2024

IOWA

CENTURY 21 AMES

HOME SALES MARKET ANALYSIS

Introduction

Century 21 Ames, a real estate company based in Ames, Iowa, has asked us to provide an estimate of sale price of homes based on their square footage in the North Ames, Edwards, and Brookside neighborhoods. They would like to know how the sale price of a house is related to its square footage and whether this relationship varies depending on the neighborhood the house is located in. Century 21 Ames seeks to answer the following questions: 1) Provide estimates for sale price vs. square footage 2) Determine the difference in sale price between N. Ames, Edwards, and Brookside 3) Estimate the confidence interval limits for each neighborhood.

Data Description

The data set is sourced directly from the Kaggle website and maintained in a zip file. The size of the file is approximately 957.39 kB and can be downloaded and unzipped to the user's local hard drive. The unzipped folder contains four files, three comma separated value (CSV) files and a text file. The two files, train.csv and test.csv, which contain 2919 observations total (1460 training observations and 1459 test observations), are used to train, and test our regression model. The train.csv file contains 81 columns, whereas the test.csv file contains 80 columns (excluding sale price). Additionally, a text file labeled "data_description.txt" is included to provide full descriptions of each column or variables found in the data set. A sample CSV file is also provided to assist with the formatting and submission process. We are specifically concerned with "Neighborhood", "SalePrice", "GrLivArea" columns with our initial regression analysis. (See Reference for website link).

Analysis Question 1:

What is the relationship between property sale price and square footage based on the neighborhoods North Ames, Edwards, Brookside in Ames, Iowa?

The Model

According to the request of client, we want to build a regression model whose response variable is SalePrice($) and explanatory variable is GrLIvArea(sqft) in the neighborhood of NAmes (North Ames), Edwards and Brkside(Brookside) in Ames, Iowa. Moreover, we want to utilize log-log transformation to build the regression models to meet the assumptions of linear regression. When the variable of Brkside is reference, the regression equation is:

$$\text{Pred}[\log(\textit{Sales price})] = \beta 0 + \beta 1 * \log(\textit{GrLIvArea}) + \beta 2 * \textit{NAmes} + \beta 3 * \textit{Edwards} + \beta 4 * \textit{NAmes} * \log(\textit{GrLIvArea}) + \beta 5 * \textit{Edwards} * \log(\textit{GrLIvArea})$$

Here, = 5.91, = 0.819, = 2.57, = 2.09, = -0.346 and = -0.299 (Please see details in the appendix) and the regression equation for each neighborhood is in the Fig.1.

Here, the regression equations for each neighborhood are:

$$\text{Pred}[\log(\text{Sales price\_NAmes}) = 8.48 + 0.473 * \text{GRLIvArea}$$

$$\text{Pred}[\log(\text{Sales price\_Edwards}) = 8 + 0.52 * \text{GRLIvArea}$$

$$\text{Pred}[\log(\text{Sales price\_Brkside}) = 5.91 + 0.819 * \text{GRLIvArea}$$

```
##
## Call:
## lm(formula = log_SalePrice ~ log_GrLivArea + log_GrLivArea *
##     NAmes_dummy + log_GrLivArea * Edwards_dummy, data = filtered_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.72080 -0.10353  0.02184  0.10586  0.80470
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      5.91292    0.50459  11.718  < 2e-16 ***
## log_GrLivArea                    0.81965    0.07163  11.443  < 2e-16 ***
## NAmes_dummy                      2.57981    0.59988   4.301 2.17e-05 ***
## Edwards_dummy                    2.09359    0.64589   3.241   0.0013 **
## log_GrLivArea:NAmes_dummy       -0.34662    0.08482  -4.087 5.35e-05 ***
## log_GrLivArea:Edwards_dummy -0.29998    0.09122  -3.289   0.0011 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1923 on 377 degrees of freedom
## Multiple R-squared:  0.5121, Adjusted R-squared:  0.5056
## F-statistic: 79.14 on 5 and 377 DF,  p-value: < 2.2e-16
```

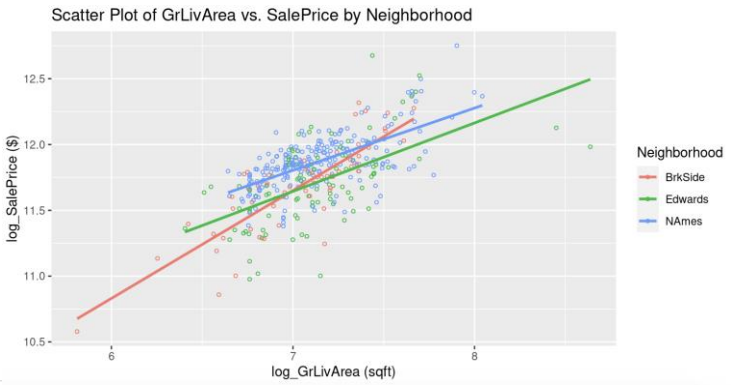


Fig. 1. Multiple linear regression model parameter estimates(left) and Scatter plot for each neighborhood (right)

Model Assumptions

In the Fig. 2. the residual plot of log-log data for the Ames housing data of three neighborhood is displayed. The residuals are well scattered randomly except some of outliers of data 251, 411 and 725.

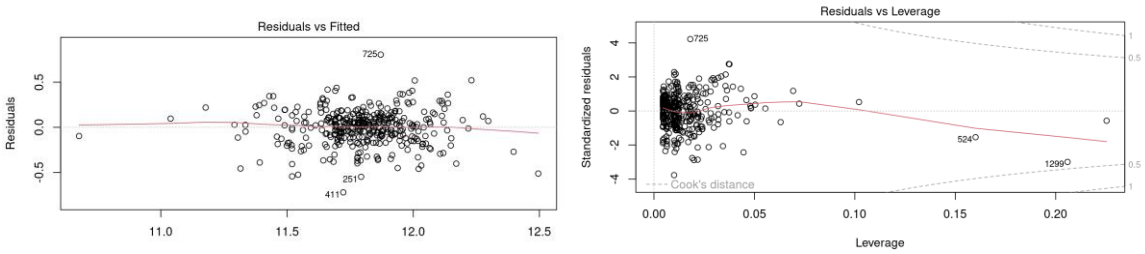


Fig. 2. Residual plot (left) and Standardized Residual vs. Leverage plot (right).

- The Standardized Residuals vs Leverage is displayed on the Fig. 3. There is relatively high leverage and low Cook's D for data 524 and 1299 also low leverage and high residuals of 725.
- Equal standard deviation: Influential point analysis for the residuals vs. leverage plot that shows the two observations possibly having high leverage and high residuals, however, does not appear to have a high influence that would affect the regression model's parameter estimates given the magnitude of the scale.

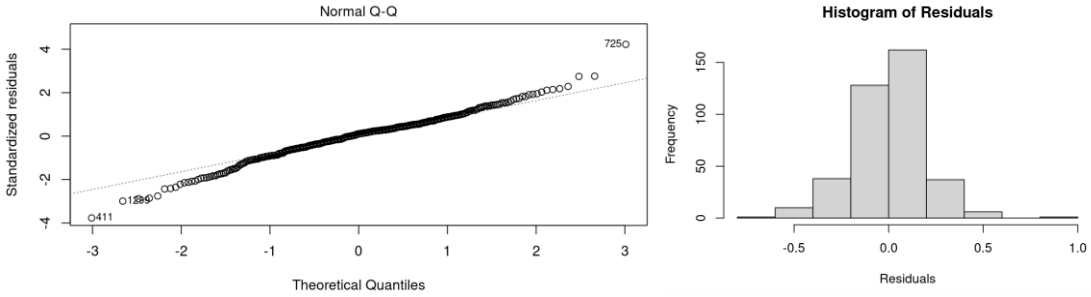


Fig. 3. QQ plot (left) and Histogram of residuals (right)

- Linearity: Visual inspection of quantile plot of residuals appears linear and supports normality assumptions for the log transformations.
- Normality: Visual check of the histogram of residuals appears to be normally distributed following a general bell-shaped curve with potential outliers identified earlier but not having much effect overall on the spread.
- Independant observations: Based on the large sample size and number of observations gathered in this study, we will assume the observations are independent.

Next, we want to compare our full model with the reduced model using the extra sum of squares test to determine if there is a difference in the sale price between North Ames and Edwards neighborhoods after accounting for square footage:

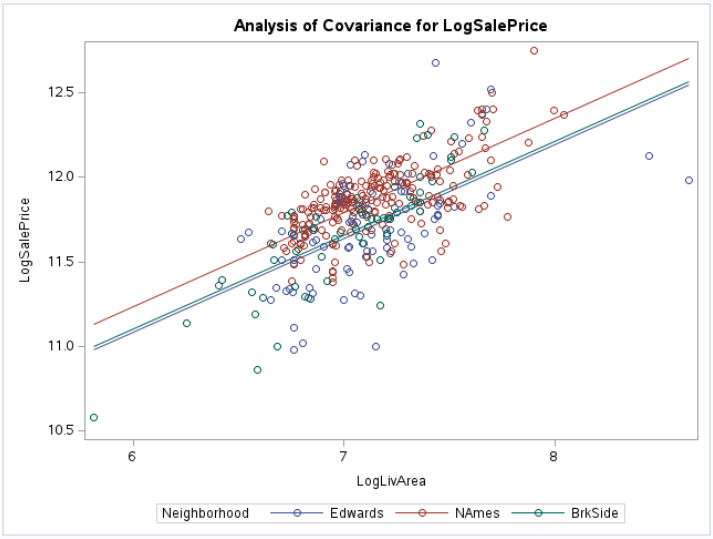


Fig. 4. Scatter plot with parallel slopes for reduced linear regression model

$$Pred[\log(Sales\ price)] = \beta 0 + \beta 1 * \log(GrLIvArea) + \beta 2 * Edwards + \beta 3 * NAmes$$

Here, β0 = 7.75338, β1= 0.56824, and the scatter plot with model is displayed on the Fig. 6.

We want to perform the lack of fit testing to compare simple linear regression model (equal mean model) and multiple linear regression model (separate mean model) for validating our model. After lack of fit testing in the Table. 1., there is strong evidence to suggest the linear regression model has a lack of fit with respect to the separate means model (P-VALUE = 0.0002133). Moreover, we can conclude that our model gives better inference about Sales Price based on GrLivArea for each neighborhood as North Ames, Edwards and Brookside.

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	14.62857557	2.92571511	79.14	<.0001
Error	377	13.93775037	0.03697016		
Corrected Total	382	28.56632594			

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	13.98906597	4.66302199	121.24	<.0001
Error	379	14.57725997	0.03846243		
Corrected Total	382	28.56632594			

Table 1. Extra Sum of Squares Test Full (left) Reduced (right) Models

Source	df	SS		MS	F	Pr > F
Model	2	0.6392		0.3196	8.6448	0.0002133
Error (SMM)	377	13.9378		0.03697		
Total (EMM)	379	14.577				

Table 2. Extra sum of squares test

After performing an extra sum of squares test, we concluded there is strong evidence at the α=0.05 level of significance to suggest that the multiple linear regression model is a better fit than the reduced model for estimating / predicting the sale price for homes in North Ames, Edwards and Brookside (p-value = 0.0002133).

Stepwise Selection Summary							
Step	Effect Entered	Effect Removed	Number Effects In	Number Parms In	Adjusted R-Square	SBC	CV PRESS
0	Intercept		1	1	0.0000	-988.2458	28.8322
1	LogLivAre*Neighborho		2	4	0.4837	-1226.6166	15.2597
2	Neighborhood		3	6	0.5056*	-1233.3570*	14.7279*
* Optimal Value of Criterion							

Stepwise Selection Summary							
Step	Effect Entered	Effect Removed	Number Effects In	Number Parms In	Adjusted R-Square	SBC	CV PRESS
0	Intercept		1	1	0.0000	-988.2458	28.8322
1	LogLivArea		2	2	0.4188	-1191.1728	16.9186
2	Neighborhood		3	4	0.4857*	-1228.0710*	15.1822*
* Optimal Value of Criterion							

Table 3. Adjusted R^2 and CV PRESS statistics on full (left) and reduced (right) model

Comparison: Adjusted R^2

Performing stepwise selection on both models yields an adjusted R^2 value of 0.5056 for the separate slope (full) model and 0.4857 for the equal slope (reduced) model. The adjusted R^2 for our model is higher than simple linear regression model and our model gives better fit than simple model.

Comparison: Internal CV Press

Performing stepwise selection on both models yields an CV Press value of 14.727 for the separate slope (full) model and 15.182 for the equal slope (reduced) model. The CV press for our model is smaller than simple linear regression model so our model provides better fit.

Parameters Estimates

```
##
## Call:
## lm(formula = log_SalePrice ~ log_GrLivArea + log_GrLivArea *
##     NAmes_dummy + log_GrLivArea * Edwards_dummy, data = filtered_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.72080 -0.10353  0.02184  0.10586  0.80470
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   5.91292    0.50459  11.718  < 2e-16 ***
## log_GrLivArea                 0.81965    0.07163  11.443  < 2e-16 ***
## NAmes_dummy                   2.57981    0.59988   4.301 2.17e-05 ***
## Edwards_dummy                 2.09359    0.64589   3.241   0.0013 **
## log_GrLivArea:NAmes_dummy    -0.34662    0.08482  -4.087 5.35e-05 ***
## log_GrLivArea:Edwards_dummy -0.29998    0.09122  -3.289   0.0011 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1923 on 377 degrees of freedom
## Multiple R-squared:  0.5121, Adjusted R-squared:  0.5056
## F-statistic: 79.14 on 5 and 377 DF,  p-value: < 2.2e-16
```

```
##                                 2.5 %     97.5 %
## (Intercept)                 4.9207572  6.9050843
## log_GrLivArea               0.6788064  0.9604897
## NAmes_dummy                 1.4002744  3.7593394
## Edwards_dummy               0.8235795  3.3635933
## log_GrLivArea:NAmes_dummy  -0.5134042 -0.1798447
## log_GrLivArea:Edwards_dummy -0.4793353 -0.1206263
```

95% confidence interval for parameter estimates. (log-log transformation)

```
##                                  2.5 %      97.5 %
## (Intercept)                137.1063869 997.3325860
## log_GrLivArea                1.9715232   2.6129757
## NAmes_dummy                  4.0563130  42.9200629
## Edwards_dummy                2.2786418  28.8928265
## log_GrLivArea:NAmes_dummy    0.5984549   0.8353999
## log_GrLivArea:Edwards_dummy  0.6191948   0.8863651
```

95% Confidence interval for parameter estimates after performing exponentiation.

Table 4. Parameter estimates for analysis 1.

Interpretation

$Intercept\ (\beta_0)$: The intercept provides an estimate, 5.91, for a home in Brookside (reference Neighborhood) with 0 square footage. Although this value is not practical, it is a statistically significant value extrapolated from the model.

$LogGrLivArea(\beta_1)$: For every additional 100 square feet in a home, the estimated / predicted sale price increases by 0.82.

$Neighborhood\ Edwards(\beta_2)$: The adjustment of the intercept for a home in Edwards with respect to Brookside (reference neighborhood). For a home with 0 square feet, this home has an estimated sale price 2.09 times more than a home in Brookside.

$Neighborhood\ NAmes\ (\beta_3)$: The adjustment of the intercept for a home in North Ames with respect to Brookside (reference neighborhood). For a home with 0 square feet, this home has an estimated sale price 2.58 times more than a home in Brookside.

$LogGrLivArea \cdot NAmes(\beta_4)$: For every additional 100 square feet for a home in North Ames, the estimated / predicted sale price decreases 0.3 from the change with respect to Brookside.

$LogGrLivArea \cdot Edwards\ (\beta_5)$: For every additional 100 square feet for a home in North Ames, the estimated / predicted sale price decreases 0.34 from the change with respect to Brookside.

Conclusion

It is estimated that the doubling of a home's square footage is associated with a 76% ($2^{0.82}$ = 1.76) increase in the sale price for a home in Brookside. A 95% confidence interval for this increase is approximately 60% to 95% (1.60, 1.95) in sale price while accounting for North Ames and Edwards neighborhoods. Additionally, the sale price for a home in Edwards is predicted to be on average 26% ($e^{-0.3}$ = 0.74) less than a home in Brookside (p-value = 0.0011) while the sale price for a home in North Ames is predicted to be on average 29% ($e^{-0.35}$= 0.71) less compared to a home in Brookside (p-value < 0.0001). We are 95% confident that the median home sale price for Edwards will be approximately between 11% and 38% less than a home in Brookside based on square footage (95% CI: $-0.3 \pm 1.97 \cdot 0.09$ = [-0.48, -0.12]; $e^{-0.48}$, $e^{-0.12}$; [0.62, 0.89]). We also estimate the 95%

confidence interval for the median home sale price for North Ames will be approximately between 16% and 40% less than a home in Brookside based on square footage (95% CI: $-0.35 \pm 1.97 \cdot 0.085$ = [-0.51, -0.182]; $e^{-0.513}$, $e^{-0.18}$; [0.60, 0.84]). The proportion of variation in home sale price that is explained by the combination of explanatory variables, square feet and neighborhood, is 51.21%.

We can only associate these home sale prices within the given population sample. This is an observational study; no casual inference or generalizations can be made about the effects of on sale prices for the entire population of homes.

Analysis Question 2:

What is the best model to predict home sale prices for all homes in Ames, Iowa?

Single Linear Regression Model

$$\Pr e\, d\ \log(Sale\ \Pr i\, ce)\ =\ \beta_0\ +\ \beta_1 \cdot \log(GrLivArea)$$

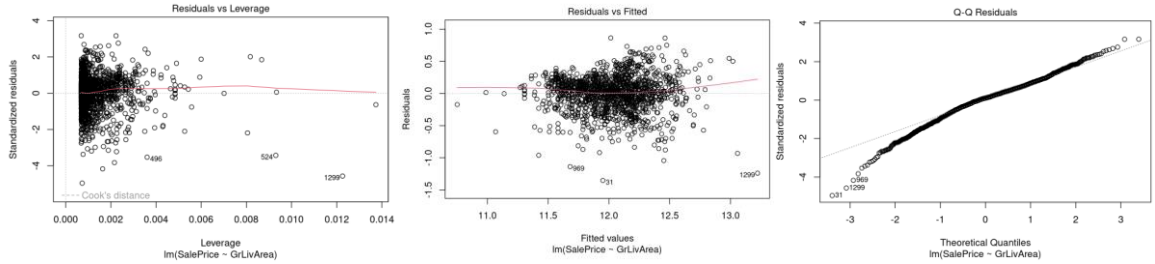


Fig. 5. Residual plots for single linear regression model

Multiple Linear Regression Model

$$pred\ \log(Sale\ \Pr i\, ce)\ =\ \beta_0 + \beta_1 \cdot \log(GrLivArea)\ +\ \beta_2 \cdot FullBath$$

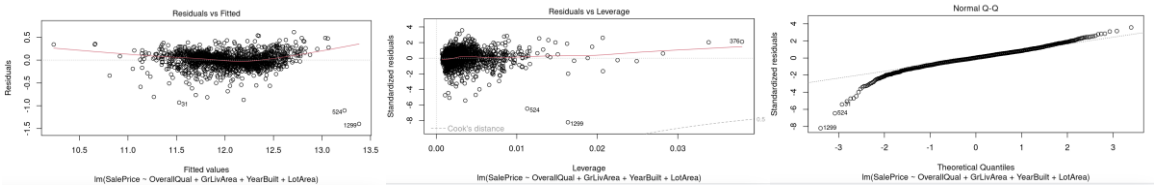
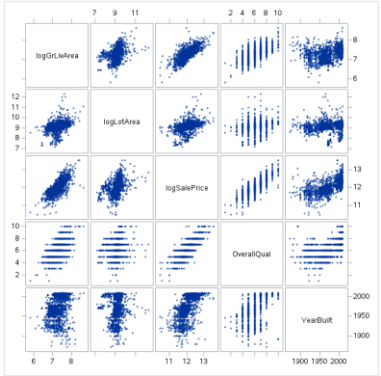


Fig. 6. Residual plots for multiple linear regression model

Multiple Linear Regression Model 2

$$pred\ \log(Sale\ \Pr i\, ce)$$
$$= \beta_0 + \beta_1 \cdot OverallQual + \beta_2 \cdot \log(GrLivArea) + \beta_3 \cdot YearBuilt + \beta_4 \cdot \log(LotArea)$$




```
##
## Call:
## lm(formula = SalePrice ~ OverallQual + GrLivArea + YearBuilt +
##     LotArea, data = log_transformed_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.39900 -0.08517  0.00640  0.09932  0.61313
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -47.122391   2.705045  -17.42   <2e-16 ***
## OverallQual   0.810103   0.033400   24.25   <2e-16 ***
## GrLivArea     0.413013   0.018401   22.45   <2e-16 ***
## YearBuilt     7.014078   0.358032   19.59   <2e-16 ***
## LotArea       0.149057   0.009471   15.74   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1719 on 1455 degrees of freedom
## Multiple R-squared:  0.8154, Adjusted R-squared:  0.8148
## F-statistic:  1606 on 4 and 1455 DF,  p-value: < 2.2e-16
```

Fig. 7. scatter plot for the multiple linear regression model (right) and parameter estimates (left)

Multiple Linear Regression Model 3

$$Pred[\log(Sales\ price)] = \beta_0 + \beta_1 * 1stFlrSF + \beta_2 * YearBuilt + \beta_3 * ScreenPorch + \beta_4 * LottFrontage + \beta_5 * YearRemodAdd + \beta_6 * Exterior2nd + \beta_7 * Exterior1st + \beta_8 * Functional + \beta_9 * GarageQual + \beta_{10} * BsmtFullBath$$

Checking Assumptions : we want to check assumption for multiple linear regression model 2 which has highest kaggle score and lowest CV press.
Residual Plots : The residual plot resembles of a random scattered near zero with some of outliers on the negative side.
Influential point analysis (Cook's D and Leverage) : The high leverage and low cook's D are presented for point number 1299 but most of the points are scattered as low leverage and low cook's D.

- Linearity: According to the linearity of qq plot, we can assume the linear model can be applied on this dataset.
- Normality: The normal distribution of residuals, we can assume the data is normally distributed.
- Constant variance: From the randomly distributed of residuals in the residual scatter plot, we can assume constant variance of the data.
- Independent observations: We assume this data is collected independently according to the large size of dataset and normally distributed.

Comparing Competing Models

Predictive Models	Adjusted R2	CV PRESS	Kaggle Score
Single Linear Regression	0.533	109.0518	0.28948
Multiple Linear Regression	0.555	104.0772	0.28409
Multiple linear regression 3	0.9035	1.64277E12	0.19978
Multiple linear regression 2	0.8118	43.41798	0.16933

Table 4. Model evaluation scores for analysis 2.

Adjusted R^2: The highest adjusted R^2 is 0.9035 for the multiple linear regression 3.
CV PRESS: The lowest CV Press is 43.41798 for the multiple linear regression 2.
Kaggle Score: The lowest Kaggle score is 0.16933 for the Multiple linear regression model 2.

Conclusion:

In this analysis, we predict the sales price of houses in the Ames, Iowa using linear regression models. The variable selection method is applied to find best model which is multiple linear regression 2 whose CV Press is 43.4 and Kaggle Score is 0.169.

REFERENCES

Kaggle Competition:

https://www.kaggle.com/c/house-prices-advanced-regression-techniques

GitHub Page Links

https://ddxbugs.github.io/
https://gwonchan.github.io/2022/11/26/why_i_choose_data_science/

APPENDIX

Analysis 1: SAS

```
* Reduced model;
proc glm data=logtrain;
    class Neighborhood(ref="BrkSide");
    model LogSalePrice=LogLivArea Neighborhood / solution;
    run;
*** STEPWISE Full v. Reduced CVPRESS ADJRSQ;
proc glmselect data=logtrain seed=1232112;
    class Neighborhood(ref="BrkSide");
    model LogSalePrice=LogLivArea | Neighborhood / selection=stepwise(stop=cv) cvdetails=cvpress stats=adjrsq;
proc glmselect data=logtrain seed=1232112;
    class Neighborhood(ref="BrkSide");
    model LogSalePrice=LogLivArea Neighborhood / selection=stepwise(stop=cv) cvdetails=cvpress stats=adjrsq;
```

Analysis 1: R

```
############### analysis 1
data <- read.csv("train.csv")

# Filter neighborhoods
filtered_data <- subset(data, Neighborhood %in% c("NAmes", "Edwards", "BrkSide"))

# Create dummy variables for neighborhoods
filtered_data <- mutate(filtered_data,NAmes_dummy = ifelse(Neighborhood == "NAmes", 1, 0),Edwards_dummy = ifelse(Neighborhoo
d == "Edwards", 1, 0))
```

```
# Log transformation of SalePrice and GrLivArea
filtered_data$log_SalePrice <- log(filtered_data$SalePrice)
filtered_data$log_GrLivArea <- log(filtered_data$GrLivArea)
```

```
# Fit the regression model
model <- lm(log_SalePrice ~ log_GrLivArea + log_GrLivArea * NAmes_dummy + log_GrLivArea * Edwards_dummy, data = filtered_dat
a)
# Print the summary of the model
summary(model)
```

```
# Obtain the residuals
residuals <- residuals(model)

# Create a histogram of residuals
hist(residuals, main = "Histogram of Residuals", xlab = "Residuals", ylab = "Frequency")
```

```
# Scatter plot with regression lines for all neighborhoods
ggplot(filtered_data, aes(x = log_GrLivArea, y = log_SalePrice, color = Neighborhood)) +
  geom_point(shape=1,size =1) +  # Add points
  geom_smooth(aes(group = Neighborhood), method = "lm", se = FALSE) +  # Add regression lines without confidence bands
  labs(title = "Scatter Plot of GrLivArea vs. SalePrice by Neighborhood",
       x = "log_GrLivArea (100 sqft)",
       y = "log_SalePrice ($)",
       color = "Neighborhood")
```

```
# simple linear regression models
model1 <- lm(log_SalePrice ~ log_GrLivArea, data = filtered_data)
summary(model1)

# Create a scatter plot
plot(filtered_data$log_GrLivArea,filtered_data$log_SalePrice,
     ylab = "log_SalePrice", xlab = "log_GrLivArea",
     main = "Scatter Plot of log_SalePrice vs. log_GrLivArea")

# Add regression line to the plot
abline(a= 7.753378, b= 0.5682412, col = "red")
```

```
# Calculate adjusted R-squared for each model
adj_r_squared <- summary(model)$adj.r.squared
adj_r_squared1 <- summary(model1)$adj.r.squared
```

Analysis 2: SAS

```
*** MLR 1;
proc reg data=logtrain plots=(CooksD all);
    model logSalePrice = logGrLivArea FullBath;
    run;
*** MLR 2;
data mlr2;
    set WORK.IMPORT;
    logGrLivArea = log(GrLivArea);
    logLotArea = log(LotArea);
    logSalePrice = log(SalePrice);
    keep Id logGrLivArea logLotArea OverallQual YearBuilt logSalePrice;
    run;

proc sgscatter data=mlr2;
    matrix logGrLivArea logLotArea logSalePrice OverallQual YearBuilt;
    run;

proc corr data=mlr2;
    var logGrLivArea logLotArea logSalePrice OverallQual YearBuilt;
    run;

proc glmselect data=mlr2 seed=222222;
    class OverallQual YearBuilt;
    model logSalePrice = logGrLivArea logLotArea | OverallQual YearBuilt / selection = forward(stop=cv) cvmethod=random(5) cvdetails stats=adjrsq;
    run;
proc glmselect data=mlr2 seed=222222;
    class OverallQual YearBuilt;
    model logSalePrice = logGrLivArea logLotArea | OverallQual YearBuilt / selection = backward(stop=cv) cvmethod=random(5) cvdetails stats=adjrsq;
    run;
proc glmselect data=mlr2 seed=222222;
    class OverallQual YearBuilt;
    model logSalePrice = logGrLivArea logLotArea | OverallQual YearBuilt / selection = stepwise(stop=cv) cvmethod=random(5) cvdetails stats=adjrsq;
    run;

proc glm data=mlr2 plots(unpack)=all;
    model logSalePrice = logGrLivArea logLotArea | OverallQual YearBuilt / solution;
    run;
proc glm data=sample plots=all;
    class Street LandContour Utilities Neighborhood Condition2 BldgType
    OverallQual OverallCond RoofMatl MasVnrType ExterQual BsmtQual BsmtExposure
    HeatingQC FullBath BedroomAbvGr KitchenAbvGr KitchenQual TotRmsAbvGrd
    FireplaceQu GarageFinish GarageCars GarageCond PoolQC SaleType;
    model SalePrice = LotArea Street LandContour Utilities Neighborhood Condition2
    BldgType OverallQual OverallCond RoofMatl MasVnrType MasVnrArea ExterQual
    BsmtQual BsmtExposure BsmtFinSF1 BsmtFinSF2 BsmtUnfSF HeatingQC FullBath BedroomAbvGr
    KitchenAbvGr KitchenQual TotRmsAbvGrd FireplaceQu GarageFinish GarageCars GarageCond
    WoodDeckSF PoolArea PoolQC SaleType / solution;
    output out = result p = predict;
```

Analysis 2: R

```r
########## analysis 2
data <- read.csv("train.csv")

# Select only numeric columns
numeric_data <- data[, sapply(data, is.numeric)]

# Check the structure of the new dataset
str(numeric_data)
```

```r
# Apply log transformation to each numeric column

log_transformed_data <- log(numeric_data+1)  # Adding 1 to avoid log(0) issues
```

```r
###################### Simple Linear Regression
fit1 = lm(SalePrice~GrLivArea, data = log_transformed_data)
summary(fit1)
```

```r
# Step 1: Create cross-validation folds
folds <- createFolds(log_transformed_data$SalePrice, k = 5)  # Create 5-fold cross-validation

# Step 2-4: Fit model and compute residuals for each fold
press_residuals <- sapply(folds, function(i) {
  # Split data into training and testing sets
  train_data <- log_transformed_data[-i, ]  # Exclude the fold 'i' for training
  test_data <- log_transformed_data[i, ]    # Fold 'i' for testing

  # Fit your regression model (replace 'lm' with your desired model)
  model <- lm(SalePrice ~ GrLivArea, data = train_data)

  # Predict on the testing set
  predictions <- predict(model, newdata = test_data)

  # Compute residuals
  residuals <- test_data$SalePrice - predictions

  # Return the squared residuals
  return(residuals^2)
})

# Step 5: Compute PRESS statistic
cv_press <- sum(unlist(press_residuals))

cv_press
```

```r
###################### Multiple Linear Regression
fit0 = lm(SalePrice~GrLivArea + FullBath, data = log_transformed_data)
summary(fit0)
```

```r
# Load required libraries
library(caret)

## CV press
# Step 1: Create cross-validation folds
folds <- createFolds(log_transformed_data$SalePrice, k = 5)  # Create 5-fold cross-validation

# Step 2-4: Fit model and compute residuals for each fold
press_residuals <- sapply(folds, function(i) {
  # Split data into training and testing sets
  train_data <- log_transformed_data[-i, ]  # Exclude the fold 'i' for training
  test_data <- log_transformed_data[i, ]    # Fold 'i' for testing

  # Fit your regression model (replace 'lm' with your desired model)
  model <- lm(SalePrice ~ GrLivArea + FullBath, data = train_data)

  # Predict on the testing set
  predictions <- predict(model, newdata = test_data)

  # Compute residuals
  residuals <- test_data$SalePrice - predictions

  # Return the squared residuals
  return(residuals^2)
})

# Step 5: Compute PRESS statistic
cv_press <- sum(unlist(press_residuals))

cv_press
```

```r
# Stepwise
ols_step_both_p(fit, prem = 0.05, pent = 0.05, details = FALSE)
```

```r
# vaiable selection: OverallQual, GrLivArea, YearBuilt, LotArea
fit2 = lm(SalePrice~ OverallQual + GrLivArea + YearBuilt + LotArea, data = log_transformed_data)

summary(fit2)
```

```r
############################ variable selection for MLR 2
fit = lm(SalePrice ~ ., data = log_transformed_data)
# Backward
#ols_step_backward_p(fit, prem = 0.05, details = TRUE)

# Forward
#ols_step_forward_p(fit, penter = 0.05, details = TRUE)

library(olsrr)
```

```r
## CV press
# Step 1: Create cross-validation folds
folds <- createFolds(log_transformed_data$SalePrice, k = 5)  # Create 5-fold cross-validation

# Step 2-4: Fit model and compute residuals for each fold
press_residuals <- sapply(folds, function(i) {
  # Split data into training and testing sets
  train_data <- log_transformed_data[-i, ]  # Exclude the fold 'i' for training
  test_data <- log_transformed_data[i, ]    # Fold 'i' for testing

  # Fit your regression model (replace 'lm' with your desired model)
  model <- lm(SalePrice ~ OverallQual + GrLivArea + YearBuilt + LotArea, data = train_data)

  # Predict on the testing set
  predictions <- predict(model, newdata = test_data)

  # Compute residuals
  residuals <- test_data$SalePrice - predictions

  # Return the squared residuals
  return(residuals^2)
})

# Step 5: Compute PRESS statistic
cv_press <- sum(unlist(press_residuals))

cv_press
```

```r
################## kaggle submission
# Read the test data from test.csv
test_data <- read.csv("test.csv", header = TRUE)

# Check the structure of the test data
# str(test_data)

# Select only numeric columns
numeric_test_data <- test_data[, sapply(test_data, is.numeric)]

# Check the structure of the new dataset
# str(numeric_test_data)

# Apply log transformation to each numeric column
log_transformed_test_data <- log(numeric_test_data+1)

# Make predictions on the test data
predictions <- predict(fit0, newdata = log_transformed_test_data)

# View the predictions
# print(predictions)

# Back-transform the predicted values
back_transformed_predictions <- exp(predictions)

# Check the length of the back_transformed_predictions vector
# length(back_transformed_predictions)

test_data_ids <- test_data$Id  # Replace "ID" with the actual column name in your test data
predictions <- back_transformed_predictions  # Replace "your_predictions_vector" with your actual vector of predictions

# Create a data frame with the test data IDs and predictions
submission_df <- data.frame(Id = test_data_ids, SalePrice = predictions)

# Write the data frame to a CSV file
write.csv(submission_df, "submission.csv", row.names = FALSE)
```