Stats 101A - Final Project Report

Predicting House Prices in Ames, Iowa

Hao Ma

Winter 2019

Abstract

This is a report of the regression model study on Kaggle for predicting house prices in Ames, Iowa. The study consists of analyzing the given data set, selecting predictors, creating new factor variables, choosing appropriate transformations and interactions, building the final regression model, and checking the validity of the model. The user name on Kaggle is 'Hao Ma Lec 1' with public ranking #109 and a R^2 of 0.90562; private ranking #94 and a R^2 of 0.91281. There are 13 predictors in the final model with a R^2 of 0.907.

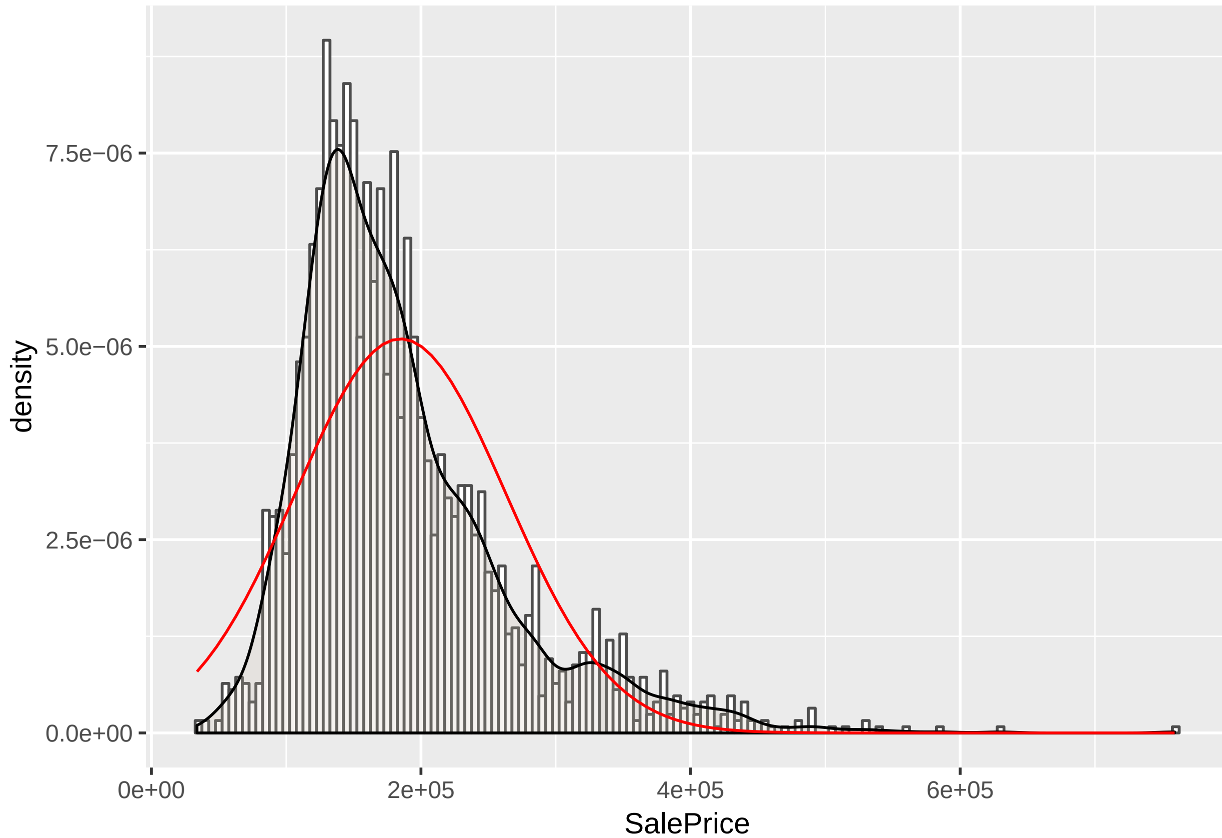
Introduction

Given 80 variables related to houses in Ames, Iowa, this study builds a multiple linear regression model using the training data set to predict the price of houses in the testing data set. The training data set contains 2500 observations of houses with 80 variables (excluding the numbers of observation and including the sale prices). The goal is to build the model and apply it to the testing data set to get predictions of sale prices as close to the real prices as possible.

Methodology

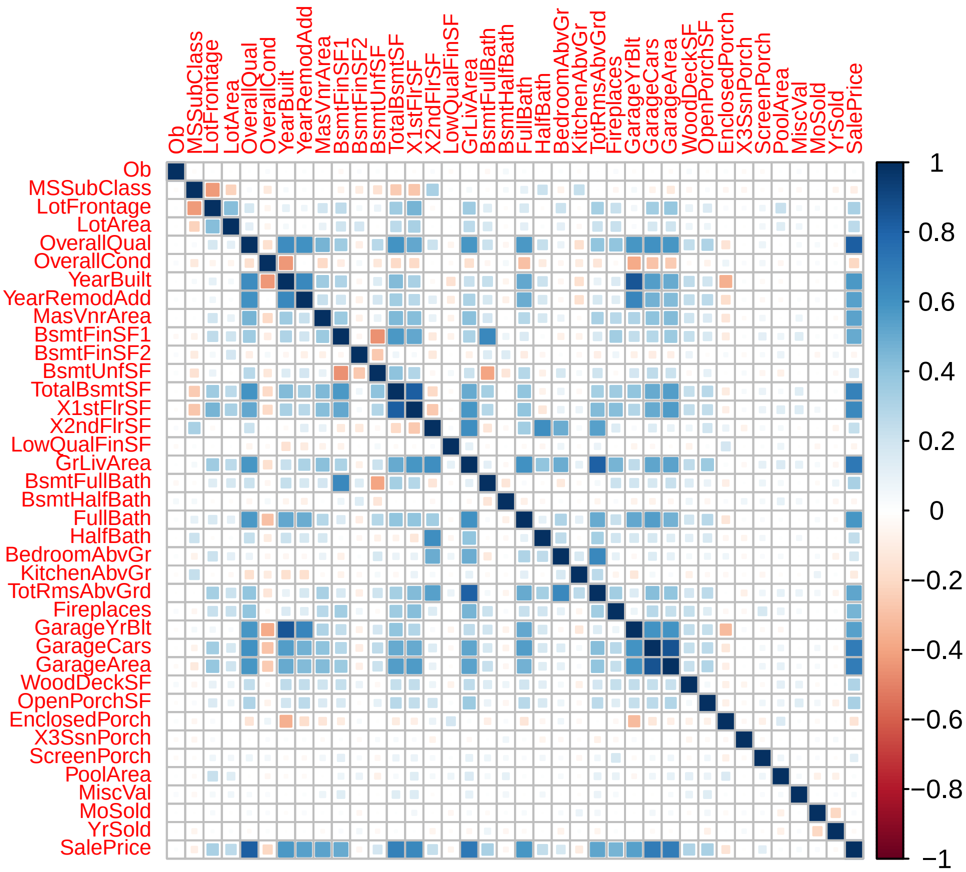
```r
train <- read.csv("HTrainW19Final.csv")

ggplot(data = train, aes(SalePrice)) + geom_histogram(aes(y = ..density..), binwidth = 5000,
                                                      color = "grey30", fill = "white") +
  geom_density(alpha = .2, fill = "antiquewhite3") +
stat_function(fun = dnorm, args = list(mean = mean(train$SalePrice), sd = sd(train$SalePrice)),
              col = "red")
```



Firstly, get a basic idea of what the distribution of sale prices looks like. It is clear that transformation is needed to make the response variable more 'normal'.

```
numeric <- train[,sapply(train, is.numeric)]
corr <- round(cor(numeric, use = "complete.obs"), 4)
corrplot(corr, method = "square", tl.cex = 0.7)
```



Separate numeric variables from the data set and based on the correlation plot, 11 numeric variables are chosen as predictors: `LotArea`, `OverallQual`, `TotalBsmtSF`, `YearBuilt`, `YearRemodAdd`, `GrLivArea`, `FullBath`, `TotRmsAbvGrd`, `Fireplaces`, `GarageCars`, `MasVnrArea`.

```
sum(is.na(train$LotArea))
```

```
## [1] 0
```

```
sum(is.na(train$OverallQual))
```

```
## [1] 0
```

```
sum(is.na(train$TotalBsmtSF))
```

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

```
sum(is.na(train$YearBuilt))
```

```
## [1] 0
```

```
sum(is.na(train$YearRemodAdd))
```

```
## [1] 0
```

```
sum(is.na(train$GrLivArea))
```

```
## [1] 0
```

```
sum(is.na(train$FullBath))
```

```
## [1] 0
```

```
sum(is.na(train$TotRmsAbvGrd))
```

```
## [1] 0
```

```
sum(is.na(train$Fireplaces))
```

```
## [1] 0
```

```
sum(is.na(train$GarageCars))
```

```
## [1] 0
```

```
sum(is.na(train$MasVnrArea))
```

```
## [1] 16
```

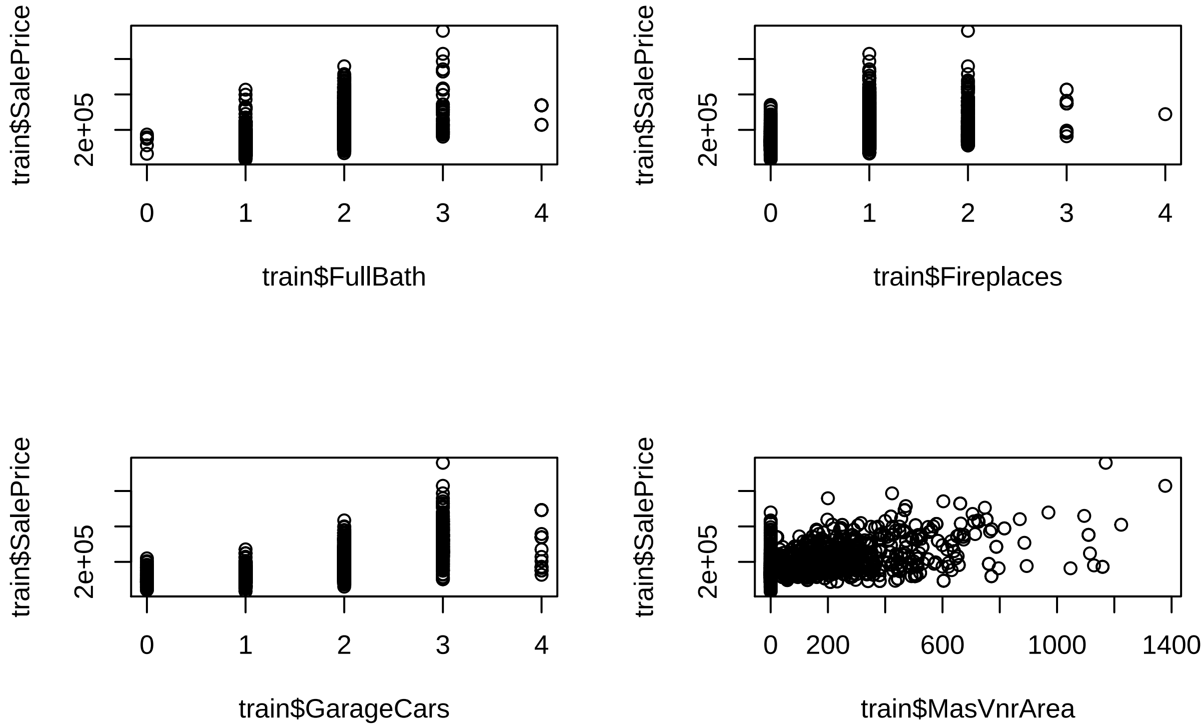
```
train$TotalBsmtSF[which(is.na(train$TotalBsmtSF))] <-
  median(na.omit(train$TotalBsmtSF[which(train$TotalBsmtSF != 0)]))
train$MasVnrArea[which(is.na(train$MasVnrArea))] <-
  median(na.omit(train$MasVnrArea[which(train$MasVnrArea != 0)]))
```

It is important to see if there are any NA's in the selected variables. In this study, there are NA's in `TotalBsmtSF` and `MasVnrArea`. They are replaced by the median (median without 0's and NA's).

```
train$Age1 <- 2019 - train$YearBuilt
train$Age2 <- 2019 - train$YearRemodAdd
```

Two new variables are created to replace YearBuilt and YearRemodAdd: `Age1` and `Age2`, since a year such as 1990 does not make sense in a regression formula. The actual number of years from that year to now is a good choice.

```
par(mfrow=c(2,2))
plot(train$SalePrice ~ train$FullBath)
plot(train$SalePrice ~ train$Fireplaces)
plot(train$SalePrice ~ train$GarageCars)
plot(train$SalePrice ~ train$MasVnrArea)
```




```r
summary(train$MasVnrArea)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0     0.0   106.8   176.0  1378.0
```

```r
train$FullBath <- as.factor(train$FullBath)
train$Fireplaces <- as.factor(train$Fireplaces)
train$GarageCars <- as.factor(train$GarageCars)

med.mas.vnr <- median(train$MasVnrArea[which(train$MasVnrArea != 0)])
for(i in 1:nrow(train)){
  if(train$MasVnrArea[i] == 0) {train$MasVnr[i] <- "No MasVnr"}
  if(train$MasVnrArea[i] != 0 & train$MasVnrArea[i] <= med.mas.vnr) {train$MasVnr[i] <- "Small MasVnr"}
  if(train$MasVnrArea[i] != 0 & train$MasVnrArea[i] > med.mas.vnr) {train$MasVnr[i] <- "Large MasVnr"}
}
train$MasVnr <- as.factor(train$MasVnr)
```

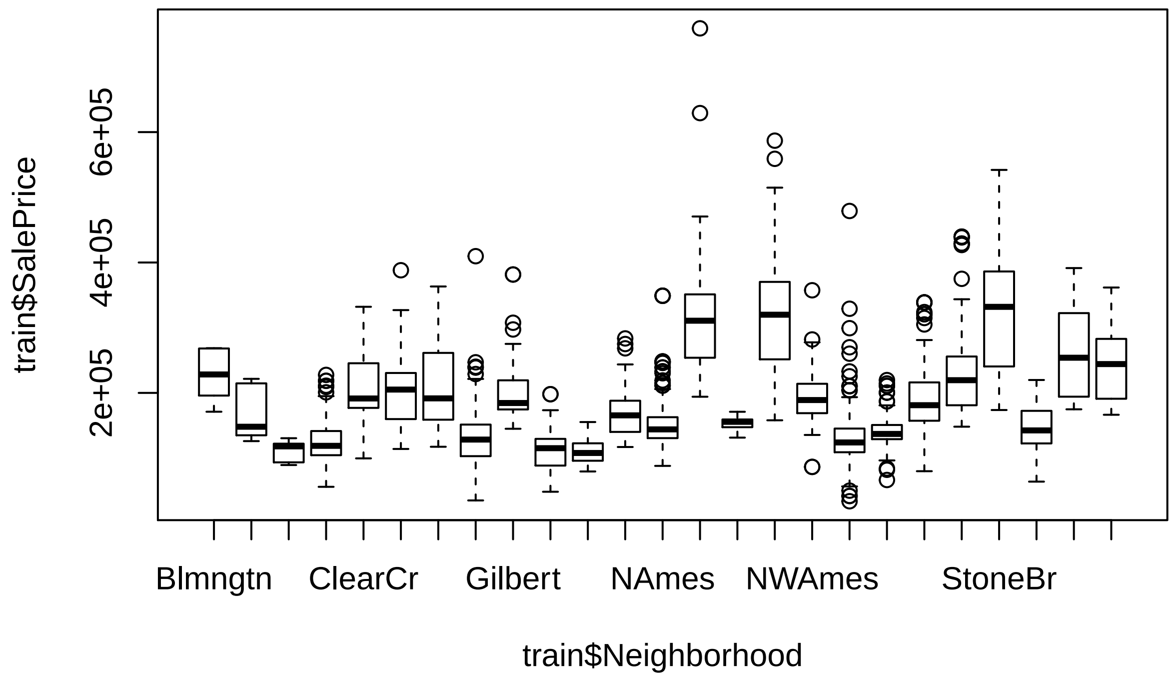
After analyzing some of the selected predictors, it is clear that turning `FullBath`, `Fireplaces`, and `GarageCars` into factor variables would be a good choice since they represent the number of full bathrooms, fireplaces, and cars in garage, therefore they all have several categories and it would be better to use them as categorical predictors rather than numerical predictors. In addition, there are too many 0's in the variable `MasVnrArea`, and even its median turns out to be 0. Thus considering grouping the numerical predictor `MasVnrArea` to create a new categorical (factor) predictor `MasVnr` with 3 categories: 'No MasVnr', 'Small', and 'Large'.

```r
summary(lm(SalePrice ~ Neighborhood, data = train))
```

```
##
## Call:
```

```
## lm(formula = SalePrice ~ Neighborhood, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -156350  -28280   -4276   21256  437172
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           224769      10383  21.648  < 2e-16 ***
## NeighborhoodBlueste   -60302      20181  -2.988  0.00283 **
## NeighborhoodBrDale   -113507      14542  -7.806 8.67e-15 ***
## NeighborhoodBrkSide   -98360      11632  -8.456  < 2e-16 ***
## NeighborhoodClearCr   -17311      13440  -1.288  0.19788
## NeighborhoodCollgCr   -23446      10947  -2.142  0.03231 *
## NeighborhoodCrawfor   -17663      11931  -1.480  0.13891
## NeighborhoodEdwards   -93754      11038  -8.494  < 2e-16 ***
## NeighborhoodGilbert   -26474      11304  -2.342  0.01925 *
## NeighborhoodIDOTRR   -113209      11895  -9.517  < 2e-16 ***
## NeighborhoodMeadowV  -116036      13369  -8.680  < 2e-16 ***
## NeighborhoodMitchel   -54896      11828  -4.641 3.64e-06 ***
## NeighborhoodNAmes     -74529      10739  -6.940 4.99e-12 ***
## NeighborhoodNoRidge    97171      12421   7.823 7.55e-15 ***
## NeighborhoodNPkVill   -72094      18232  -3.954 7.89e-05 ***
## NeighborhoodNridgHt    89148      11189   7.968 2.45e-15 ***
## NeighborhoodNWAmes    -30633      11382  -2.691  0.00716 **
## NeighborhoodOldTown   -93332      11032  -8.460  < 2e-16 ***
## NeighborhoodSawyer    -84331      11330  -7.443 1.35e-13 ***
## NeighborhoodSawyerW   -29914      11474  -2.607  0.00919 **
## NeighborhoodSomerst     6273      11215   0.559  0.57596
## NeighborhoodStoneBr   105323      13057   8.066 1.12e-15 ***
## NeighborhoodSWISU     -80436      12804  -6.282 3.94e-10 ***
## NeighborhoodTimber     39913      12167   3.280  0.00105 **
## NeighborhoodVeenker    18075      15367   1.176  0.23963
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 51910 on 2475 degrees of freedom
## Multiple R-squared:  0.5643, Adjusted R-squared:   0.56
## F-statistic: 133.5 on 24 and 2475 DF,  p-value: < 2.2e-16
```

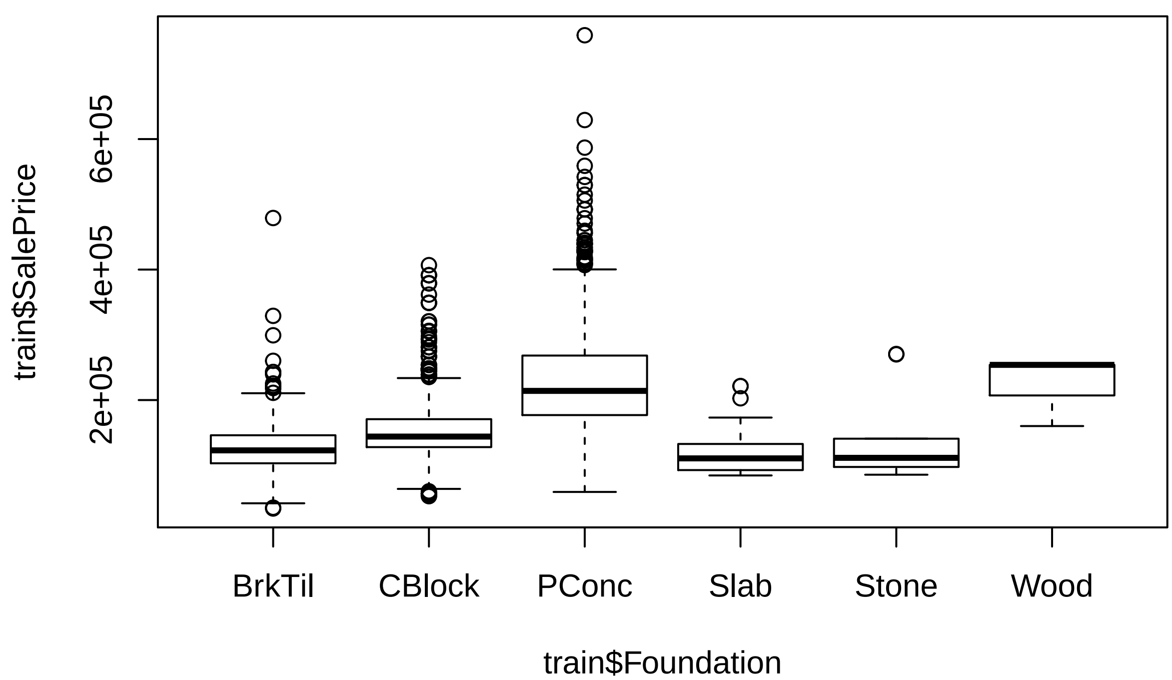
```r
plot(train$SalePrice ~ train$Neighborhood)
```




```r
summary(lm(SalePrice ~ Foundation, data = train))
```

```
##
## Call:
## lm(formula = SalePrice ~ Foundation, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -172652  -38274  -10470   23986  527262
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        129235       3945  32.758  < 2e-16 ***
## FoundationCBlock    24206       4433   5.460 5.24e-08 ***
## FoundationPConc    102616       4409  23.275  < 2e-16 ***
## FoundationSlab     -10852      10250  -1.059   0.2898
## FoundationStone      2866      17956   0.160   0.8732
## FoundationWood      93489      38046   2.457   0.0141 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 65540 on 2494 degrees of freedom
## Multiple R-squared:  0.3001, Adjusted R-squared:  0.2987
## F-statistic: 213.9 on 5 and 2494 DF,  p-value: < 2.2e-16
```

```r
plot(train$SalePrice ~ train$Foundation)
```




```r
sum(is.na(train$Neighborhood))
```

```
## [1] 0
```

```r
sum(is.na(train$Foundation))
```

```
## [1] 0
```

Summary (focusing on R^2 values) and plots are analyzed to select other categorical predictors. In this study, `Neighborhood` and `Foundation` are chosen since they have relatively high R^2 values which means that they could explain a lot of variations in the response variable (see also the box plots). In addition, they have no NA values, which is another important reason for them to be chosen.

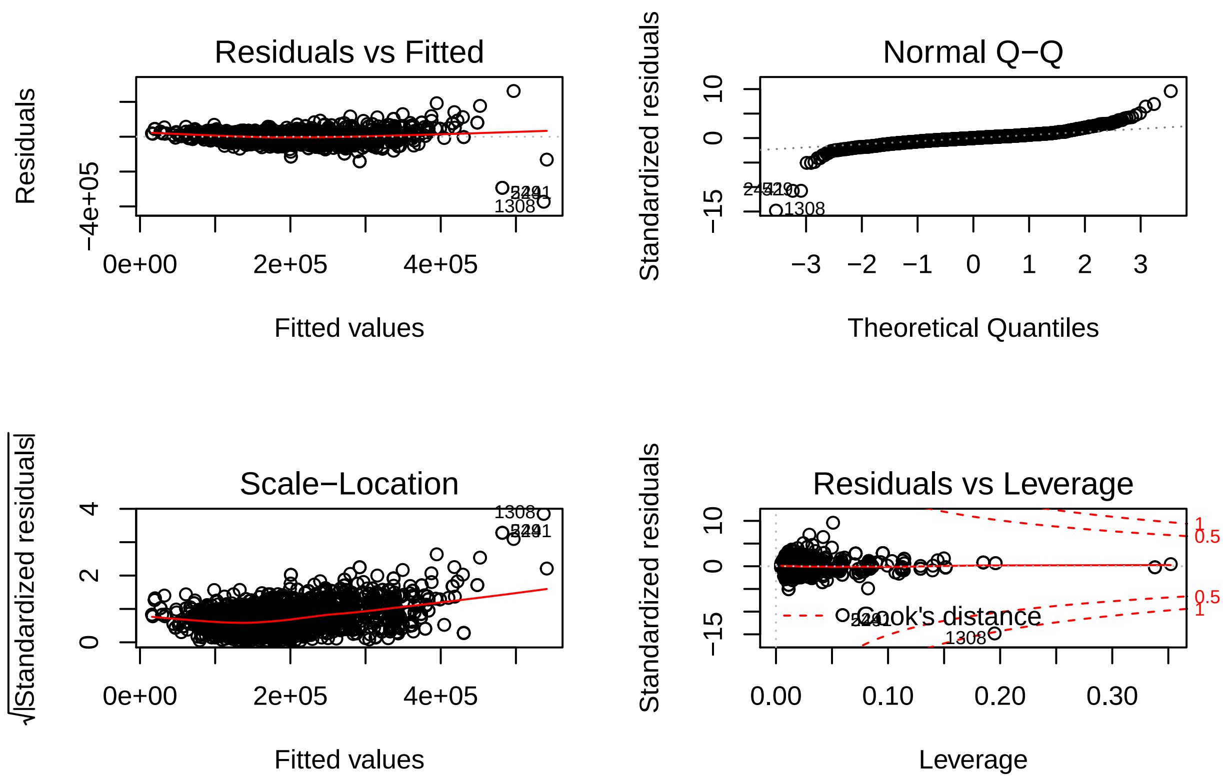
```r
model1 <- lm(SalePrice ~ LotArea + OverallQual + TotalBsmtSF + Age1 + Age2 + GrLivArea +
               FullBath + TotRmsAbvGrd + Fireplaces + Neighborhood + Foundation + GarageCars +
               MasVnr, data = train)
summary(model1)
```

```
##
## Call:
## lm(formula = SalePrice ~ LotArea + OverallQual + TotalBsmtSF +
##     Age1 + Age2 + GrLivArea + FullBath + TotRmsAbvGrd + Fireplaces +
##     Neighborhood + Foundation + GarageCars + MasVnr, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -372933  -12039    -115   11698  262158
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -5.460e+03  1.333e+04  -0.410 0.682106
## LotArea              4.761e-01  6.345e-02   7.503 8.66e-14 ***
## OverallQual          1.686e+04  7.646e+02  22.051  < 2e-16 ***
## TotalBsmtSF          2.187e+01  1.865e+00  11.732  < 2e-16 ***
## Age1                -2.434e+02  5.073e+01  -4.798 1.70e-06 ***
## Age2                -2.152e+02  3.949e+01  -5.449 5.56e-08 ***
## GrLivArea            3.849e+01  2.618e+00  14.705  < 2e-16 ***
## FullBath1           -4.991e+03  1.033e+04  -0.483 0.629184
## FullBath2           -7.453e+03  1.042e+04  -0.715 0.474595
## FullBath3            7.847e+03  1.139e+04   0.689 0.490778
## FullBath4            1.301e+04  1.566e+04   0.831 0.406266
## TotRmsAbvGrd         3.187e+02  6.593e+02   0.483 0.628936
## Fireplaces1          4.386e+03  1.430e+03   3.066 0.002192 **
## Fireplaces2          1.851e+04  2.605e+03   7.107 1.55e-12 ***
## Fireplaces3         -4.753e+03  8.900e+03  -0.534 0.593338
## Fireplaces4          1.758e+04  2.893e+04   0.608 0.543513
## NeighborhoodBlueste -2.392e+03  1.123e+04  -0.213 0.831326
## NeighborhoodBrDale  -2.368e+04  8.654e+03  -2.736 0.006255 **
## NeighborhoodBrkSide  4.312e+03  7.101e+03   0.607 0.543771
## NeighborhoodClearCr  9.416e+03  7.742e+03   1.216 0.224036
## NeighborhoodCollgCr  6.293e+03  6.135e+03   1.026 0.305151
## NeighborhoodCrawfor  2.487e+04  7.012e+03   3.547 0.000397 ***
## NeighborhoodEdwards -6.924e+03  6.549e+03  -1.057 0.290499
## NeighborhoodGilbert  1.038e+03  6.325e+03   0.164 0.869606
## NeighborhoodIDOTRR  -6.271e+02  7.345e+03  -0.085 0.931969
## NeighborhoodMeadowV -2.335e+03  7.811e+03  -0.299 0.765015
## NeighborhoodMitchel -3.019e+03  6.764e+03  -0.446 0.655350
## NeighborhoodNAmes    2.351e+03  6.438e+03   0.365 0.714991
## NeighborhoodNoRidge  3.815e+04  7.023e+03   5.432 6.13e-08 ***
## NeighborhoodNPkVill -6.864e+03  1.020e+04  -0.673 0.501190
## NeighborhoodNridgHt  3.645e+04  6.229e+03   5.851 5.54e-09 ***
## NeighborhoodNWAmes   3.968e+03  6.567e+03   0.604 0.545752
## NeighborhoodOldTown -4.784e+03  7.004e+03  -0.683 0.494587
## NeighborhoodSawyer   2.564e+03  6.643e+03   0.386 0.699614
## NeighborhoodSawyerW  2.084e+03  6.515e+03   0.320 0.749053
## NeighborhoodSomerst  8.719e+03  6.232e+03   1.399 0.161902
## NeighborhoodStoneBr  5.604e+04  7.263e+03   7.715 1.74e-14 ***
## NeighborhoodSWISU   -3.518e+03  7.706e+03  -0.457 0.648024
## NeighborhoodTimber   1.651e+04  6.819e+03   2.421 0.015542 *
## NeighborhoodVeenker  2.826e+04  8.734e+03   3.236 0.001230 **
## FoundationCBlock     8.466e+03  2.502e+03   3.383 0.000728 ***
## FoundationPConc      7.037e+03  2.864e+03   2.457 0.014068 *
## FoundationSlab       1.695e+04  5.125e+03   3.308 0.000953 ***
## FoundationStone     -4.721e+03  7.868e+03  -0.600 0.548500
## FoundationWood       1.546e+04  1.676e+04   0.922 0.356447
## GarageCars1          3.952e+03  2.816e+03   1.403 0.160605
## GarageCars2          5.853e+03  2.948e+03   1.985 0.047233 *
## GarageCars3          4.049e+04  3.759e+03  10.772  < 2e-16 ***
## GarageCars4          6.650e+04  7.724e+03   8.610  < 2e-16 ***
## MasVnrNo MasVnr     -1.865e+03  1.858e+03  -1.004 0.315450
```

```
## MasVnrSmall MasVnr  -5.701e+03  1.916e+03  -2.975 0.002955 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 28070 on 2449 degrees of freedom
## Multiple R-squared:  0.8739, Adjusted R-squared:  0.8713
## F-statistic: 339.5 on 50 and 2449 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(model1)
```




```r
vif(model1)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## LotArea      1.357977  1        1.165323
## OverallQual  3.807552  1        1.951295
## TotalBsmtSF  2.321089  1        1.523512
## Age1         7.601896  1        2.757154
## Age2         2.193649  1        1.481097
## GrLivArea    5.660557  1        2.379192
## FullBath     4.354318  4        1.201891
## TotRmsAbvGrd 3.404723  1        1.845189
## Fireplaces   2.139517  4        1.099739
## Neighborhood 53.928859 24       1.086625
## Foundation   6.918490  5        1.213392
## GarageCars   5.005511  4        1.223013
## MasVnr       1.925102  2        1.177914
```

This is the first model without any transformations or interaction terms. Clearly, transformations are needed and the predictors also needs some modifications to improve the model and the R^2 value.

```
summary(powerTransform(cbind(train$SalePrice, train$LotArea, train$GrLivArea)~1))
```

```
## bcPower Transformations to Multinormality
##    Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## Y1   -0.0453        0.00      -0.1088       0.0182
## Y2    0.0659        0.07       0.0325       0.0993
## Y3   -0.0744        0.00      -0.1564       0.0076
##
## Likelihood ratio test that transformation parameters are equal to 0
##  (all log transformations)
##                                LRT df       pval
## LR test, lambda = (0 0 0) 18.96137  3 0.00027847
##
## Likelihood ratio test that no transformations are needed
##                                LRT df       pval
## LR test, lambda = (1 1 1) 5564.304  3 < 2.22e-16
```

```
tSalePrice <- log(train$SalePrice)
tLotArea <- log(train$LotArea)
tGrLivArea <- log(train$GrLivArea)
```

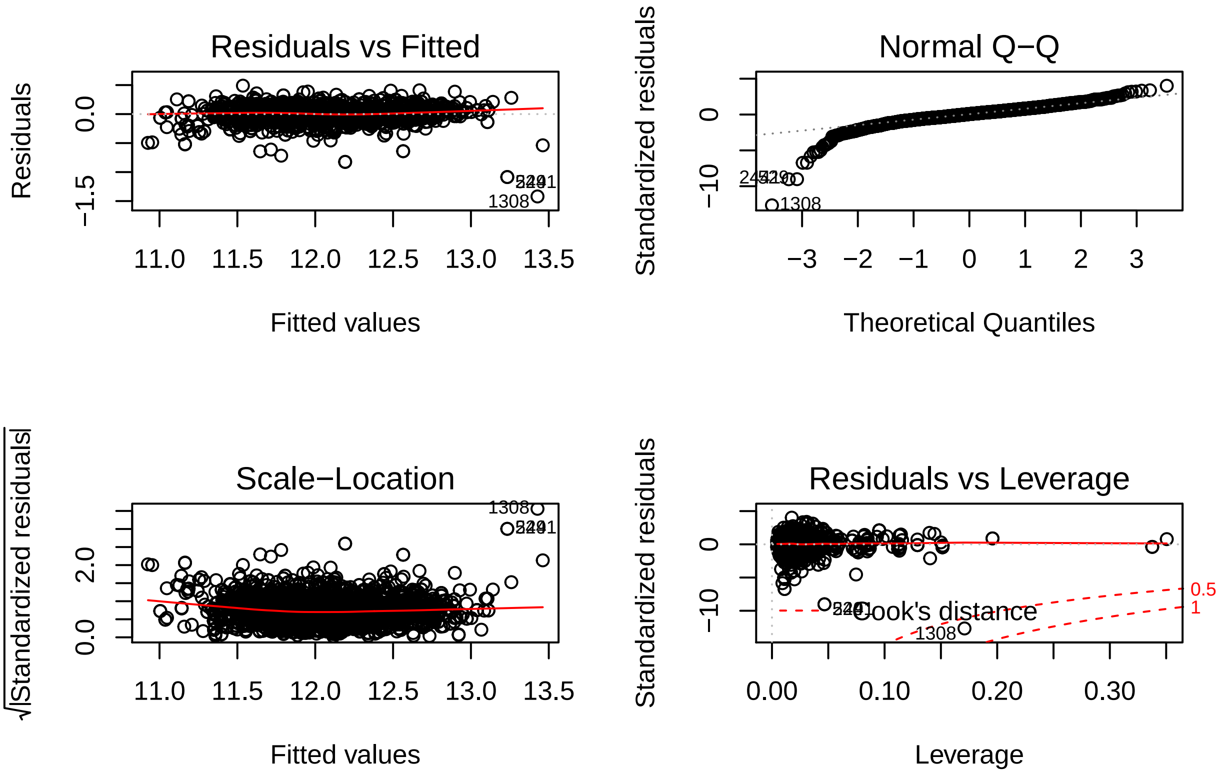
Using box-cox method, the suggested transformation for `SalePrice`, `LotArea`, and `GrLivArea` is log transformation. Note that the other numerical predictors are not transformed because they have 0's.

```
model2_1 <- lm(tSalePrice ~ tLotArea + OverallQual + TotalBsmtSF + Age1 + Age2 +
               tGrLivArea + FullBath + TotRmsAbvGrd + Fireplaces + Neighborhood +
               Foundation + GarageCars + MasVnr, data = train)
summary(model2_1)
```

```
##
## Call:
## lm(formula = tSalePrice ~ tLotArea + OverallQual + TotalBsmtSF +
##     Age1 + Age2 + tGrLivArea + FullBath + TotRmsAbvGrd + Fireplaces +
##     Neighborhood + Foundation + GarageCars + MasVnr, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.42012 -0.05665  0.00848  0.06876  0.48980
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          8.207e+00  1.262e-01  65.024  < 2e-16 ***
## tLotArea             7.016e-02  7.288e-03   9.627  < 2e-16 ***
## OverallQual          8.838e-02  3.368e-03  26.245  < 2e-16 ***
## TotalBsmtSF          9.129e-05  8.165e-06  11.181  < 2e-16 ***
## Age1                -1.926e-03  2.243e-04  -8.585  < 2e-16 ***
## Age2                -1.545e-03  1.732e-04  -8.923  < 2e-16 ***
## tGrLivArea           3.605e-01  1.807e-02  19.954  < 2e-16 ***
## FullBath1           -3.372e-02  4.535e-02  -0.744 0.457248
## FullBath2           -3.295e-02  4.572e-02  -0.721 0.471209
## FullBath3           -1.498e-02  4.973e-02  -0.301 0.763301
## FullBath4            7.092e-02  6.858e-02   1.034 0.301124
## TotRmsAbvGrd        -3.399e-03  2.906e-03  -1.170 0.242225
```

```
## Fireplaces1          2.348e-02  6.337e-03   3.705 0.000216 ***
## Fireplaces2          7.125e-02  1.138e-02   6.263 4.45e-10 ***
## Fireplaces3         -6.379e-02  3.885e-02  -1.642 0.100711
## Fireplaces4          5.943e-02  1.268e-01   0.469 0.639383
## NeighborhoodBlueste -1.647e-02  4.933e-02  -0.334 0.738436
## NeighborhoodBrDale  -1.446e-01  3.810e-02  -3.794 0.000152 ***
## NeighborhoodBrkSide -2.900e-02  3.133e-02  -0.926 0.354610
## NeighborhoodClearCr -4.069e-03  3.501e-02  -0.116 0.907502
## NeighborhoodCollgCr -3.835e-02  2.803e-02  -1.368 0.171362
## NeighborhoodCrawfor  8.206e-02  3.132e-02   2.620 0.008843 **
## NeighborhoodEdwards -1.076e-01  2.938e-02  -3.663 0.000254 ***
## NeighborhoodGilbert -7.950e-02  2.906e-02  -2.735 0.006275 **
## NeighborhoodIDOTRR  -1.096e-01  3.259e-02  -3.364 0.000779 ***
## NeighborhoodMeadowV -4.608e-02  3.428e-02  -1.344 0.179000
## NeighborhoodMitchel -5.388e-02  3.076e-02  -1.752 0.079922 .
## NeighborhoodNAmes   -3.495e-02  2.895e-02  -1.207 0.227424
## NeighborhoodNoRidge  4.806e-02  3.133e-02   1.534 0.125190
## NeighborhoodNPkVill -4.721e-02  4.476e-02  -1.055 0.291671
## NeighborhoodNridgHt  2.842e-02  2.800e-02   1.015 0.310112
## NeighborhoodNWAmes  -4.832e-02  2.977e-02  -1.623 0.104668
## NeighborhoodOldTown -8.534e-02  3.092e-02  -2.760 0.005831 **
## NeighborhoodSawyer  -3.288e-02  2.994e-02  -1.098 0.272251
## NeighborhoodSawyerW -5.641e-02  2.961e-02  -1.905 0.056868 .
## NeighborhoodSomerst -2.637e-02  2.779e-02  -0.949 0.342792
## NeighborhoodStoneBr  9.209e-02  3.237e-02   2.845 0.004474 **
## NeighborhoodSWISU   -2.310e-02  3.396e-02  -0.680 0.496467
## NeighborhoodTimber  -7.658e-03  3.090e-02  -0.248 0.804264
## NeighborhoodVeenker  4.409e-02  3.923e-02   1.124 0.261263
## FoundationCBlock     7.375e-02  1.100e-02   6.707 2.46e-11 ***
## FoundationPConc      5.439e-02  1.258e-02   4.324 1.59e-05 ***
## FoundationSlab       3.253e-02  2.246e-02   1.448 0.147632
## FoundationStone     -2.315e-03  3.454e-02  -0.067 0.946580
## FoundationWood       1.155e-01  7.336e-02   1.574 0.115582
## GarageCars1          8.615e-02  1.240e-02   6.949 4.68e-12 ***
## GarageCars2          1.172e-01  1.303e-02   8.997  < 2e-16 ***
## GarageCars3          1.882e-01  1.677e-02  11.224  < 2e-16 ***
## GarageCars4          2.855e-01  3.394e-02   8.413  < 2e-16 ***
## MasVnrNo MasVnr      4.518e-03  8.154e-03   0.554 0.579568
## MasVnrSmall MasVnr   3.325e-04  8.391e-03   0.040 0.968399
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1231 on 2449 degrees of freedom
## Multiple R-squared:  0.9032, Adjusted R-squared:  0.9012
## F-statistic: 456.9 on 50 and 2449 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(model2_1)
```

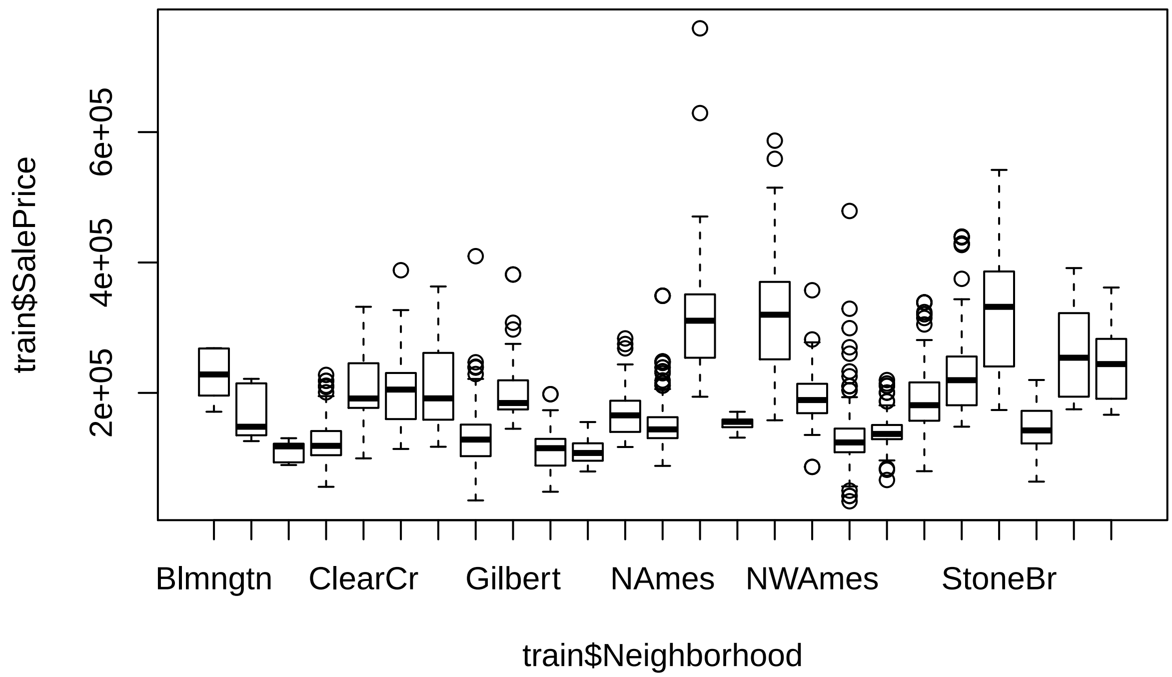



```r
vif(model2_1)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## tLotArea     2.489696  1        1.577877
## OverallQual  3.838875  1        1.959305
## TotalBsmtSF  2.313517  1        1.521025
## Age1         7.723897  1        2.779190
## Age2         2.192387  1        1.480671
## tGrLivArea   5.612234  1        2.369015
## FullBath     4.300606  4        1.200028
## TotRmsAbvGrd 3.436977  1        1.853908
## Fireplaces   2.113560  4        1.098062
## Neighborhood 81.757402 24       1.096085
## Foundation   6.940698  5        1.213781
## GarageCars   5.267476  4        1.230836
## MasVnr       1.924094  2        1.177759
```

This is the second model, with transformations added. This model has higher R^2 and better diagnostics. But it seems that grouping the variable `Neighborhood` would be essential to avoid high vif and to fix the problem of too many insignificant slopes for `Neighborhood`.

```r
plot(train$SalePrice ~ train$Neighborhood)
```



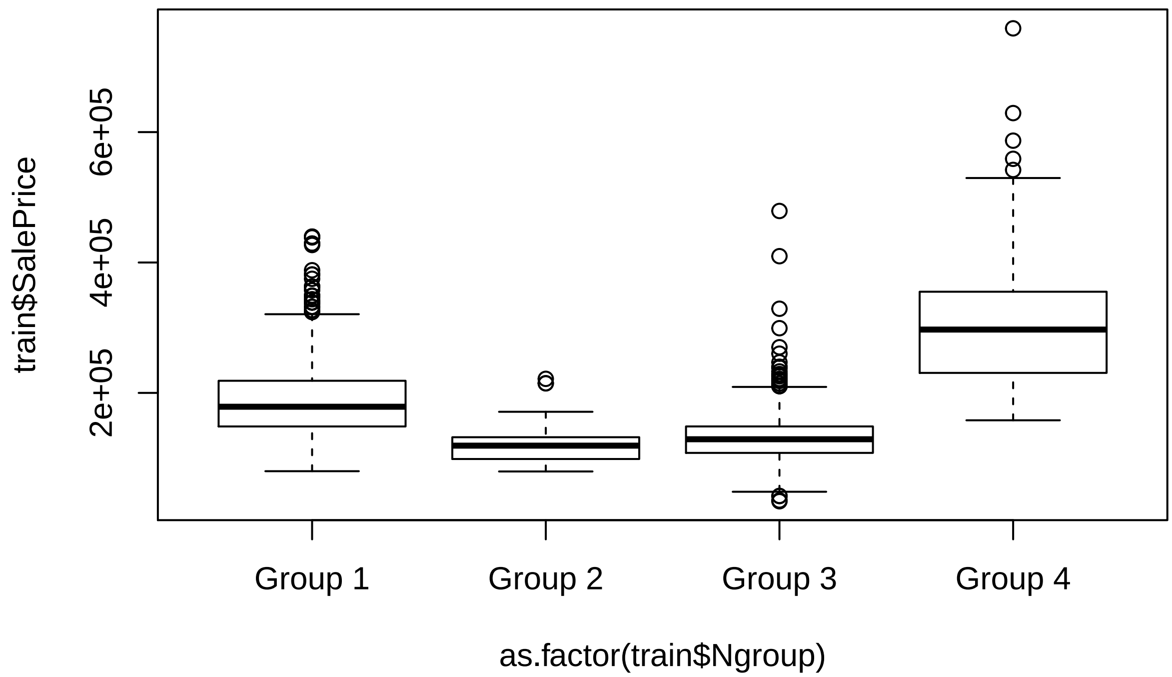

```r
N <- as.integer(train$Neighborhood)
for(i in 1:nrow(train)){
   if(N[i] %in% c(1,5,6,7,9,12,13,17,20,21)){
     train$Ngroup[i] <- "Group 1"
   }else if(N[i] %in% c(2,3,11,15)){
     train$Ngroup[i] <- "Group 2"
   }else if(N[i] %in% c(4,8,10,18,19,23)){
     train$Ngroup[i] <- "Group 3"
   }else{
     train$Ngroup[i] <- "Group 4"
   }
}

train$Ngroup <- as.factor(train$Ngroup)

table(train$Ngroup)
```

```
##
## Group 1 Group 2 Group 3 Group 4
##    1328      85     743     344
```

```r
plot(train$SalePrice~as.factor(train$Ngroup))
```




```r
summary(lm(train$SalePrice~train$Ngroup))
```

```
##
## Call:
## lm(formula = train$SalePrice ~ train$Ngroup)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -145377  -35843   -6657   26553  455749
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           188343       1550  121.53   <2e-16 ***
## train$NgroupGroup 2   -66731       6319  -10.56   <2e-16 ***
## train$NgroupGroup 3   -57398       2587  -22.18   <2e-16 ***
## train$NgroupGroup 4   115021       3417   33.66   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 56480 on 2496 degrees of freedom
## Multiple R-squared:  0.4799, Adjusted R-squared:  0.4793
## F-statistic: 767.8 on 3 and 2496 DF,  p-value: < 2.2e-16
```

After examining the first box plot (`SalePrice` vs `Neighborhood`), the neighborhoods are divided into 4 groups based on the distribution of sale prices. Group 1: 'Blmngtn', 'ClearCr', 'CollgCr', 'Crawfor', 'Gilbert', 'Mitchel', 'NAmes', 'NWAmes', 'SawyerW', 'Somerst'; Group 2: 'Blueste', 'BrDale', 'MeadoV', 'NpkVill'; Group 3: 'BrkSide', 'Edwards', 'IDOTRR', 'OldTown', 'Sawyer', 'SWISU'; Group 4: 'NoRidge', 'NridgHt', 'StoneBr', 'Timber', 'Veenker'.

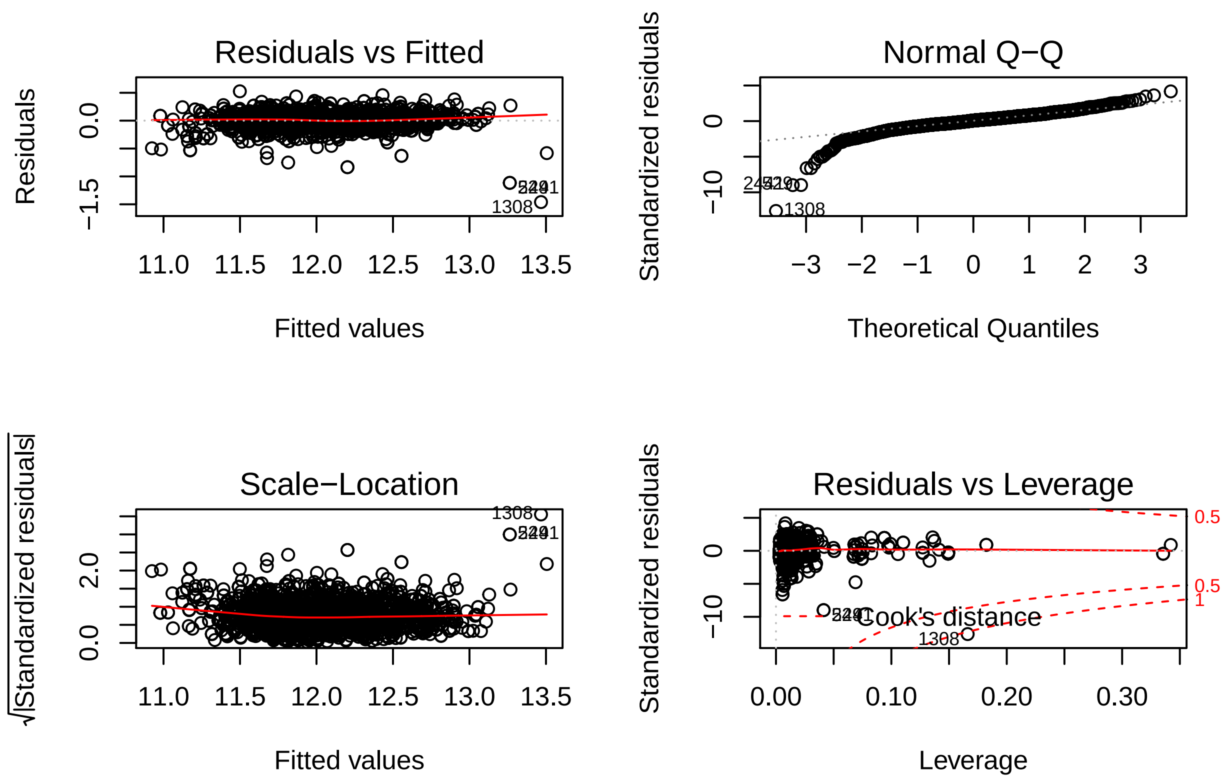
As we can see, the new factor variable `Ngroup` itself explains a lot of variations in the response variable.

```
model2_2 <- lm(tSalePrice ~ tLotArea + OverallQual + TotalBsmtSF + Age1 + Age2 +
               tGrLivArea + FullBath + TotRmsAbvGrd + Fireplaces + Ngroup +
               Foundation + GarageCars + MasVnr, data = train)
summary(model2_2)
```

```
##
## Call:
## lm(formula = tSalePrice ~ tLotArea + OverallQual + TotalBsmtSF +
##     Age1 + Age2 + tGrLivArea + FullBath + TotRmsAbvGrd + Fireplaces +
##     Ngroup + Foundation + GarageCars + MasVnr, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.46024 -0.05717  0.00966  0.07027  0.52645
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           8.179e+00  1.266e-01  64.620  < 2e-16 ***
## tLotArea              5.675e-02  6.666e-03   8.514  < 2e-16 ***
## OverallQual           9.101e-02  3.321e-03  27.402  < 2e-16 ***
## TotalBsmtSF           1.012e-04  8.072e-06  12.538  < 2e-16 ***
## Age1                 -1.574e-03  1.866e-04  -8.434  < 2e-16 ***
## Age2                 -1.471e-03  1.742e-04  -8.441  < 2e-16 ***
## tGrLivArea            3.687e-01  1.799e-02  20.491  < 2e-16 ***
## FullBath1            -2.794e-02  4.593e-02  -0.608   0.5430
## FullBath2            -3.267e-02  4.621e-02  -0.707   0.4797
## FullBath3            -7.502e-03  5.026e-02  -0.149   0.8814
## FullBath4             5.091e-02  7.035e-02   0.724   0.4693
## TotRmsAbvGrd         -4.953e-03  2.925e-03  -1.694   0.0905 .
## Fireplaces1           3.258e-02  6.186e-03   5.267 1.50e-07 ***
## Fireplaces2           8.769e-02  1.133e-02   7.740 1.44e-14 ***
## Fireplaces3          -5.848e-02  3.973e-02  -1.472   0.1411
## Fireplaces4           7.264e-02  1.276e-01   0.569   0.5691
## NgroupGroup 2        -5.330e-02  1.788e-02  -2.981   0.0029 **
## NgroupGroup 3        -5.006e-02  7.956e-03  -6.292 3.69e-10 ***
## NgroupGroup 4         6.922e-02  9.418e-03   7.350 2.69e-13 ***
## FoundationCBlock      7.211e-02  1.071e-02   6.735 2.03e-11 ***
## FoundationPConc       5.643e-02  1.278e-02   4.416 1.05e-05 ***
## FoundationSlab        3.901e-02  2.248e-02   1.736   0.0828 .
## FoundationStone      -2.480e-02  3.527e-02  -0.703   0.4821
## FoundationWood        8.582e-02  7.480e-02   1.147   0.2514
## GarageCars1           9.679e-02  1.243e-02   7.786 1.01e-14 ***
## GarageCars2           1.258e-01  1.287e-02   9.774  < 2e-16 ***
## GarageCars3           1.985e-01  1.651e-02  12.021  < 2e-16 ***
## GarageCars4           2.889e-01  3.437e-02   8.405  < 2e-16 ***
## MasVnrNo MasVnr       1.487e-02  7.912e-03   1.880   0.0603 .
## MasVnrSmall MasVnr    6.459e-03  8.358e-03   0.773   0.4397
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1267 on 2470 degrees of freedom
## Multiple R-squared:  0.8966, Adjusted R-squared:  0.8953
```

```
## F-statistic: 738.2 on 29 and 2470 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(model2_2)
```




```
vif(model2_2)
```

```
##                 GVIF Df GVIF^(1/(2*Df))
## tLotArea     1.966472  1        1.402309
## OverallQual  3.524755  1        1.877433
## TotalBsmtSF  2.134215  1        1.460895
## Age1         5.045820  1        2.246290
## Age2         2.094033  1        1.447077
## tGrLivArea   5.253185  1        2.291983
## FullBath     3.588080  4        1.173161
## TotRmsAbvGrd 3.286231  1        1.812797
## Fireplaces   1.721764  4        1.070278
## Ngroup       4.795043  3        1.298570
## Foundation   4.812399  5        1.170136
## GarageCars   4.045045  4        1.190873
## MasVnr       1.642264  2        1.132037
```

The model looks more valid, but the R^2 value still needs to be higher, therefore consider adding interaction terms to build the final model.

```
model3 <- lm(tSalePrice ~ MasVnr:tLotArea + MasVnr:OverallQual + MasVnr:TotalBsmtSF +
               MasVnr:Age1 + MasVnr:Age2 + MasVnr:tGrLivArea + FullBath +
               MasVnr:TotRmsAbvGrd + Fireplaces + MasVnr:Ngroup + Foundation +
```

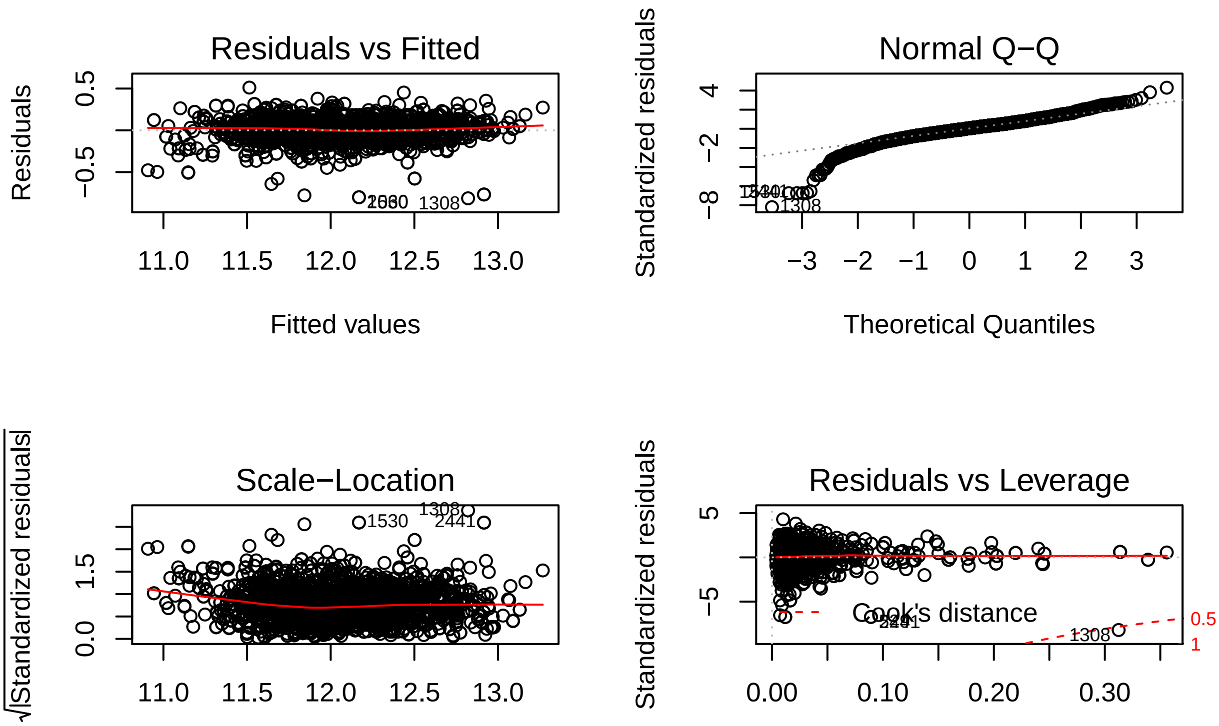
```r
              MasVnr:GarageCars, data = train)
summary(model3)
```

```
##
## Call:
## lm(formula = tSalePrice ~ MasVnr:tLotArea + MasVnr:OverallQual +
##     MasVnr:TotalBsmtSF + MasVnr:Age1 + MasVnr:Age2 + MasVnr:tGrLivArea +
##     FullBath + MasVnr:TotRmsAbvGrd + Fireplaces + MasVnr:Ngroup +
##     Foundation + MasVnr:GarageCars, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.81364 -0.05881  0.00696  0.06448  0.51144
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      8.353e+00  1.223e-01  68.308  < 2e-16
## FullBath1                       -3.920e-02  4.370e-02  -0.897 0.369759
## FullBath2                       -4.881e-02  4.391e-02  -1.112 0.266417
## FullBath3                       -7.052e-03  4.792e-02  -0.147 0.883020
## FullBath4                        6.518e-02  7.340e-02   0.888 0.374587
## Fireplaces1                      2.716e-02  5.944e-03   4.569 5.14e-06
## Fireplaces2                      8.991e-02  1.096e-02   8.207 3.62e-16
## Fireplaces3                     -2.670e-02  3.782e-02  -0.706 0.480280
## Fireplaces4                      1.635e-01  1.214e-01   1.346 0.178452
## FoundationCBlock                 6.773e-02  1.032e-02   6.562 6.46e-11
## FoundationPConc                  5.770e-02  1.213e-02   4.758 2.07e-06
## FoundationSlab                   8.135e-02  2.177e-02   3.737 0.000190
## FoundationStone                 -3.339e-02  3.356e-02  -0.995 0.319874
## FoundationWood                   9.163e-02  7.148e-02   1.282 0.200002
## MasVnrLarge MasVnr:tLotArea      5.245e-02  1.367e-02   3.836 0.000128
## MasVnrNo MasVnr:tLotArea         5.550e-02  8.119e-03   6.835 1.03e-11
## MasVnrSmall MasVnr:tLotArea      7.719e-02  1.300e-02   5.936 3.34e-09
## MasVnrLarge MasVnr:OverallQual   9.854e-02  7.910e-03  12.457  < 2e-16
## MasVnrNo MasVnr:OverallQual      9.367e-02  3.910e-03  23.956  < 2e-16
## MasVnrSmall MasVnr:OverallQual   8.913e-02  7.975e-03  11.175  < 2e-16
## MasVnrLarge MasVnr:TotalBsmtSF  -3.255e-06  1.365e-05  -0.238 0.811572
## MasVnrNo MasVnr:TotalBsmtSF      1.813e-04  1.091e-05  16.619  < 2e-16
## MasVnrSmall MasVnr:TotalBsmtSF   1.265e-04  1.570e-05   8.058 1.20e-15
## MasVnrLarge MasVnr:Age1          3.720e-04  6.469e-04   0.575 0.565373
## MasVnrNo MasVnr:Age1            -1.630e-03  1.910e-04  -8.534  < 2e-16
## MasVnrSmall MasVnr:Age1         -2.410e-03  6.400e-04  -3.765 0.000170
## MasVnrLarge MasVnr:Age2         -3.226e-03  6.140e-04  -5.254 1.62e-07
## MasVnrNo MasVnr:Age2            -1.462e-03  1.824e-04  -8.012 1.74e-15
## MasVnrSmall MasVnr:Age2         -1.239e-03  5.285e-04  -2.345 0.019125
## MasVnrLarge MasVnr:tGrLivArea    4.128e-01  2.559e-02  16.131  < 2e-16
## MasVnrNo MasVnr:tGrLivArea       3.242e-01  1.831e-02  17.709  < 2e-16
## MasVnrSmall MasVnr:tGrLivArea    3.495e-01  2.504e-02  13.955  < 2e-16
## MasVnrLarge MasVnr:TotRmsAbvGrd -1.102e-02  4.888e-03  -2.254 0.024304
## MasVnrNo MasVnr:TotRmsAbvGrd     6.200e-03  3.331e-03   1.861 0.062815
## MasVnrSmall MasVnr:TotRmsAbvGrd -1.773e-02  5.106e-03  -3.473 0.000524
## MasVnrLarge MasVnr:NgroupGroup 2 -2.326e-01  3.921e-02  -5.932 3.41e-09
## MasVnrNo MasVnr:NgroupGroup 2     9.791e-03  2.083e-02   0.470 0.638299
## MasVnrSmall MasVnr:NgroupGroup 2 -2.272e-02  6.323e-02  -0.359 0.719360
```

```
## MasVnrLarge MasVnr:NgroupGroup 3 -2.469e-01  2.630e-02  -9.388  < 2e-16
## MasVnrNo MasVnr:NgroupGroup 3    -2.465e-02  9.161e-03  -2.691 0.007176
## MasVnrSmall MasVnr:NgroupGroup 3 -2.690e-02  1.641e-02  -1.639 0.101268
## MasVnrLarge MasVnr:NgroupGroup 4  1.312e-01  1.525e-02   8.604  < 2e-16
## MasVnrNo MasVnr:NgroupGroup 4     3.473e-02  1.768e-02   1.965 0.049570
## MasVnrSmall MasVnr:NgroupGroup 4  8.829e-03  1.603e-02   0.551 0.581827
## MasVnrLarge MasVnr:GarageCars1   -1.935e-01  7.445e-02  -2.599 0.009409
## MasVnrNo MasVnr:GarageCars1       1.169e-01  1.259e-02   9.291  < 2e-16
## MasVnrSmall MasVnr:GarageCars1    1.027e-02  3.992e-02   0.257 0.796979
## MasVnrLarge MasVnr:GarageCars2   -2.149e-01  7.358e-02  -2.921 0.003522
## MasVnrNo MasVnr:GarageCars2       1.481e-01  1.326e-02  11.170  < 2e-16
## MasVnrSmall MasVnr:GarageCars2    1.461e-02  4.091e-02   0.357 0.720974
## MasVnrLarge MasVnr:GarageCars3   -1.345e-01  7.607e-02  -1.768 0.077143
## MasVnrNo MasVnr:GarageCars3       2.098e-01  2.153e-02   9.745  < 2e-16
## MasVnrSmall MasVnr:GarageCars3    8.352e-02  4.556e-02   1.833 0.066921
## MasVnrLarge MasVnr:GarageCars4   -3.361e-02  9.520e-02  -0.353 0.724050
## MasVnrNo MasVnr:GarageCars4       3.656e-01  5.168e-02   7.075 1.94e-12
## MasVnrSmall MasVnr:GarageCars4    1.402e-01  6.634e-02   2.113 0.034663
##
## (Intercept)                      ***
## FullBath1
## FullBath2
## FullBath3
## FullBath4
## Fireplaces1                      ***
## Fireplaces2                      ***
## Fireplaces3
## Fireplaces4
## FoundationCBlock                 ***
## FoundationPConc                  ***
## FoundationSlab                   ***
## FoundationStone
## FoundationWood
## MasVnrLarge MasVnr:tLotArea      ***
## MasVnrNo MasVnr:tLotArea         ***
## MasVnrSmall MasVnr:tLotArea      ***
## MasVnrLarge MasVnr:OverallQual   ***
## MasVnrNo MasVnr:OverallQual      ***
## MasVnrSmall MasVnr:OverallQual   ***
## MasVnrLarge MasVnr:TotalBsmtSF
## MasVnrNo MasVnr:TotalBsmtSF      ***
## MasVnrSmall MasVnr:TotalBsmtSF   ***
## MasVnrLarge MasVnr:Age1
## MasVnrNo MasVnr:Age1             ***
## MasVnrSmall MasVnr:Age1          ***
## MasVnrLarge MasVnr:Age2          ***
## MasVnrNo MasVnr:Age2             ***
## MasVnrSmall MasVnr:Age2          *
## MasVnrLarge MasVnr:tGrLivArea    ***
## MasVnrNo MasVnr:tGrLivArea       ***
## MasVnrSmall MasVnr:tGrLivArea    ***
## MasVnrLarge MasVnr:TotRmsAbvGrd  *
## MasVnrNo MasVnr:TotRmsAbvGrd     .
## MasVnrSmall MasVnr:TotRmsAbvGrd  ***
```

```
## MasVnrLarge MasVnr:NgroupGroup 2 ***
## MasVnrNo MasVnr:NgroupGroup 2
## MasVnrSmall MasVnr:NgroupGroup 2
## MasVnrLarge MasVnr:NgroupGroup 3 ***
## MasVnrNo MasVnr:NgroupGroup 3    **
## MasVnrSmall MasVnr:NgroupGroup 3
## MasVnrLarge MasVnr:NgroupGroup 4 ***
## MasVnrNo MasVnr:NgroupGroup 4    *
## MasVnrSmall MasVnr:NgroupGroup 4
## MasVnrLarge MasVnr:GarageCars1   **
## MasVnrNo MasVnr:GarageCars1      ***
## MasVnrSmall MasVnr:GarageCars1
## MasVnrLarge MasVnr:GarageCars2   **
## MasVnrNo MasVnr:GarageCars2      ***
## MasVnrSmall MasVnr:GarageCars2
## MasVnrLarge MasVnr:GarageCars3   .
## MasVnrNo MasVnr:GarageCars3      ***
## MasVnrSmall MasVnr:GarageCars3   .
## MasVnrLarge MasVnr:GarageCars4
## MasVnrNo MasVnr:GarageCars4      ***
## MasVnrSmall MasVnr:GarageCars4   *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1195 on 2444 degrees of freedom
## Multiple R-squared:  0.909,  Adjusted R-squared:  0.907
## F-statistic:   444 on 55 and 2444 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(model3)
```




```r
check <- lm(tSalePrice ~ tLotArea + OverallQual + TotalBsmtSF + Age1 + Age2 + tGrLivArea +
              FullBath + TotRmsAbvGrd + Fireplaces + Ngroup + Foundation + GarageCars +
              MasVnr, data = train)
vif(check)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## tLotArea     1.966472  1        1.402309
## OverallQual  3.524755  1        1.877433
## TotalBsmtSF  2.134215  1        1.460895
## Age1         5.045820  1        2.246290
## Age2         2.094033  1        1.447077
## tGrLivArea   5.253185  1        2.291983
## FullBath     3.588080  4        1.173161
## TotRmsAbvGrd 3.286231  1        1.812797
## Fireplaces   1.721764  4        1.070278
## Ngroup       4.795043  3        1.298570
## Foundation   4.812399  5        1.170136
## GarageCars   4.045045  4        1.190873
## MasVnr       1.642264  2        1.132037
```

In this final step, each predictor is taken out once and added as an interaction term with the other predictors. Finally, the optimal interaction term is chosen as `MasVnr`: the (valid) model gives the highest adjusted R^2 with `MasVnr` interacting with the other predictors, excluding those variables with NA's appearing in the summary of model (here NA shows serious multicollinearity problem). Then we get the final model with adjusted R^2 0.907, as mentioned at the very beginning. **Evidence for the validity of the final model:** Residuals vs Fitted plot and Scale Location plot both give an almost flat line which means that there are no problems with linearity, independence, or constant variance of errors. The Residuals vs Leverage plot shows one bad leverage point (#1308) which acceptable given the large data set. Although there are some points deviating from the straight line in Normal QQ plot, this is still acceptable given the large data set. Moreover, there is no multicollinearity problem since the vif's are all very small (smaller than 5).

Results

```
test <- read.csv("HTestW19Final No Y values.csv")

test$Age1 <- 2019 - test$YearBuilt
test$Age2 <- 2019 - test$YearRemodAdd

test$TotalBsmtSF[which(is.na(test$TotalBsmtSF))] <-
  median(na.omit(test$TotalBsmtSF[which(test$TotalBsmtSF != 0)]))

tLotArea <- log(test$LotArea)
tGrLivArea <- log(test$GrLivArea)

test$GarageCars <- as.factor(test$GarageCars)
test$Fireplaces <- as.factor(test$Fireplaces)
test$FullBath <- as.factor(test$FullBath)

test$MasVnrArea[which(is.na(test$MasVnrArea))] <-
  median(na.omit(test$MasVnrArea[which(test$MasVnrArea != 0)]))
med.mas.vnr <- median(test$MasVnrArea[which(test$MasVnrArea != 0)])
for(i in 1:nrow(test)){
  if(test$MasVnrArea[i] == 0) {test$MasVnr[i] <- "No MasVnr"}
  if(test$MasVnrArea[i] != 0 & test$MasVnrArea[i] <= med.mas.vnr) {test$MasVnr[i] <- "Small MasVnr"}
  if(test$MasVnrArea[i] != 0 & test$MasVnrArea[i] > med.mas.vnr) {test$MasVnr[i] <- "Large MasVnr"}
}
test$MasVnr <- as.factor(test$MasVnr)


Ntest <- as.integer(test$Neighborhood)
for(i in 1:nrow(test)){
   if(Ntest[i] %in% c(1,5,6,7,9,12,13,17,20,21)){
     test$Ngroup[i] <- "Group 1"
   }else if(Ntest[i] %in% c(2,3,11,15)){
     test$Ngroup[i] <- "Group 2"
   }else if(Ntest[i] %in% c(4,8,10,18,19,23)){
     test$Ngroup[i] <- "Group 3"
   }else{
     test$Ngroup[i] <- "Group 4"
   }
}
test$Ngroup <- as.factor(test$Ngroup)


p <- predict(model3, newdata = test)
p[is.na(p)] <- median(na.omit(p))
price <- exp(p)
my_prediction <- data.frame(Ob = 1:1500, SalePrice = round(price,2))
write.csv(my_prediction, "SalePrice_Hao_Ma_Lec1.csv")
```

```r
head(round(price,2))
```

```
##        1        2        3        4        5        6
## 192305.6 160718.6 145048.8 146714.9 115212.7 231639.7
```

```r
summary(round(price,2))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   40670  129732  168950  183875  219132  578157
```

For the testing data set, work is done to create the same new variables as in the training data as well as performing same transformation and modifications, in order to apply the model to testing data set. The predictions of house prices for the testing dataset using the final model built in this study are written into a csv file. The first several entries and the summary statistics of the predictions are shown above.

Discussion

The CI, t-test are still reasonable and the estimates of slopes are still unbiased even if the Normal QQ plot does not look very good, regarding our large data set; variable selecting methods based on adjusted R^2, AIC, AICc, and BIC are not used in this study; there should be a more advanced method for finding the best interaction term to save time.

Limitations and Conclusions

As mentioned above, the Normal QQ plot does not look very good, therefore PI's might be questionable. Also, there is for sure a bad leverage point based on Residuals vs Leverage plot. Some more advanced analysis or models are needed to improve the predictions. Overall, we can conclude that the multiple linear regression model is valid and acceptable; the predictions and inference based on the model are valid and acceptable; the minor problems are also acceptable.

References

Training, testing data sets and data descriptions from Kaggle: https://www.kaggle.com/c/stat101ahouseprice/data