04  1.161e+04   1.472 0.141257
## YearRemodAdd1961     1.574e+04  1.401e+04   1.123 0.261574
## YearRemodAdd1962     1.756e+04  1.099e+04   1.599 0.110106
## YearRemodAdd1963     1.365e+04  1.124e+04   1.214 0.224776
## YearRemodAdd1964     1.966e+04  1.224e+04   1.606 0.108505
## YearRemodAdd1965     4.334e+03  9.743e+03   0.445 0.656510
## YearRemodAdd1966     2.200e+04  1.062e+04   2.072 0.038452 *
## YearRemodAdd1967     5.468e+03  1.172e+04   0.467 0.640877
## YearRemodAdd1968     1.651e+04  1.010e+04   1.634 0.102476
## YearRemodAdd1969     1.493e+04  1.082e+04   1.379 0.167975
## YearRemodAdd1970     1.678e+04  8.627e+03   1.945 0.052007 .
## YearRemodAdd1971     4.419e+03  1.045e+04   0.423 0.672355
## YearRemodAdd1972     9.182e+03  9.556e+03   0.961 0.336827
## YearRemodAdd1973     1.721e+04  1.280e+04   1.344 0.179078
## YearRemodAdd1974     1.579e+04  1.511e+04   1.045 0.296157
## YearRemodAdd1975     1.959e+04  1.271e+04   1.542 0.123344
## YearRemodAdd1976     1.407e+04  8.577e+03   1.641 0.101120
## YearRemodAdd1977     1.671e+04  8.749e+03   1.910 0.056385 .
## YearRemodAdd1978     1.052e+04  1.058e+04   0.994 0.320389
## YearRemodAdd1979     5.128e+03  1.285e+04   0.399 0.690007
## YearRemodAdd1980     1.693e+04  1.265e+04   1.338 0.181025
## YearRemodAdd1981     2.052e+04  1.397e+04   1.469 0.142055
## YearRemodAdd1982     1.064e+04  1.484e+04   0.717 0.473365
## YearRemodAdd1983     1.155e+04  1.748e+04   0.661 0.508974
## YearRemodAdd1984    -6.788e+03  1.506e+04  -0.451 0.652212
## YearRemodAdd1985     7.810e+02  1.345e+04   0.058 0.953715
## YearRemodAdd1986     9.357e+02  1.743e+04   0.054 0.957183
## YearRemodAdd1987     2.466e+04  1.254e+04   1.967 0.049341 *
## YearRemodAdd1988     2.918e+04  1.326e+04   2.201 0.027938 *
## YearRemodAdd1989     2.670e+04  1.212e+04   2.202 0.027831 *
## YearRemodAdd1990     3.348e+04  1.041e+04   3.216 0.001329 **
## YearRemodAdd1991     2.350e+04  1.092e+04   2.153 0.031513 *
## YearRemodAdd1992    -4.893e+02  1.004e+04  -0.049 0.961135
## YearRemodAdd1993     2.874e+04  9.585e+03   2.999 0.002759 **
## YearRemodAdd1994     2.502e+04  9.038e+03   2.769 0.005702 **
## YearRemodAdd1995     3.376e+04  7.701e+03   4.384 1.25e-05 ***
## YearRemodAdd1996     2.710e+04  7.367e+03   3.678 0.000244 ***
## YearRemodAdd1997     4.145e+04  8.497e+03   4.878 1.20e-06 ***
## YearRemodAdd1998     2.295e+04  7.432e+03   3.088 0.002059 **
## YearRemodAdd1999     2.341e+04  7.996e+03   2.928 0.003467 **
## YearRemodAdd2000     1.324e+04  6.268e+03   2.113 0.034784 *
## YearRemodAdd2001     2.563e+04  9.059e+03   2.830 0.004729 **
## YearRemodAdd2002     3.024e+04  6.764e+03   4.470 8.46e-06 ***
## YearRemodAdd2003     2.477e+04  6.515e+03   3.802 0.000150 ***
## YearRemodAdd2004     1.849e+04  6.185e+03   2.989 0.002853 **
## YearRemodAdd2005     1.913e+04  6.014e+03   3.180 0.001504 **
## YearRemodAdd2006     3.024e+04  5.626e+03   5.375 9.02e-08 ***
## YearRemodAdd2007     2.713e+04  5.987e+03   4.531 6.38e-06 ***
## YearRemodAdd2008     2.710e+04  7.545e+03   3.592 0.000340 ***
## YearRemodAdd2009     5.681e+04  9.192e+03   6.180 8.45e-10 ***
## YearRemodAdd2010     7.412e+04  1.625e+04   4.562 5.52e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37470 on 1358 degrees of freedom
## Multiple R-squared:  0.793,  Adjusted R-squared:  0.7776
## F-statistic:  51.5 on 101 and 1358 DF,  p-value: < 2.2e-16
```

The In sample R-squared is: .78. The In sample RMSE is: 37470

Submission Predictions

Factoring and checking for NA in the test dataset

```
#Factoring chosen predictors on the test data set

test <- test %>%
  mutate(Neighborhood = factor(Neighborhood),
         HouseStyle = factor(HouseStyle),
         OverallQual = factor(OverallQual),
         YearRemodAdd = factor(YearRemodAdd))



#checktest#rechecking for NA's in the test set

test %>%
  summarize_all(count_missings)
```

```
## # A tibble: 1 × 80
##      Id MSSubClass MSZoning LotFrontage LotArea Street Alley LotShape
##   <int>      <int>    <int>       <int>   <int>  <int> <int>    <int>
## 1     0          0        4         227       0      0  1352        0
## # ℹ 72 more variables: LandContour <int>, Utilities <int>, LotConfig <int>,
## #   LandSlope <int>, Neighborhood <int>, Condition1 <int>, Condition2 <int>,
## #   BldgType <int>, HouseStyle <int>, OverallQual <int>, OverallCond <int>,
## #   YearBuilt <int>, YearRemodAdd <int>, RoofStyle <int>, RoofMatl <int>,
## #   Exterior1st <int>, Exterior2nd <int>, MasVnrType <int>, MasVnrArea <int>,
## #   ExterQual <int>, ExterCond <int>, Foundation <int>, BsmtQual <int>,
## #   BsmtCond <int>, BsmtExposure <int>, BsmtFinType1 <int>, BsmtFinSF1 <int>, …
```

```
test %>%
  summarize_all(count_missings) %>% select(LotArea, Neighborhood, HouseStyle, OverallQua
l,
YearRemodAdd)
```

```
## # A tibble: 1 × 5
##   LotArea Neighborhood HouseStyle OverallQual YearRemodAdd
##     <int>        <int>      <int>       <int>        <int>
## 1       0            0          0           0            0
```

There are no NAs for the 4 chosen predictors in the test dataset

Using the model to predict the missing SalePrice in the test set

```
submission_predictions <- predict(submission_model, newdata = test)



head(submission_predictions)
```

```
##        1        2        3        4        5        6 
## 138063.0 143592.0 161852.9 175440.5 260786.3 177542.9
```

Formatting submission file

```
submission <- test %>%
  select(Id) |>
  mutate(SalePrice = submission_predictions)

# Checking data for submission
head(submission)
```

```
## # A tibble: 6 × 2
##      Id SalePrice
##   <dbl>     <dbl>
## 1  1461   138063.
## 2  1462   143592.
## 3  1463   161853.
## 4  1464   175441.
## 5  1465   260786.
## 6  1466   177543.
```

```
#Writing to CSV
write.csv(submission, "kaggle_submission.csv", row.names = F)
```

Kaggle Score: 0.19413