Linear Regression Analysis: Regression Case Study

Neerja Doshi, Sri Santhosh Hari, Ker-Yu Ong, Nicha Ruchirawat

Part 0: Exploratory Data Analysis

```
rawDF <- read.csv("/Users/booranium/usf/601_regression/project/housing.txt",
    stringsAsFactors = T)
# rawDF <-
# read.csv('/Users/santhoshhari/Documents/Coursework/LinearRegression/IowaHousing/Data/housing.txt',
# stringsAsFactors = T)
```

Structure of Data:

The Iowa housing dataset contains 1460 rows and 81 variables, a glimpse of which is as follows:

```
str(rawDF)
```

```
## 'data.frame':    1460 obs. of  81 variables:
##  $ Id           : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ MSSubClass   : int  60 20 60 70 60 50 20 60 50 190 ...
##  $ MSZoning     : Factor w/ 5 levels "C (all)","FV",..: 4 4 4 4 4 4 4 4 5 4 ...
##  $ LotFrontage  : int  65 80 68 60 84 85 75 NA 51 50 ...
##  $ LotArea      : int  8450 9600 11250 9550 14260 14115 10084 10382 6120 7420 ...
##  $ Street       : Factor w/ 2 levels "Grvl","Pave": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Alley        : Factor w/ 2 levels "Grvl","Pave": NA NA NA NA NA NA NA NA NA NA ...
##  $ LotShape     : Factor w/ 4 levels "IR1","IR2","IR3",..: 4 4 1 1 1 1 4 1 4 4 ...
##  $ LandContour  : Factor w/ 4 levels "Bnk","HLS","Low",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ Utilities    : Factor w/ 2 levels "AllPub","NoSeWa": 1 1 1 1 1 1 1 1 1 1 ...
##  $ LotConfig    : Factor w/ 5 levels "Corner","CulDSac",..: 5 3 5 1 3 5 5 1 5 1 ...
##  $ LandSlope    : Factor w/ 3 levels "Gtl","Mod","Sev": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Neighborhood : Factor w/ 25 levels "Blmngtn","Blueste",..: 6 25 6 7 14 12 21 17 18 4 ...
##  $ Condition1   : Factor w/ 9 levels "Artery","Feedr",..: 3 2 3 3 3 3 3 5 1 1 ...
##  $ Condition2   : Factor w/ 8 levels "Artery","Feedr",..: 3 3 3 3 3 3 3 3 3 1 ...
##  $ BldgType     : Factor w/ 5 levels "1Fam","2fmCon",..: 1 1 1 1 1 1 1 1 1 2 ...
##  $ HouseStyle   : Factor w/ 8 levels "1.5Fin","1.5Unf",..: 6 3 6 6 6 1 3 6 1 2 ...
##  $ OverallQual  : int  7 6 7 7 8 5 8 7 7 5 ...
##  $ OverallCond  : int  5 8 5 5 5 5 5 6 5 6 ...
##  $ YearBuilt    : int  2003 1976 2001 1915 2000 1993 2004 1973 1931 1939 ...
##  $ YearRemodAdd : int  2003 1976 2002 1970 2000 1995 2005 1973 1950 1950 ...
##  $ RoofStyle    : Factor w/ 6 levels "Flat","Gable",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ RoofMatl     : Factor w/ 8 levels "ClyTile","CompShg",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ Exterior1st  : Factor w/ 15 levels "AsbShng","AsphShn",..: 13 9 13 14 13 13 13 7 4 9 ...
##  $ Exterior2nd  : Factor w/ 16 levels "AsbShng","AsphShn",..: 14 9 14 16 14 14 14 7 16 9 ...
##  $ MasVnrType   : Factor w/ 4 levels "BrkCmn","BrkFace",..: 2 3 2 3 2 3 4 4 3 3 ...
##  $ MasVnrArea   : int  196 0 162 0 350 0 186 240 0 0 ...
##  $ ExterQual    : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 4 3 4 3 4 3 4 4 4 ...
##  $ ExterCond    : Factor w/ 5 levels "Ex","Fa","Gd",..: 5 5 5 5 5 5 5 5 5 5 ...
##  $ Foundation   : Factor w/ 6 levels "BrkTil","CBlock",..: 3 2 3 1 3 6 3 2 1 1 ...
##  $ BsmtQual     : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 3 3 4 3 3 1 3 4 4 ...
##  $ BsmtCond     : Factor w/ 4 levels "Fa","Gd","Po",..: 4 4 4 2 4 4 4 4 4 4 ...
```

```
##  $ BsmtExposure : Factor w/ 4 levels "Av","Gd","Mn",..: 4 2 3 4 1 4 1 3 4 4 ...
##  $ BsmtFinType1 : Factor w/ 6 levels "ALQ","BLQ","GLQ",..: 3 1 3 1 3 3 3 1 6 3 ...
##  $ BsmtFinSF1   : int  706 978 486 216 655 732 1369 859 0 851 ...
##  $ BsmtFinType2 : Factor w/ 6 levels "ALQ","BLQ","GLQ",..: 6 6 6 6 6 6 6 2 6 6 ...
##  $ BsmtFinSF2   : int  0 0 0 0 0 0 0 32 0 0 ...
##  $ BsmtUnfSF    : int  150 284 434 540 490 64 317 216 952 140 ...
##  $ TotalBsmtSF  : int  856 1262 920 756 1145 796 1686 1107 952 991 ...
##  $ Heating      : Factor w/ 6 levels "Floor","GasA",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ HeatingQC    : Factor w/ 5 levels "Ex","Fa","Gd",..: 1 1 1 3 1 1 1 1 3 1 ...
##  $ CentralAir   : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Electrical   : Factor w/ 5 levels "FuseA","FuseF",..: 5 5 5 5 5 5 5 5 2 5 ...
##  $ X1stFlrSF    : int  856 1262 920 961 1145 796 1694 1107 1022 1077 ...
##  $ X2ndFlrSF    : int  854 0 866 756 1053 566 0 983 752 0 ...
##  $ LowQualFinSF : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ GrLivArea    : int  1710 1262 1786 1717 2198 1362 1694 2090 1774 1077 ...
##  $ BsmtFullBath : int  1 0 1 1 1 1 1 1 0 1 ...
##  $ BsmtHalfBath : int  0 1 0 0 0 0 0 0 0 0 ...
##  $ FullBath     : int  2 2 2 1 2 1 2 2 2 1 ...
##  $ HalfBath     : int  1 0 1 0 1 1 0 1 0 0 ...
##  $ BedroomAbvGr : int  3 3 3 3 4 1 3 3 2 2 ...
##  $ KitchenAbvGr : int  1 1 1 1 1 1 1 1 2 2 ...
##  $ KitchenQual  : Factor w/ 4 levels "Ex","Fa","Gd",..: 3 4 3 3 3 4 3 4 4 4 ...
##  $ TotRmsAbvGrd : int  8 6 6 7 9 5 7 7 8 5 ...
##  $ Functional   : Factor w/ 7 levels "Maj1","Maj2",..: 7 7 7 7 7 7 7 7 3 7 ...
##  $ Fireplaces   : int  0 1 1 1 1 0 1 2 2 2 ...
##  $ FireplaceQu  : Factor w/ 5 levels "Ex","Fa","Gd",..: NA 5 5 3 5 NA 3 5 5 5 ...
##  $ GarageType   : Factor w/ 6 levels "2Types","Attchd",..: 2 2 2 6 2 2 2 2 6 2 ...
##  $ GarageYrBlt  : int  2003 1976 2001 1998 2000 1993 2004 1973 1931 1939 ...
##  $ GarageFinish : Factor w/ 3 levels "Fin","RFn","Unf": 2 2 2 3 2 3 2 2 3 2 ...
##  $ GarageCars   : int  2 2 2 3 3 2 2 2 2 1 ...
##  $ GarageArea   : int  548 460 608 642 836 480 636 484 468 205 ...
##  $ GarageQual   : Factor w/ 5 levels "Ex","Fa","Gd",..: 5 5 5 5 5 5 5 5 2 3 ...
##  $ GarageCond   : Factor w/ 5 levels "Ex","Fa","Gd",..: 5 5 5 5 5 5 5 5 5 5 ...
##  $ PavedDrive   : Factor w/ 3 levels "N","P","Y": 3 3 3 3 3 3 3 3 3 3 ...
##  $ WoodDeckSF   : int  0 298 0 0 192 40 255 235 90 0 ...
##  $ OpenPorchSF  : int  61 0 42 35 84 30 57 204 0 4 ...
##  $ EnclosedPorch: int  0 0 0 272 0 0 0 228 205 0 ...
##  $ X3SsnPorch   : int  0 0 0 0 0 320 0 0 0 0 ...
##  $ ScreenPorch  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PoolArea     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PoolQC       : Factor w/ 3 levels "Ex","Fa","Gd": NA NA NA NA NA NA NA NA NA NA ...
##  $ Fence        : Factor w/ 4 levels "GdPrv","GdWo",..: NA NA NA NA NA 3 NA NA NA NA ...
##  $ MiscFeature  : Factor w/ 4 levels "Gar2","Othr",..: NA NA NA NA NA 3 NA 3 NA NA ...
##  $ MiscVal      : int  0 0 0 0 0 700 0 350 0 0 ...
##  $ MoSold       : int  2 5 9 2 12 10 8 11 4 1 ...
##  $ YrSold       : int  2008 2007 2008 2006 2008 2009 2007 2009 2008 2008 ...
##  $ SaleType     : Factor w/ 9 levels "COD","Con","ConLD",..: 9 9 9 9 9 9 9 9 9 9 ...
##  $ SaleCondition: Factor w/ 6 levels "Abnorml","AdjLand",..: 5 5 5 1 5 5 5 5 1 5 ...
##  $ SalePrice    : int  208500 181500 223500 140000 250000 143000 307000 200000 129900 118000 ...
```

At first glance, we see that most of the variables are categorical - both numeric and character types - and only a handful are continuous. The response variable for our analysis is `SalePrice`, and the remaining 79 variables (excluding the record `ID` column) are considered potential predictor variables. Checking the data dictionary, we found the following distribution for the predictor variables:

- 49 categorical
- 19 are continuous, e.g. area, price
- 11 are discrete, e.g. count, year

There are 0 duplicate rows in the dataset.

Handling `NA` Values

Below, we compute that number and percentage of `NA`s per variable in the dataset having at least 1 `NA`.

```
NA_columns <- colnames(rawDF)[unique(which(is.na(rawDF), arr.ind = T)[,
    2])]

NA_count <- rawDF %>% select(NA_columns) %>% summarise_all(funs(sum(is.na(.)))) %>%
    gather(key = "Variable", value = "num_na", everything()) %>% arrange(desc(num_na))

NA_count %<>% mutate(perc_na = paste(round(num_na/nrow(rawDF), 4) *
    100, "%"))
colnames(NA_count) <- c("**Variable**", "**Number of NA**", "**Percentage of NA**")
row.names(NA_count) <- NULL
knitr::kable(NA_count, caption = "\\label{tab:NACount} Variable NA Count and Percentage",
    format.args = list(big.mark = ","))
```

Table 1: Variable NA Count and Percentage

Variable	**Number of NA**	**Percentage of NA**
PoolQC	1,453	99.52 %
MiscFeature	1,406	96.3 %
Alley	1,369	93.77 %
Fence	1,179	80.75 %
FireplaceQu	690	47.26 %
LotFrontage	259	17.74 %
GarageType	81	5.55 %
GarageYrBlt	81	5.55 %
GarageFinish	81	5.55 %
GarageQual	81	5.55 %
GarageCond	81	5.55 %
BsmtExposure	38	2.6 %
BsmtFinType2	38	2.6 %
BsmtQual	37	2.53 %
BsmtCond	37	2.53 %
BsmtFinType1	37	2.53 %
MasVnrType	8	0.55 %
MasVnrArea	8	0.55 %
Electrical	1	0.07 %

The data dictionary tells us that for most of the fields in Table 1, `NA` is actually meaningful, indicating non-applicability or a lack of the feature rather than missing data. After checking the data dictionary for the meaning of each field, we imputed - for every categorical variable for which `NA` was meaningful - `NA`s with `0`s.

```
# Create a copy of rawDF to be our working data frame
housingDF <- rawDF
```

```r
# Update NAs with 0s for applicable fields
levels(housingDF$PoolQC) <- c("0", levels(housingDF$PoolQC))
housingDF$PoolQC[is.na(housingDF$PoolQC)] <- "0"
levels(housingDF$MiscFeature) <- c("0", levels(housingDF$MiscFeature))
housingDF$MiscFeature[is.na(housingDF$MiscFeature)] <- "0"
levels(housingDF$Alley) <- c("0", levels(housingDF$Alley))
housingDF$Alley[is.na(housingDF$Alley)] <- "0"
levels(housingDF$Fence) <- c("0", levels(housingDF$Fence))
housingDF$Fence[is.na(housingDF$Fence)] <- "0"
levels(housingDF$FireplaceQu) <- c("0", levels(housingDF$FireplaceQu))
housingDF$FireplaceQu[is.na(housingDF$FireplaceQu)] <- "0"
levels(housingDF$GarageType) <- c("0", levels(housingDF$GarageType))
housingDF$GarageType[is.na(housingDF$GarageType)] <- "0"
levels(housingDF$GarageFinish) <- c("0", levels(housingDF$GarageFinish))
housingDF$GarageFinish[is.na(housingDF$GarageFinish)] <- "0"
levels(housingDF$GarageQual) <- c("0", levels(housingDF$GarageQual))
housingDF$GarageQual[is.na(housingDF$GarageQual)] <- "0"
levels(housingDF$GarageCond) <- c("0", levels(housingDF$GarageCond))
housingDF$GarageCond[is.na(housingDF$GarageCond)] <- "0"
levels(housingDF$BsmtExposure) <- c("0", levels(housingDF$BsmtExposure))
housingDF$BsmtExposure[is.na(housingDF$BsmtExposure)] <- "0"
levels(housingDF$BsmtFinType2) <- c("0", levels(housingDF$BsmtFinType2))
housingDF$BsmtFinType2[is.na(housingDF$BsmtFinType2)] <- "0"
levels(housingDF$BsmtQual) <- c("0", levels(housingDF$BsmtQual))
housingDF$BsmtQual[is.na(housingDF$BsmtQual)] <- "0"
levels(housingDF$BsmtCond) <- c("0", levels(housingDF$BsmtCond))
housingDF$BsmtCond[is.na(housingDF$BsmtCond)] <- "0"
levels(housingDF$BsmtFinType1) <- c("0", levels(housingDF$BsmtFinType1))
housingDF$BsmtFinType1[is.na(housingDF$BsmtFinType1)] <- "0"
```

We then re-check the count and percentage of `NA`s per variable left in the dataset.

```r
NA_columns <- colnames(housingDF)[unique(which(is.na(housingDF), arr.ind = T)[,
    2])]
NA_count <- housingDF %>% select(NA_columns) %>% summarise_all(funs(sum(is.na(.)))) %>%
    gather(key = "Variable", value = "num_na", everything()) %>% arrange(desc(num_na))

NA_count %<>% mutate(perc_na = paste(round(num_na/nrow(housingDF),
    4) * 100, "%"))
colnames(NA_count) <- c("**Variable**", "**Number of NA**", "**Percentage of NA**")
row.names(NA_count) <- NULL
knitr::kable(NA_count, caption = "\\label{tab:NACount1} Variable NA Count and Percentage(after replacing
    format.args = list(big.mark = ","))
```

Table 2: Variable NA Count and Percentage(after replacing NAs with 0s, where appropriate)

Variable	**Number of NA**	**Percentage of NA**
LotFrontage	259	17.74 %
GarageYrBlt	81	5.55 %
MasVnrType	8	0.55 %
MasVnrArea	8	0.55 %
Electrical	1	0.07 %

Table 2 shows the list of remaining variables where `NA` indicates missing data. We impute `NA`s in these variables with

- mean of the data, for continuous variables (LotFrontage)
- median of the data, for discrete variables (GarageYrBlt)
- mode of the data, for categorical variables (MasVnrType, Electrical)

```r
# Function to get mode of data
getmode <- function(v) {
    uniqv <- unique(v)
    uniqv[which.max(tabulate(match(v, uniqv)))]
}

# Impute NAs
housingDF$LotFrontage[is.na(housingDF$LotFrontage)] <- mean(housingDF$LotFrontage,
    na.rm = T)
housingDF$GarageYrBlt[is.na(housingDF$GarageYrBlt)] <- median(housingDF$GarageYrBlt,
    na.rm = T)
housingDF$MasVnrType[is.na(housingDF$MasVnrType)] <- getmode(housingDF$MasVnrType)
housingDF$MasVnrArea[is.na(housingDF$MasVnrArea)] <- 0
housingDF$Electrical[is.na(housingDF$Electrical)] <- getmode(housingDF$Electrical)
```

Since Masonry veneer area (`MasVnrArea`) is directly related to `MasVnrType`, we impute for area based on the mode of `MasVnrType`, which is None. Our cleaned dataset is named `housingDF`.

Data Visualization

With our clean dataset, we perform exploratory data visualization of the distribution of key measures such as volume and sale price of houses by what we hypothesize to be key predictor variables.

To begin with, we check the distribution of sale prices using a histogram and box-plot.

```r
# hist(housingDF$SalePrice, main = 'Histogram of Sale Price')
# boxplot(housingDF$SalePricem, main = 'Boxplot of Sale Price')
summary(housingDF$SalePrice)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   34900  129975  163000  180921  214000  755000
```

Intuition suggests the neighborhood is a key determining factor in a house's sale price, hence below, we plot the distribution of sale price by neighborhood.

```r
housingDF %>% select(Neighborhood, SalePrice) %>% ggplot(aes(factor(Neighborhood),
    SalePrice)) + geom_boxplot() + theme(axis.text.x = element_text(angle = 90,
    hjust = 1)) + xlab("Neighborhoods")
```

From Figure 1, we can observe that Brookside and Meadow Vista have the lowest median house price while Northridge and Northridge Height have the heighest median house price as well as several outliers.

We then distribution of houses by a number of key features we hypothesize to be important in determining housing price: the property's zoning class (`MSZoning`), type of road access to the property (`Street`), type of alley access to the property (`Alley`), and type of utilies available (`Utilities`)

```r
plotHist <- function(data_in, i) {
    data <- data.frame(x = data_in[[i]])
    p <- ggplot(data = data, aes(x = factor(x))) + stat_count() +
        xlab(colnames(data_in)[i]) + theme_light() + theme(axis.text.x = element_text(angle = 90,
        hjust = 1))
```

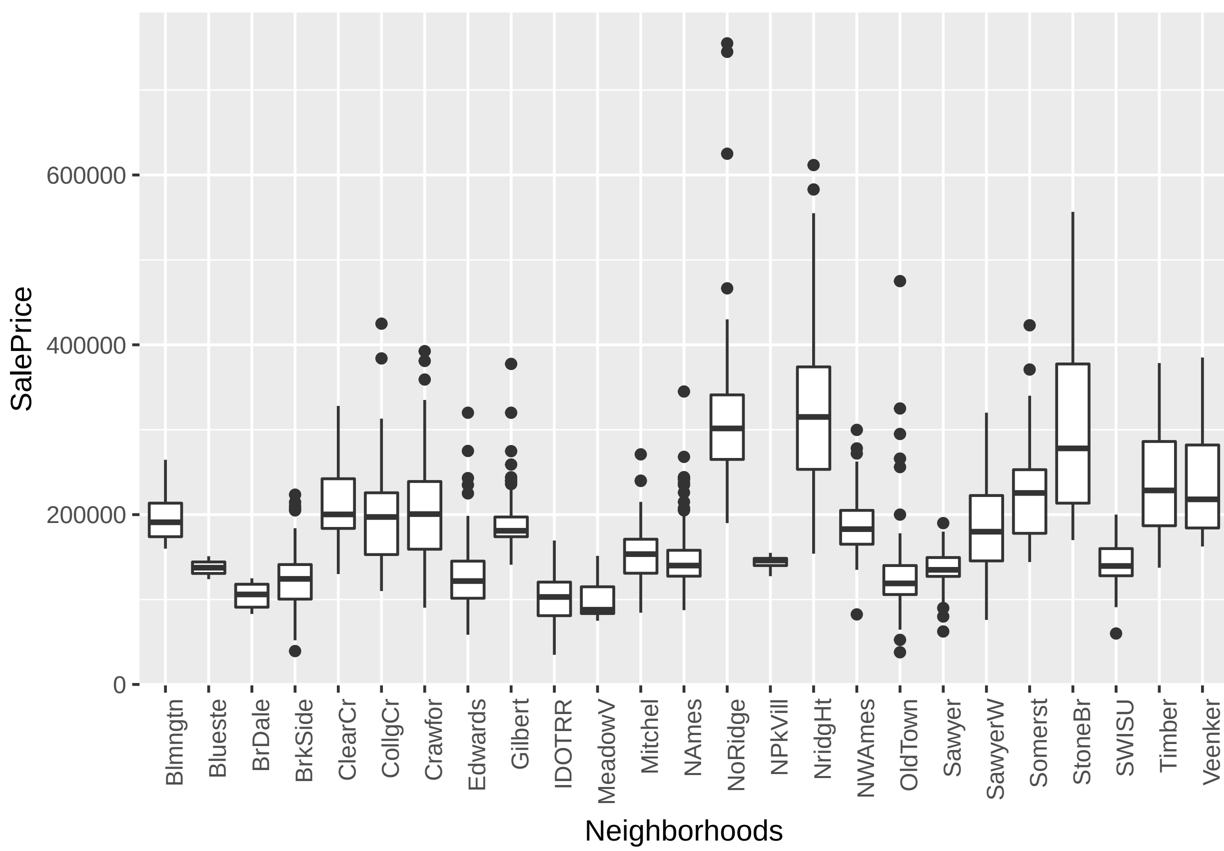


Figure 1: SalePrice distribution per neighborhood

```r
    return(p)
}

doPlots <- function(data_in, fun, ii, ncol = 3) {
    pp <- list()
    for (i in ii) {
        p <- fun(data_in = data_in, i = i)
        pp <- c(pp, list(p))
    }
    do.call("grid.arrange", c(pp, ncol = ncol))
}

plotDen <- function(data_in, i) {
    data <- data.frame(x = data_in[[i]], SalePrice = data_in$SalePrice)
    p <- ggplot(data = data) + geom_line(aes(x = x), stat = "density",
        size = 1, alpha = 1) + xlab(paste0((colnames(data_in)[i]),
        "\n", "Skewness: ", round(skewness(data_in[[i]], na.rm = TRUE),
            2))) + theme_light()
    return(p)
}

plotCorr <- function(data_in, i) {
    data <- data.frame(x = data_in[[i]], SalePrice = data_in$SalePrice)
    p <- ggplot(data, aes(x = x, y = SalePrice)) + geom_point(na.rm = TRUE) +
        geom_smooth(method = lm) + xlab(paste0(colnames(data_in)[i],
        "\n", "R-Squared: ", round(cor(data_in[[i]], data$SalePrice,
            use = "complete.obs"), 2))) + theme_light()
    return(suppressWarnings(p))
}
```

```r
doPlots(housingDF, fun = plotHist, ii = c(3, 6, 7, 10), ncol = 2)
```

Figure 2 suggests that most of the houses are located in `Medium/Low Density` residential areas. We can also observe that most of the houses have paved road access, do not have alleys and have all public utilities(E,G,W,& S). From Figure **??**{fig:hist2}, we can notice that most of the properties are regular or slightly irregular in share, built on level surfaces with gentle slope.

We also plot the distribution of houses against a number of features related to the physical geography of the property:

```r
doPlots(housingDF, fun = plotHist, ii = c(8, 9, 11, 12), ncol = 2)
```

We see that …

```r
housingDF %>% select(LandSlope, Neighborhood) %>% arrange(Neighborhood) %>%
    group_by(Neighborhood, LandSlope) %>% summarize(Count = n()) %>%
    ggplot(aes(Neighborhood, Count)) + geom_bar(aes(fill = LandSlope),
    position = "dodge", stat = "identity") + theme(axis.text.x = element_text(angle = 90,
    hjust = 1))
```

From Figure **??**{fig:hist3}, we can see that houses with severe slope are located only in Clear Creek and Timberland while more than 10 neighborhoods have properties with moderate slope.

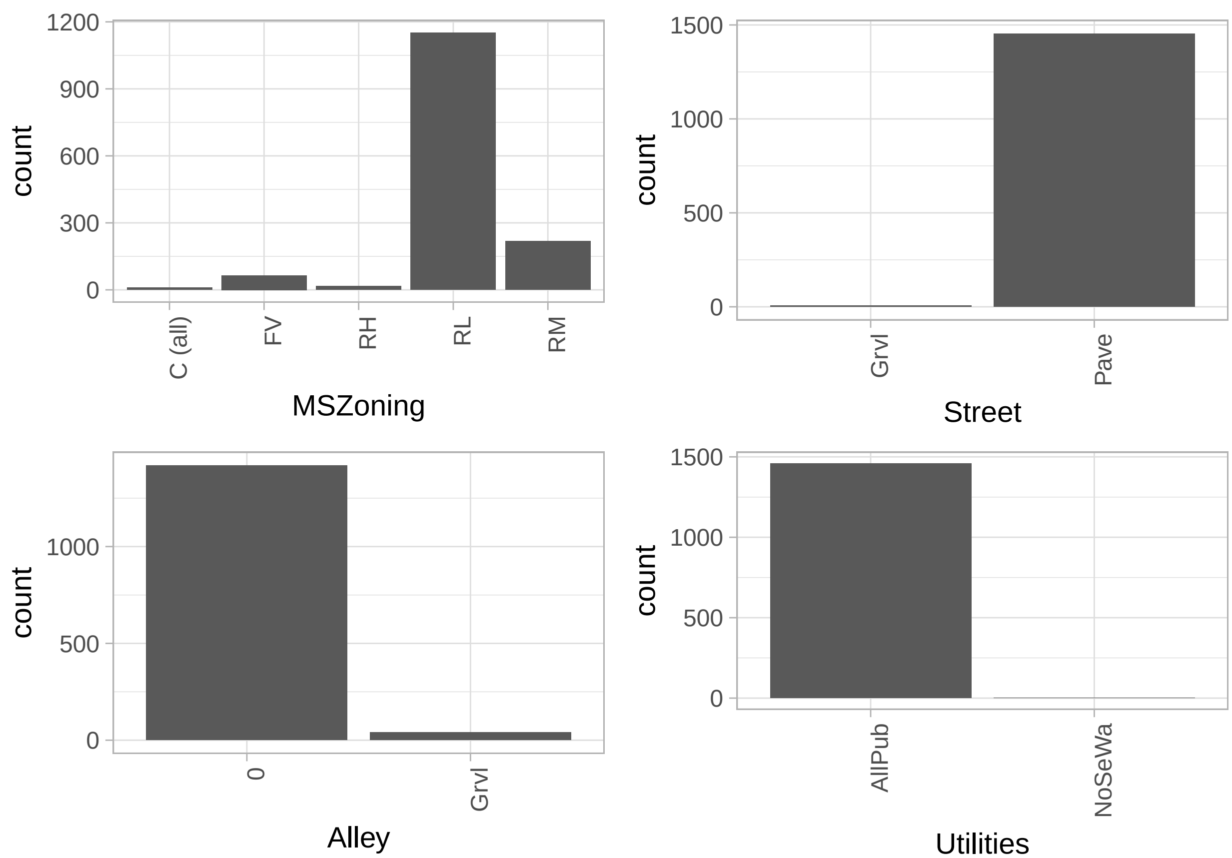


Figure 2: Locality, access, utility features distribution

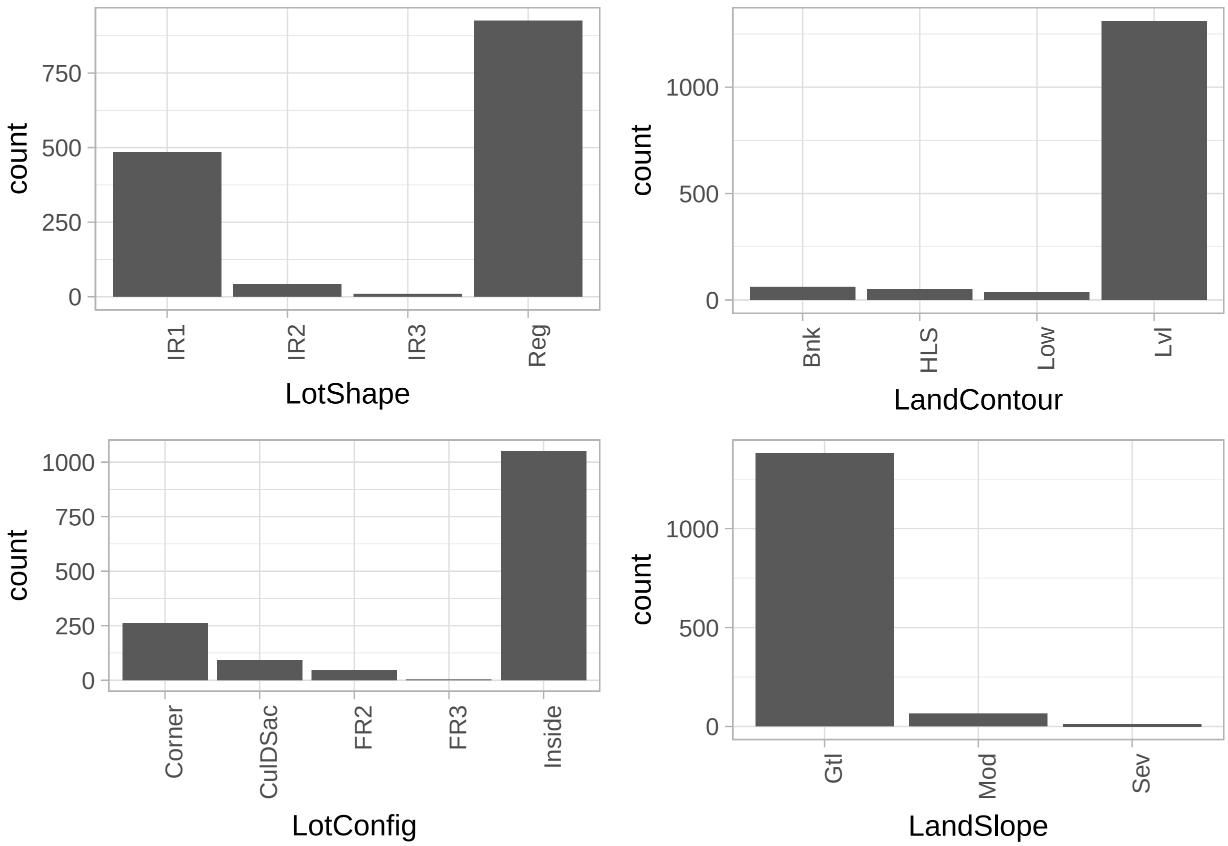


Figure 3: Lot/Land feature distribution

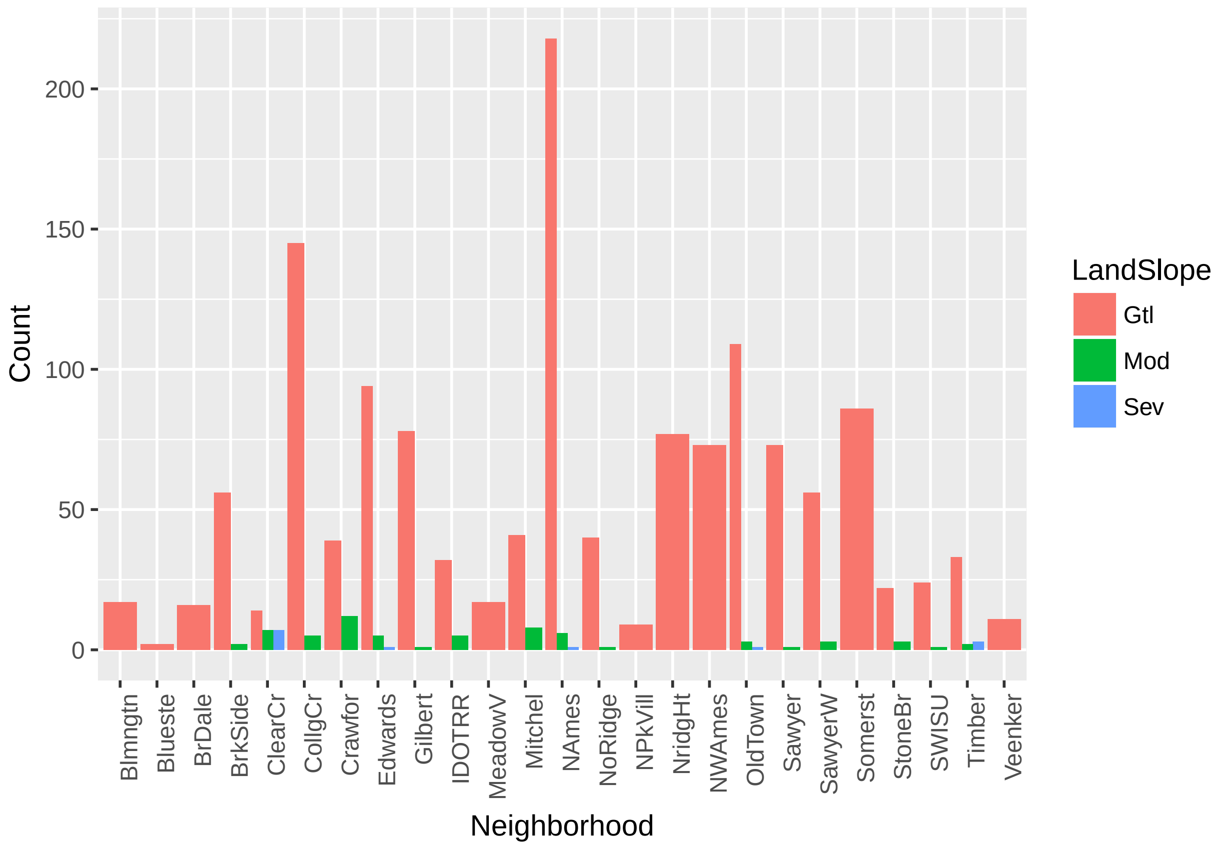


Figure 4: Neighborhood level slope distribution

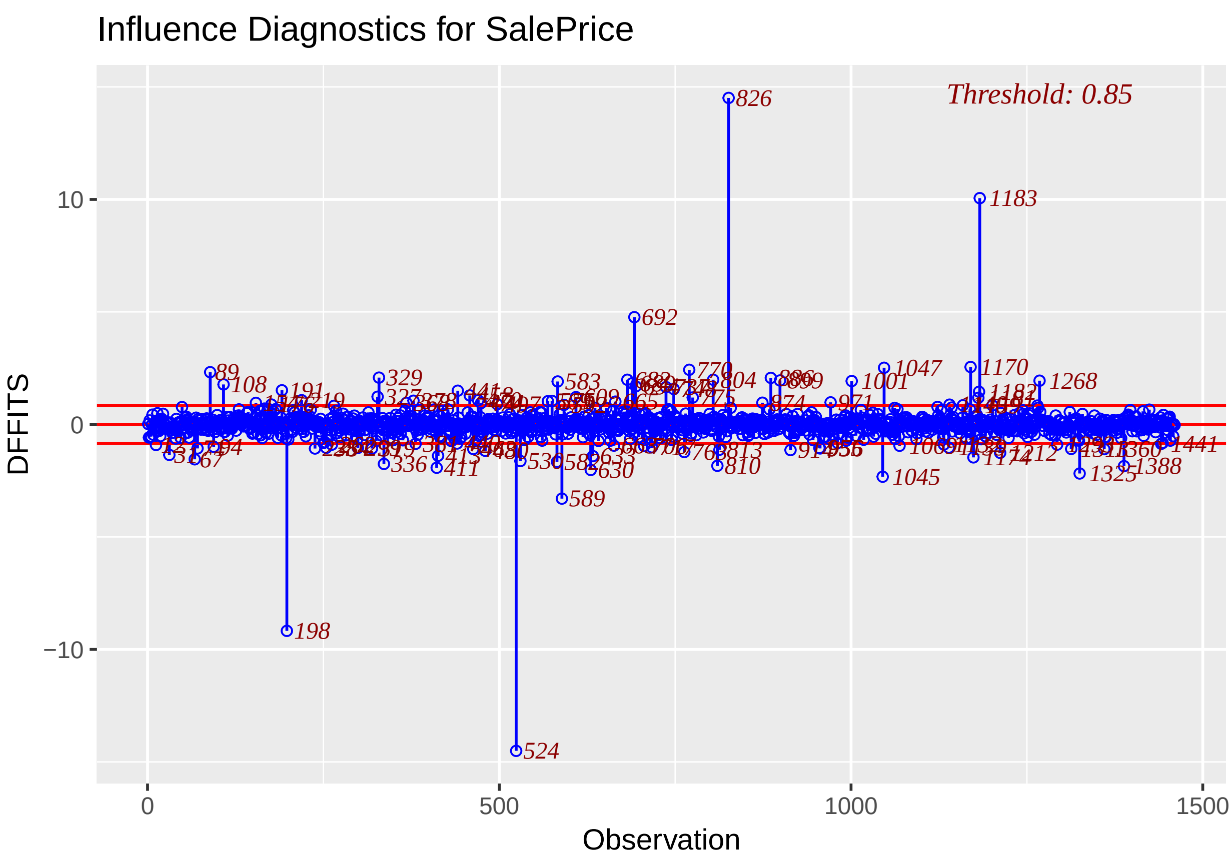
Testing for Influential Points

Having dealt with the NAs in our dataset, we use the `model.matrix()` function from the `glmnet` package to convert each categorical variable into an appropriate set of binary indicators: for a categorical variable that takes k levels, `model.matrix()` produces k-1 binary indicators. We then reappend our response vector `SalePrice` to the resulting wide design matrix `designDF` to create `workingDF`, which includes both the converted predictors and response variables.

```
designDF <- model.matrix(SalePrice ~ ., data = housingDF)[, -1]
designDF <- as.data.frame(designDF)
workingDF <- cbind(designDF, SalePrice = housingDF$SalePrice)
```

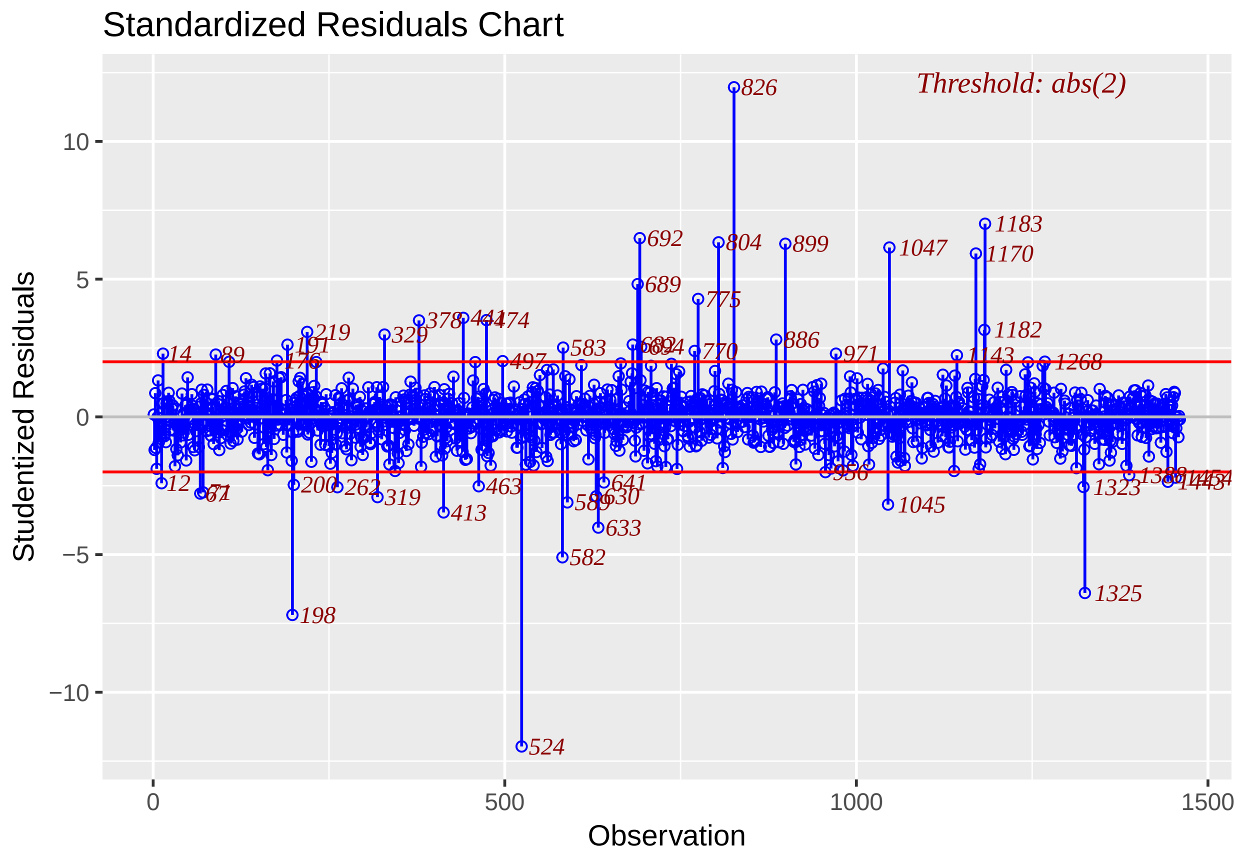
In looking for influential points, we leverage the `OLSRR` package to test observations for influence according to both the DFFITS and studentized residuals criterions. We do this by first fitting a saturated model on `workingDF` and then calling `ols_dffits_plot()` and `ols_srsd_plot()` to plot the DFFITS and studentized residuals plots respectively.

```
ols_model <- lm(SalePrice ~ ., data = workingDF)
ols_dffits_plot(ols_model)
```



The DFFITS plot shows 5 influential observations - 198, 418, 524, 826 and 1183 - with observations 524 and 826 being the most outstanding. This is corroborated by the studentized residuals plot, as shown below:

```
ols_srsd_chart(ols_model)
```



As a remedial measure, we investigate data points 524 and 826 per the original dataset:

```
filter(workingDF, Id == 524 | Id == 826)
```

```
##     Id MSSubClass MSZoningFV MSZoningRH MSZoningRL MSZoningRM LotFrontage
## 1 524          60          0          0          1          0         130
## 2 826          20          0          0          1          0         114
##   LotArea StreetPave AlleyGrvl AlleyPave LotShapeIR2 LotShapeIR3
## 1   40094          1         0         0           0           0
## 2   14803          1         0         0           0           0
##   LotShapeReg LandContourHLS LandContourLow LandContourLvl UtilitiesNoSeWa
## 1           0              0              0              0               0
## 2           1              0              0              1               0
##   LotConfigCulDSac LotConfigFR2 LotConfigFR3 LotConfigInside LandSlopeMod
## 1                0            0            0               1            0
## 2                0            0            0               1            0
##   LandSlopeSev NeighborhoodBlueste NeighborhoodBrDale NeighborhoodBrkSide
## 1            0                   0                  0                   0
## 2            0                   0                  0                   0
##   NeighborhoodClearCr NeighborhoodCollgCr NeighborhoodCrawfor
## 1                   0                   0                   0
## 2                   0                   0                   0
##   NeighborhoodEdwards NeighborhoodGilbert NeighborhoodIDOTRR
## 1                   1                   0                  0
## 2                   0                   0                  0
##   NeighborhoodMeadowV NeighborhoodMitchel NeighborhoodNAmes
## 1                   0                   0                 0
## 2                   0                   0                 0
```

```
##   NeighborhoodNoRidge NeighborhoodNPkVill NeighborhoodNridgHt
## 1                   0                   0                   1
## 2                   0                   0                   1
##   NeighborhoodNWAmes NeighborhoodOldTown NeighborhoodSawyer
## 1                  0                   0                  0
## 2                  0                   0                  0
##   NeighborhoodSawyerW NeighborhoodSomerst NeighborhoodStoneBr
## 1                   0                   0                   0
## 2                   0                   0                   0
##   NeighborhoodSWISU NeighborhoodTimber NeighborhoodVeenker Condition1Feedr
## 1                 0                  0                   0               0
## 2                 0                  0                   0               0
##   Condition1Norm Condition1PosA Condition1PosN Condition1RRAe
## 1              0              0              1              0
## 2              0              0              1              0
##   Condition1RRAn Condition1RRNe Condition1RRNn Condition2Feedr
## 1              0              0              0               0
## 2              0              0              0               0
##   Condition2Norm Condition2PosA Condition2PosN Condition2RRAe
## 1              0              0              1              0
## 2              0              0              1              0
##   Condition2RRAn Condition2RRNn BldgType2fmCon BldgTypeDuplex
## 1              0              0              0              0
## 2              0              0              0              0
##   BldgTypeTwnhs BldgTypeTwnhsE HouseStyle1.5Unf HouseStyle1Story
## 1             0              0                0                0
## 2             0              0                0                1
##   HouseStyle2.5Fin HouseStyle2.5Unf HouseStyle2Story HouseStyleSFoyer
## 1                0                0                1                0
## 2                0                0                0                0
##   HouseStyleSLvl OverallQual OverallCond YearBuilt YearRemodAdd
## 1              0          10           5      2007         2008
## 2              0          10           5      2007         2008
##   RoofStyleGable RoofStyleGambrel RoofStyleHip RoofStyleMansard
## 1              0                0            1                0
## 2              0                0            1                0
##   RoofStyleShed RoofMatlCompShg RoofMatlMembran RoofMatlMetal RoofMatlRoll
## 1             0               1               0             0            0
## 2             0               1               0             0            0
##   RoofMatlTar&Grv RoofMatlWdShake RoofMatlWdShngl Exterior1stAsphShn
## 1               0               0               0                  0
## 2               0               0               0                  0
##   Exterior1stBrkComm Exterior1stBrkFace Exterior1stCBlock
## 1                  0                  0                 0
## 2                  0                  0                 0
##   Exterior1stCemntBd Exterior1stHdBoard Exterior1stImStucc
## 1                  1                  0                  0
## 2                  1                  0                  0
##   Exterior1stMetalSd Exterior1stPlywood Exterior1stStone Exterior1stStucco
## 1                  0                  0                0                 0
## 2                  0                  0                0                 0
##   Exterior1stVinylSd Exterior1stWd Sdng Exterior1stWdShing
## 1                  0                  0                  0
## 2                  0                  0                  0
```

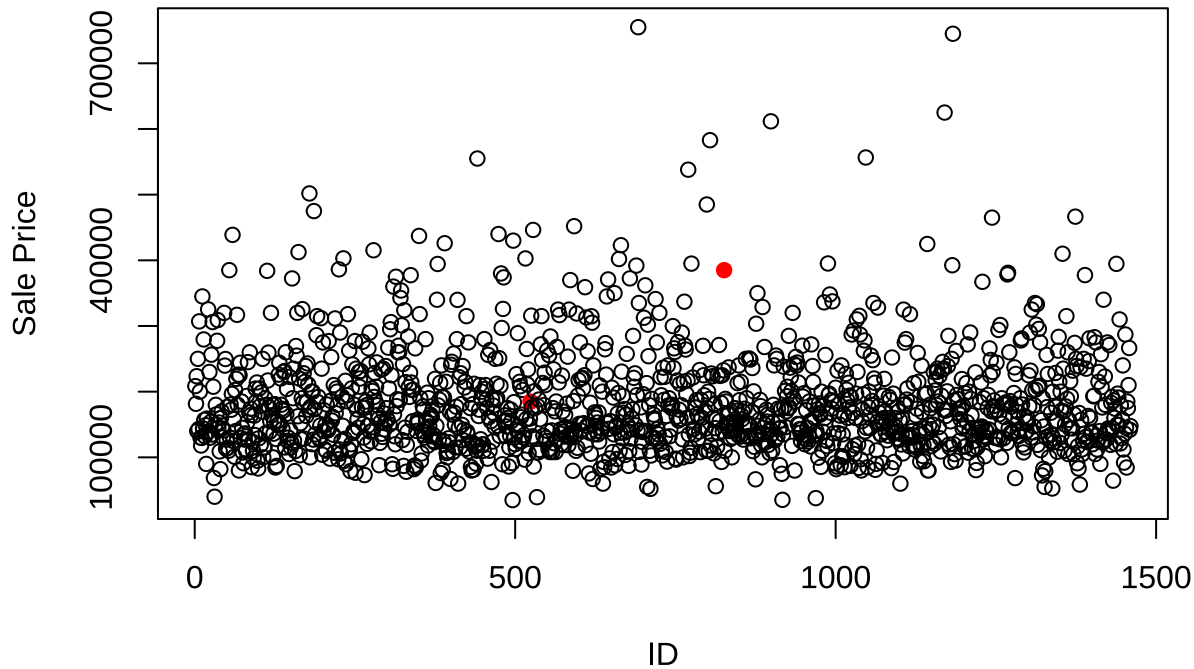
```
##   Exterior2ndAsphShn Exterior2ndBrk Cmn Exterior2ndBrkFace
## 1                  0                  0                  0
## 2                  0                  0                  0
##   Exterior2ndCBlock Exterior2ndCmentBd Exterior2ndHdBoard
## 1                 0                  1                  0
## 2                 0                  1                  0
##   Exterior2ndImStucc Exterior2ndMetalSd Exterior2ndOther
## 1                  0                  0                0
## 2                  0                  0                0
##   Exterior2ndPlywood Exterior2ndStone Exterior2ndStucco Exterior2ndVinylSd
## 1                  0                0                 0                  0
## 2                  0                0                 0                  0
##   Exterior2ndWd Sdng Exterior2ndWd Shng MasVnrTypeBrkFace MasVnrTypeNone
## 1                  0                  0                 0              0
## 2                  0                  0                 1              0
##   MasVnrTypeStone MasVnrArea ExterQualFa ExterQualGd ExterQualTA
## 1               1        762           0           0           0
## 2               0        816           0           0           0
##   ExterCondFa ExterCondGd ExterCondPo ExterCondTA FoundationCBlock
## 1           0           0           0           1                0
## 2           0           0           0           1                0
##   FoundationPConc FoundationSlab FoundationStone FoundationWood BsmtQualEx
## 1               1              0               0              0          0
## 2               1              0               0              0          0
##   BsmtQualFa BsmtQualGd BsmtQualTA BsmtCondFa BsmtCondGd BsmtCondPo
## 1          0          0          0          0          0          1
## 2          0          0          0          0          0          1
##   BsmtCondTA BsmtExposureAv BsmtExposureGd BsmtExposureMn BsmtExposureNo
## 1          0              1              0              0              0
## 2          0              0              0              0              0
##   BsmtFinType1ALQ BsmtFinType1BLQ BsmtFinType1GLQ BsmtFinType1LwQ
## 1               0               1               0               0
## 2               0               1               0               0
##   BsmtFinType1Rec BsmtFinType1Unf BsmtFinSF1 BsmtFinType2ALQ
## 1               0               0       2260               0
## 2               0               0       1636               0
##   BsmtFinType2BLQ BsmtFinType2GLQ BsmtFinType2LwQ BsmtFinType2Rec
## 1               0               0               0               1
## 2               0               0               0               1
##   BsmtFinType2Unf BsmtFinSF2 BsmtUnfSF TotalBsmtSF HeatingGasA HeatingGasW
## 1               0          0       878        3138           1           0
## 2               0          0       442        2078           1           0
##   HeatingGrav HeatingOthW HeatingWall HeatingQCFa HeatingQCGd HeatingQCPo
## 1           0           0           0           0           0           0
## 2           0           0           0           0           0           0
##   HeatingQCTA CentralAirY ElectricalFuseF ElectricalFuseP ElectricalMix
## 1           0           1               0               0             0
## 2           0           1               0               0             0
##   ElectricalSBrkr X1stFlrSF X2ndFlrSF LowQualFinSF GrLivArea BsmtFullBath
## 1               1      3138      1538            0      4676            1
## 2               1      2084         0            0      2084            1
##   BsmtHalfBath FullBath HalfBath BedroomAbvGr KitchenAbvGr KitchenQualFa
## 1            0        3        1            3            1             0
## 2            0        2        0            2            1             0
```

```
##   KitchenQualGd KitchenQualTA TotRmsAbvGrd FunctionalMaj2 FunctionalMin1
## 1             0             0           11              0              0
## 2             0             0            7              0              0
##   FunctionalMin2 FunctionalMod FunctionalSev FunctionalTyp Fireplaces
## 1              0             0             0             1          1
## 2              0             0             0             1          1
##   FireplaceQuEx FireplaceQuFa FireplaceQuGd FireplaceQuPo FireplaceQuTA
## 1             0             1             0             0             0
## 2             0             1             0             0             0
##   GarageType2Types GarageTypeAttchd GarageTypeBasment GarageTypeBuiltIn
## 1                0                0                 1                 0
## 2                1                0                 0                 0
##   GarageTypeCarPort GarageTypeDetchd GarageYrBlt GarageFinishFin
## 1                 0                0        2007               0
## 2                 0                0        2007               0
##   GarageFinishRFn GarageFinishUnf GarageCars GarageArea GarageQualEx
## 1               0               0          3        884            0
## 2               0               0          3       1220            0
##   GarageQualFa GarageQualGd GarageQualPo GarageQualTA GarageCondEx
## 1            0            0            1            0            0
## 2            0            0            1            0            0
##   GarageCondFa GarageCondGd GarageCondPo GarageCondTA PavedDriveP
## 1            0            0            1            0           0
## 2            0            0            1            0           0
##   PavedDriveY WoodDeckSF OpenPorchSF EnclosedPorch X3SsnPorch ScreenPorch
## 1           1        208         406             0          0           0
## 2           1        188          45             0          0           0
##   PoolArea PoolQCEx PoolQCFa PoolQCGd FenceGdPrv FenceGdWo FenceMnPrv
## 1        0        0        0        0          0         0          0
## 2        0        0        0        0          0         0          0
##   FenceMnWw MiscFeatureGar2 MiscFeatureOthr MiscFeatureShed
## 1         0               0               0               0
## 2         0               0               0               0
##   MiscFeatureTenC MiscVal MoSold YrSold SaleTypeCon SaleTypeConLD
## 1               0       0     10   2007           0             0
## 2               0       0      6   2008           0             0
##   SaleTypeConLI SaleTypeConLw SaleTypeCWD SaleTypeNew SaleTypeOth
## 1             0             0           0           1           0
## 2             0             0           0           1           0
##   SaleTypeWD SaleConditionAdjLand SaleConditionAlloca SaleConditionFamily
## 1          0                    0                   0                   0
## 2          0                    0                   0                   0
##   SaleConditionNormal SaleConditionPartial SalePrice
## 1                   0                    1    184750
## 2                   0                    1    385000
```

Specifically, let's look at the values they take on a number of continuous variables as compared to other observations.

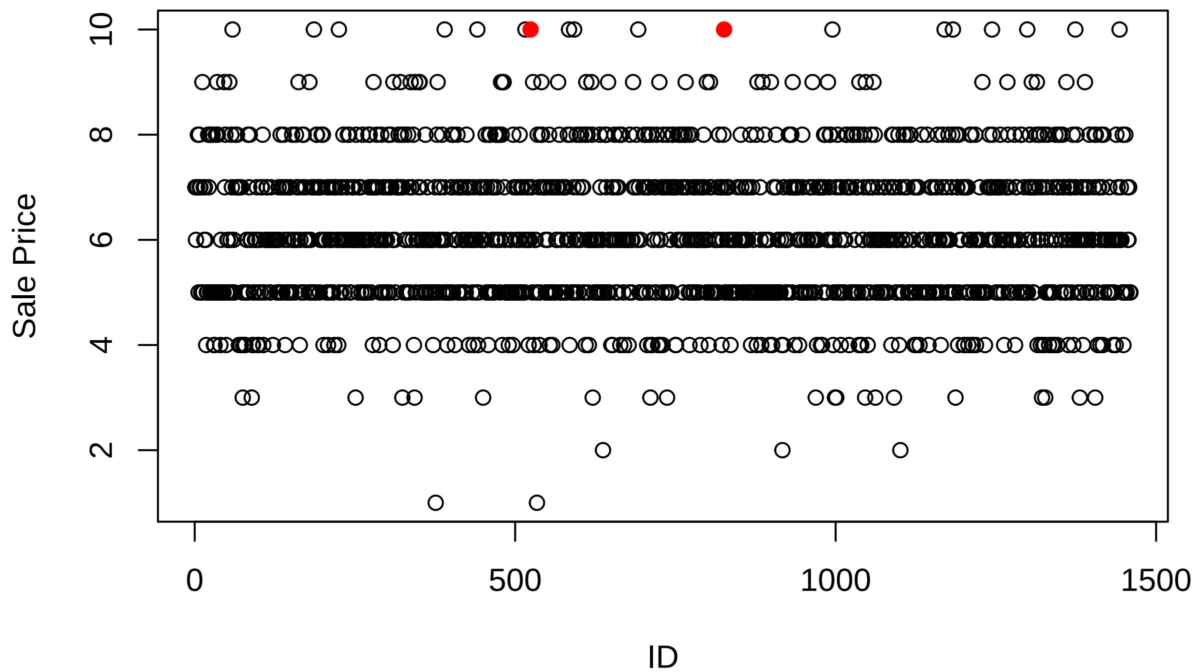
`SalePrice`:

```r
plot(workingDF$Id, workingDF$SalePrice, col = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, "red", "black"), pch = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, 19, 1), xlab = "ID", ylab = "Sale Price")
```



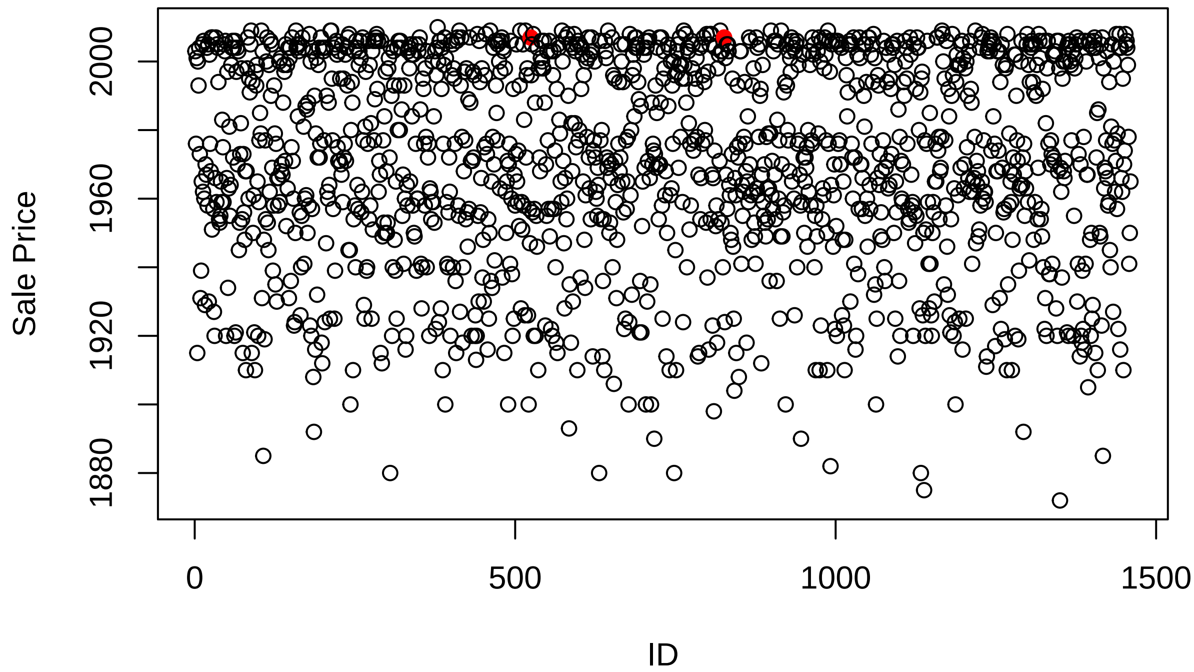
`OverallQual`: they are in the bucket of 'Excellent' overall material and finish of the house.

```
plot(workingDF$Id, workingDF$OverallQual, col = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, "red", "black"), pch = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, 19, 1), xlab = "ID", ylab = "Sale Price")
```



`YearBuilt`: they appear to be recently built

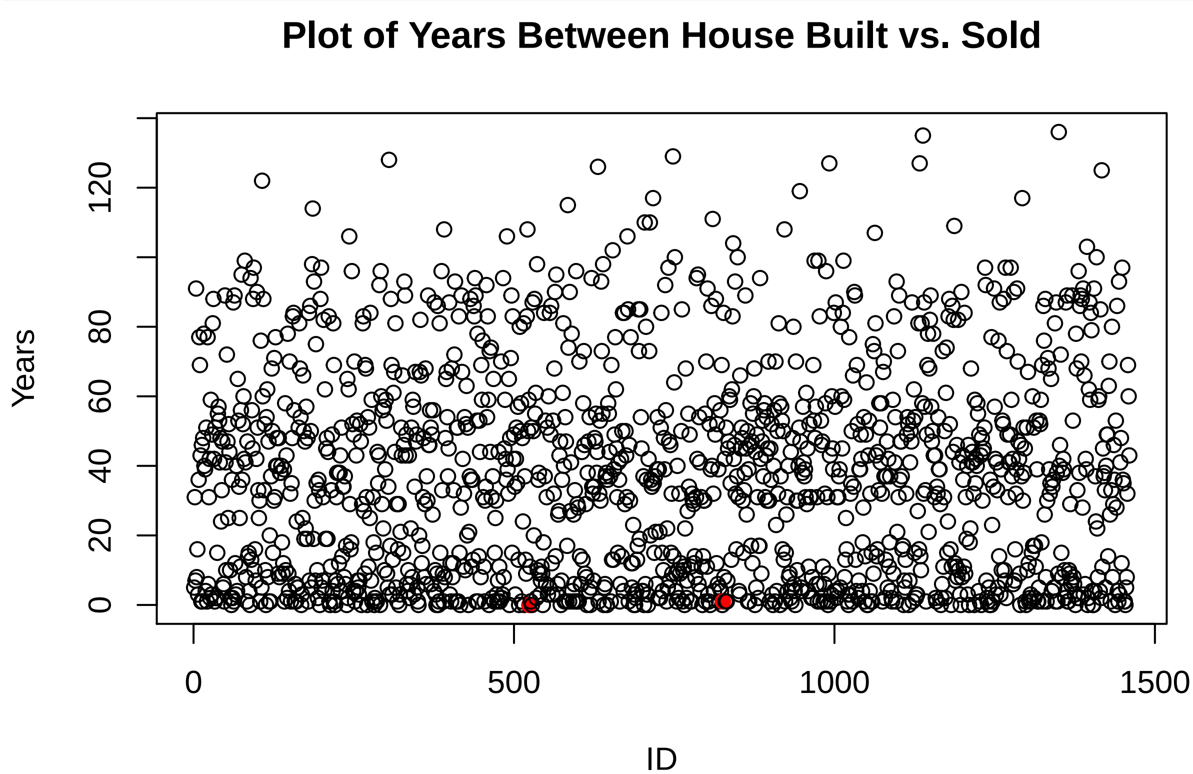
```
plot(workingDF$Id, workingDF$YearBuilt, col = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, "red", "black"), pch = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, 19, 1), xlab = "ID", ylab = "Sale Price")
```



In fact: the difference between the year built and year sold for these two houses is small compared to the same difference for other houses:

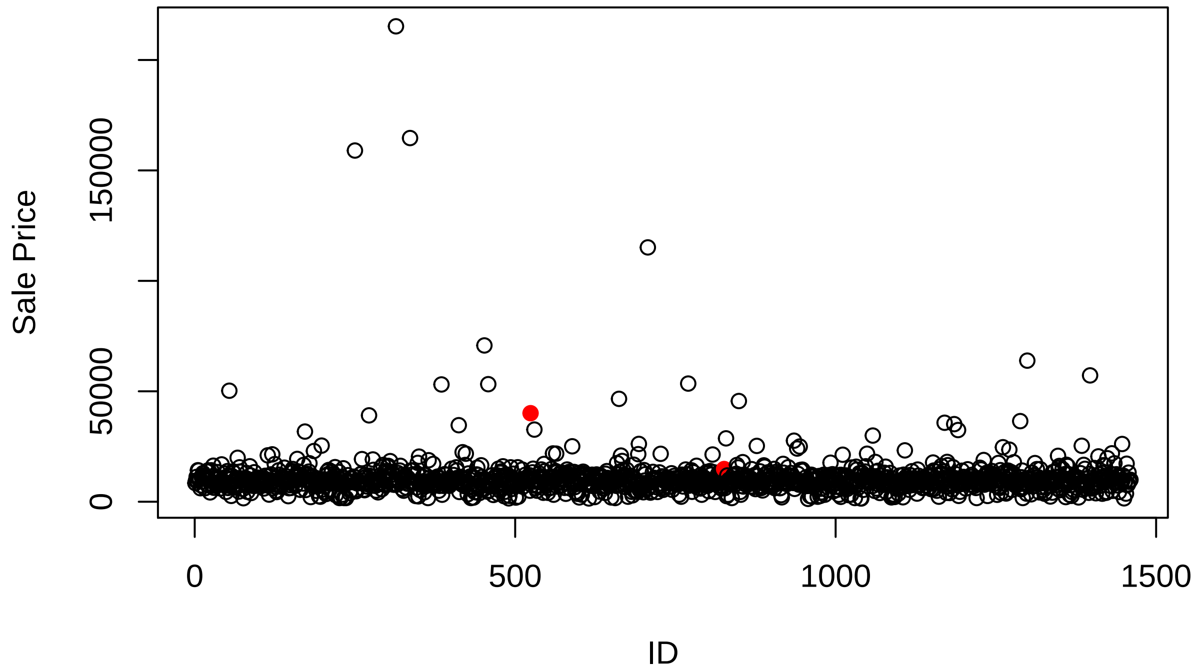
Plot of year built vs. year sold - the two observations appear to be both built and sold recently

```
plot(workingDF$Id, workingDF$YrSold - workingDF$YearBuilt, col = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, "red", "black"), pch = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, 19, 1), xlab = "ID", ylab = "Years",
    main = "Plot of Years Between House Built vs. Sold")
```



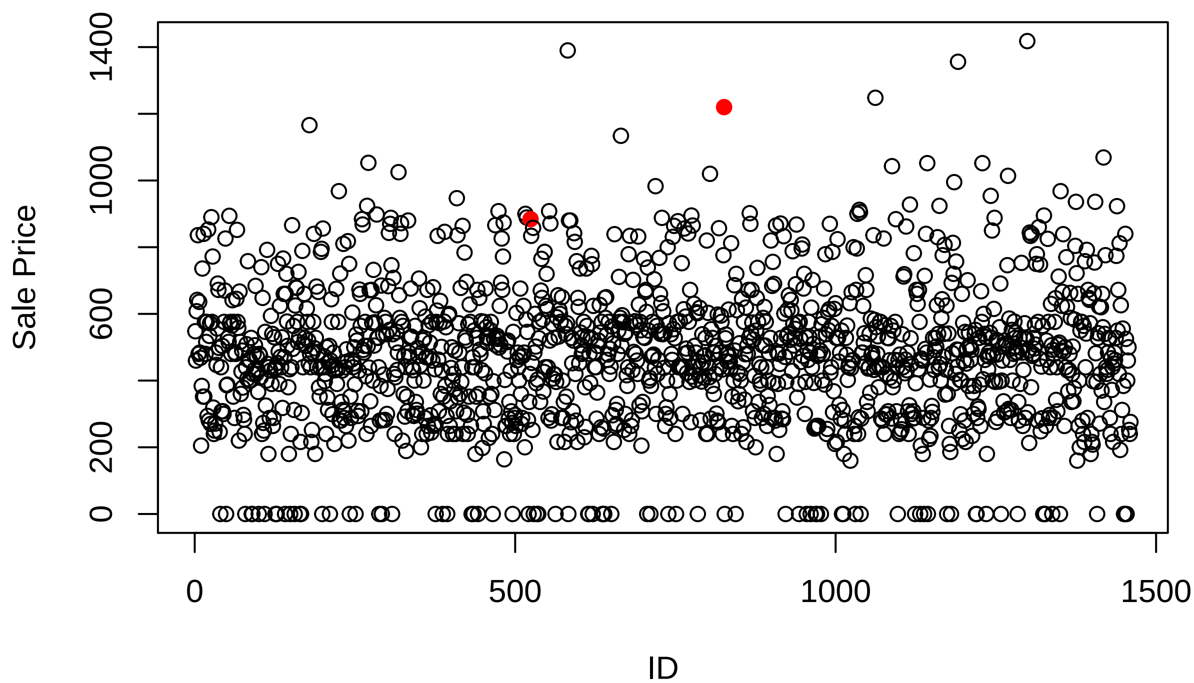
`LotArea`: Relatively large

```
plot(workingDF$Id, workingDF$LotArea, col = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, "red", "black"), pch = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, 19, 1), xlab = "ID", ylab = "Sale Price")
```

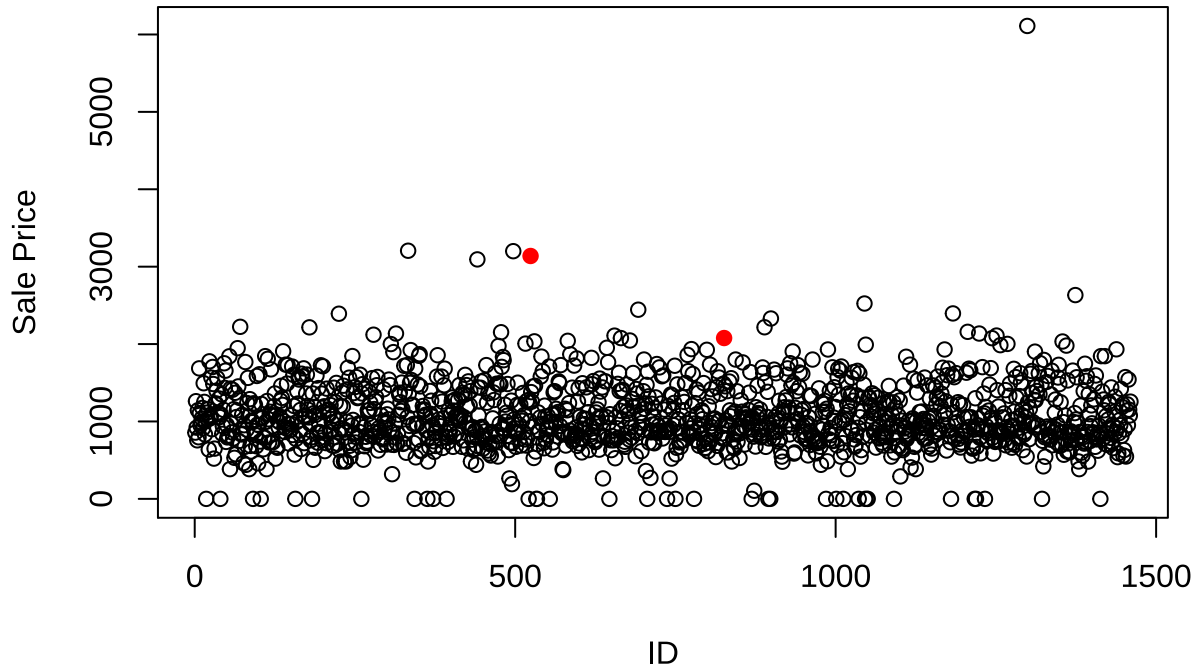


`GarageArea`: Relatively large

```
plot(workingDF$Id, workingDF$GarageArea, col = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, "red", "black"), pch = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, 19, 1), xlab = "ID", ylab = "Sale Price")
```



```r
plot(workingDF$Id, workingDF$TotalBsmtSF, col = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, "red", "black"), pch = ifelse(workingDF$Id ==
    524 | workingDF$Id == 826, 19, 1), xlab = "ID", ylab = "Sale Price")
```



We can now plot the distribution of values for the variables in our `workingDF` and spotcheck for outliers:

```r
cols <- colnames(workingDF)

for (c in cols) {
    print(c)
    data <- workingDF[[c]]
    plot(data)
}
```

Part I: Explanatory Modelling

For the purposes of variable selection, we refer to the saturated OLS model created above and perform stepwise model selection according to both AIC and BIC criterions.

```r
# Code chunk not run; load saved model for expediency
model_aic <- step(ols_model, direction = "backward", trace = F)
model_bic <- step(ols_model, k = log(nrow(workingDF)), direction = "backward",
    trace = F)

## save the models
save(model_aic, file = "/Users/booranium/usf/601_regression/project/IowaHousing/AIC_model_v2.rda")
save(model_bic, file = "/Users/booranium/usf/601_regression/project/IowaHousing/BIC_model_v2.rda")
```

Per the AIC criterion, the following are the predictor variables signficant at the alpha = 0.05 level.

```r
load("/Users/booranium/usf/601_regression/project/IowaHousing/AIC_model_v2.rda")  # model loaded as 'mo
```

```r
# Find coefficients significant at the alpha = 0.05 level
bool_aic <- summary(model_aic)$coeff[-1, 4] < 0.05
sig_var_aic <- names(bool_aic)[bool_aic == TRUE]
```

Per the BIC criterion, the following are the predictor variables signficant at the alpha = 0.05 level.

```r
load("/Users/booranium/usf/601_regression/project/IowaHousing/BIC_model.rda")  # model loaded as 'model_

# find coefficients significant at the alpha = 0.05 level
bool_bic <- summary(model_aic)$coeff[-1, 4] < 0.05
sig_var_bic <- names(bool_bic)[bool_bic == TRUE]
```

Note that that both criterions select the same set of variables.

```r
setdiff(sig_var_aic, sig_var_bic)
```

```
## character(0)
```

We can now perform OLS regression with our subset of significant variables. The model summary is as follows:

```r
model_sig_formula <- as.formula(paste("SalePrice ~ ", paste(sig_var_bic,
    collapse = "+")))
model_sig <- lm(formula = model_sig_formula, data = workingDF)
model_sig_rsq <- summary(model_sig)$r.squared  # r-squared value
```

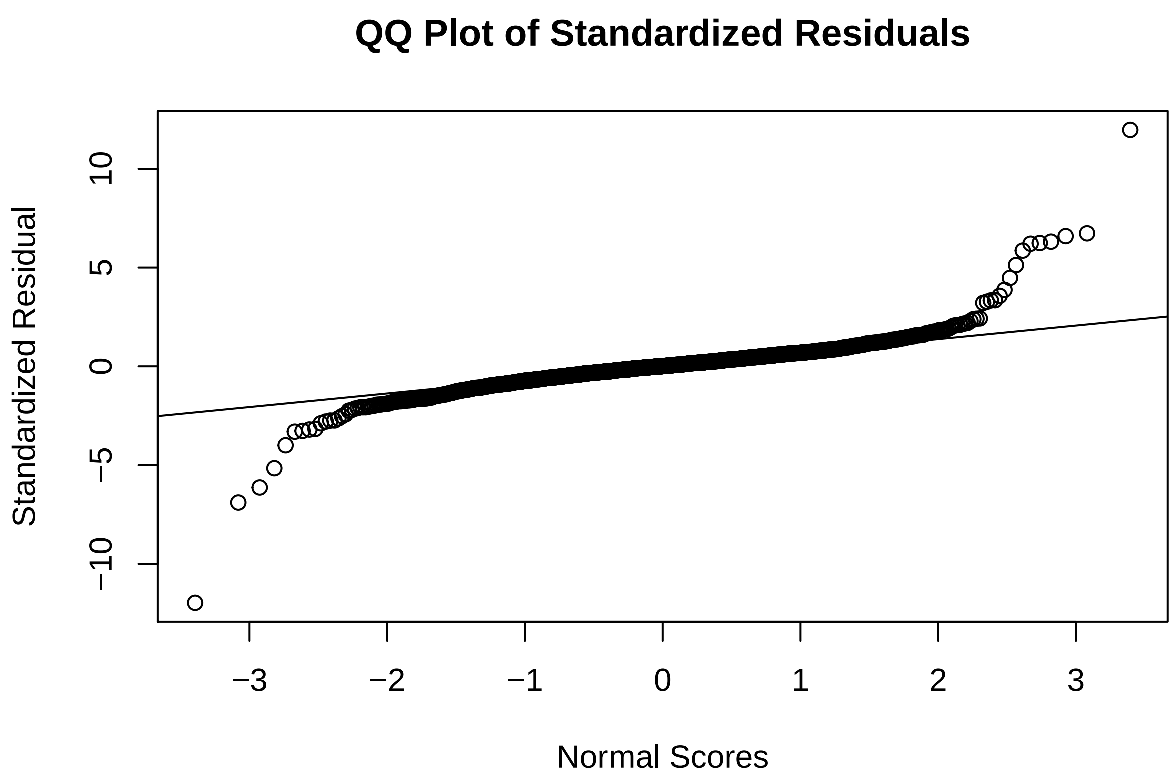
At first glance, the multiple R-squared value of 0.9252048 indicates that 91.19% of the variability in SalePrice around its mean is explained by the mode, i.e. by the predictor variables that have been included. This suggests a high-performing explanatory model. We test this conclusion by checking the residuals of the model as follows:

```r
# Residual plots

res = resid(model_sig)  # residuals
stdres = rstandard(model_sig)  # standardized residuals

# Plot of residuals vs. fitted values
plot(workingDF$SalePrice, res, main = "Plot of Residuals vs. Sale Price",
    xlab = "Sale Price", ylab = "Residual")
abline(h = c(0, 0), col = "red")
abline(v = 350000, col = "blue")
```


```r
# Normal Probability Plot of Residuals
qqnorm(stdres, main = "QQ Plot of Standardized Residuals", xlab = "Normal Scores",
    ylab = "Standardized Residual")
qqline(stdres)
```



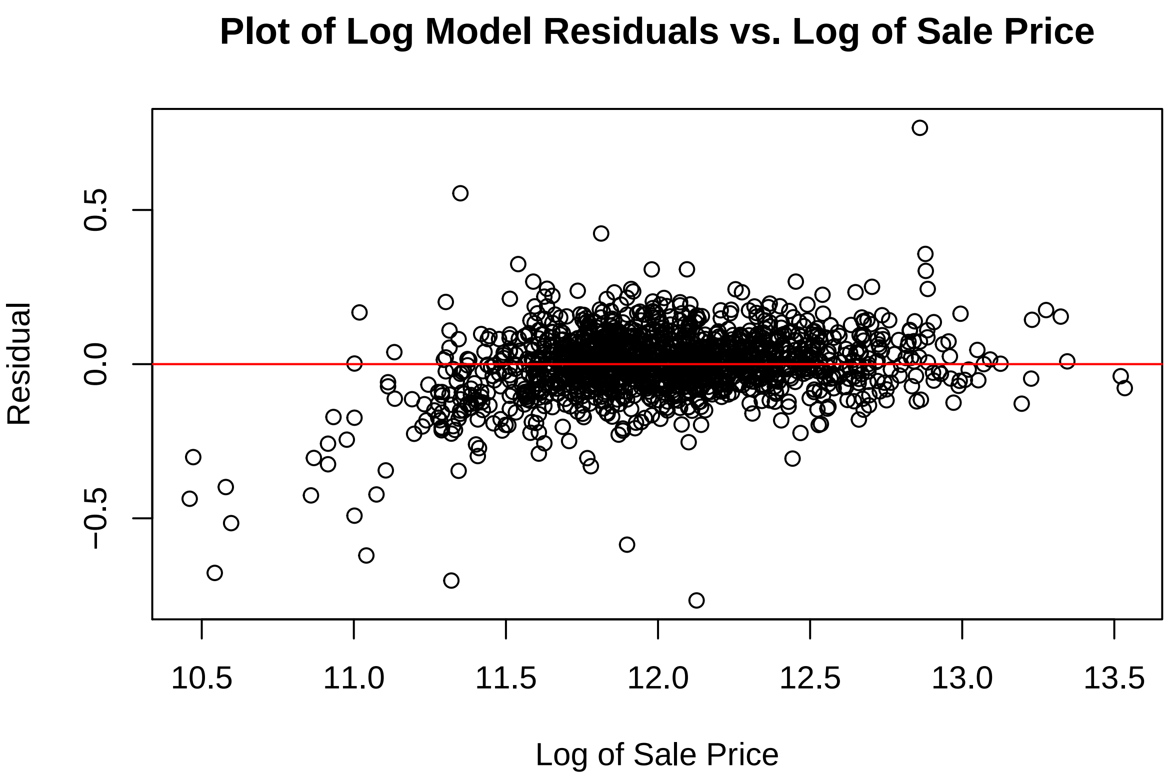
The residual plot shows that for the most part, the residuals are evenly distributed across the y = 0 line. However, we see that as Sale Price increases, the residuals start to deviate homoscedasticity. More specifically, we see this deviation happen at approximately Sale Price = $350K, which our earlier summary showed to be between the variable's 3rd quartile and maximum. This suggets that for the last quartile of high-priced houses, the fitted regression model is not as adequate as it is for the reset of the population. The normal probability (QQ) plot corroborates this finding: it shows deviance from linearity at both tail ends of the residual range.

As a remedial measure, we perform a log transformation on Sale Price, refit the model, and examine the resulting residuals.

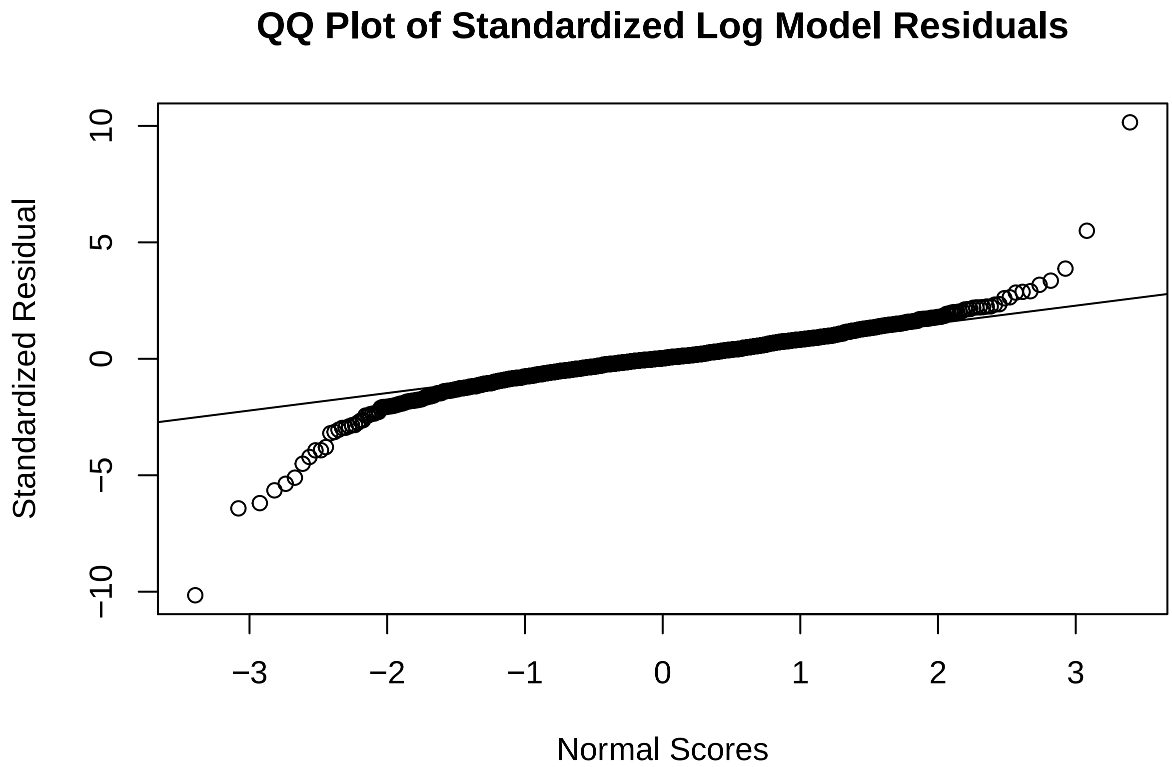
```
# Log Transformation
log_model_formula <- as.formula(paste("log(SalePrice) ~ ", paste(sig_var_bic,
    collapse = "+")))
log_model <- lm(formula = log_model_formula, data = workingDF)
log_model_rsq <- summary(log_model)$r.squared  # r-squared value

res_log = resid(log_model)  # residuals
stdres_log = rstandard(log_model)  # standardized residuals

# Plot of Residuals from Log Model vs. Fitted Values
plot(log(workingDF$SalePrice), res_log, main = "Plot of Log Model Residuals vs. Log of Sale Price",
    xlab = "Log of Sale Price", ylab = "Residual")
abline(h = c(0, 0), col = "red")
abline(v = 350000, col = "blue")
```




```
# Normal Probability Plot of Residuals from Log Model
qqnorm(stdres_log, main = "QQ Plot of Standardized Log Model Residuals",
    xlab = "Normal Scores", ylab = "Standardized Residual")
qqline(stdres_log)
```



Our new model produces an R-squared value of 0.9270108. Our plots of residuals vs. fitted values and the normal probability of the standardized residuals also show better patterns of homoscedasticity and normality respectively.

Part II: Predictive Modelling