Ames Iowa Housing Project

By Christy Wachira and Christian Castro

15 April 2024

Introduction

Background and Focus

The Ames, Iowa Housing dataset, sourced from Kaggle, contains 79 explanatory variables describing residential homes in Ames, Iowa, and offers a unique opportunity to explore the factors influencing house prices.

Century 21 Ames, a real estate company operating in the North Ames, Edwards, and Brook Side neighborhoods, has commissioned an analysis to understand the relationship between living area square footage and sale price in these specific neighborhoods. Additionally, developing a robust predictive model for house prices across all neighborhoods in Ames can provide valuable insights into the local housing market.

Objectives and Questions of Interest

1. Investigate the relationship between sale price and living area square footage in the NAmes, Edwards, and BrkSide neighborhoods:
   - How does living area affect sale price?
   - Does this relationship vary across neighborhoods?
   - Can we quantify the relationship while accounting for neighborhood differences?
2. Develop and compare predictive models for house prices across all neighborhoods in Ames:
   - How well can a simple linear regression model predict house prices?
   - Can multiple linear regression models improve predictive performance?
   - Which combination of explanatory variables yields the best predictive model?

By addressing these objectives using the Ames, Iowa Housing dataset, we aim to provide insights to support data-driven decision-making in the local real estate market.

Data Description

Data Source and Size

The dataset used in this study is the Ames, Iowa Housing dataset, sourced from Kaggle, a popular online platform for data science competitions and datasets. The dataset was compiled by Dean De Cock for use in data science education and is based on public information from the Ames, Iowa Assessor's Office.

The dataset contains information on residential property sales in Ames, Iowa, from 2006 to 2010. It consists of 2,930 observations and 80 variables, including 79 explanatory variables and the target variable, 'SalePrice'. The explanatory variables encompass a wide range of features, such as lot size, building type, number of rooms, various area measurements, overall quality, year built, and many others.

The dataset is split into two parts: a training set (1,460 observations) and a test set (1,459 observations). The training set includes the 'SalePrice' variable and is used for model building and evaluation. The test set does not include the 'SalePrice' variable and is used for the Kaggle competition to assess the predictive performance of the developed models.

Variables of Interest

ID: Unique identifier for each house

Building Class (MSSubClass): Identifies the type of dwelling in the sale

Zoning (MSZoning): Identifies the general zoning classification of the sale

Neighborhood: Identifies the physical locations within Ames city limits

Size of the Front Property Line (LotFrontage): Linear feet of street connected to the property

Lot Area (LotArea): Lot size in square feet

Overall Quality (OverallQual): Rates the overall material and finish of the house

Year Built (YearBuilt): Indicates the original construction date of the house

Exterior Material (Exterior1st, Exterior2nd): Identifies the exterior covering on the house

Living Area (GrLivArea): Represents the above-ground living area square footage

Kitchen Quality (KItchenQual): Evaluates the quality of the kitchen

Total Basement Area (TotalBsmtSF): Represents the total square footage of the basement area

Garage Area (GarageArea): Measures the size of the garage in square feet

Sale Type (SaleType): Identifies the type of sale

Analysis Question 1: Relationship between Sale Price and Living Area in Specific Neighborhoods

Restatement of Problem

Century 21 Ames wants to determine if the sale price and its relationship to the living area vary depending on the neighborhood, particularly in the North Ames, BrookSide and Edwards neighborhoods. Additionally, they would like to quantify this relationship while accounting for neighborhood differences and provide estimates for each neighborhood in 100 square feet increments.

Model Building and Fitting

To investigate the relationship between sale price and living area in the specified neighborhoods, we will build a multiple linear regression model. The model will include GrLivArea as the main explanatory variable and two dummy variables to represent the three neighborhoods of interest.

The model equation can be represented as follows:

$\text{SalePrice} = \beta_0 + \beta_1 * \text{GrLivArea} + \beta_2 * \text{Edwards} + \beta_3 * \text{BrkSide} + \varepsilon$

where:

- SalePrice is the response variable
- GrLivArea is the main explanatory variable representing the above-grade living area in square feet
- Edwards and BrkSide are dummy variables representing the respective neighborhoods (NAmes is treated as the reference level)
- β_0, β_1, β_2, and β_3 are the coefficients to be estimated
- ε is the error term

We will fit this model using the ordinary least squares (OLS) method on the relevant subset of the Ames, Iowa Housing dataset, which includes only the observations from the NAmes, Edwards, and BrkSide neighborhoods.

Checking Assumptions

Linearity: According to Figure 1, the data show a linear relationship between price and square footage

Independence: Given that the data consists of individual house sales, we can reasonably assume that the observations are independent.

Normality: We will examine the QQ plot of the residuals of the fitted model to assess the normality assumption. Based on Figure 2 and Figure 3, we will assume normality because the QQ plot appears to fit the line well and the residuals appear random.

Equal Variance: Based on Figure 3, we will assume equal variance due to the random nature of the distribution of residuals.

Influential point analysis (Cook's D and Leverage)

Figure 4 is a graph of Cook's Distance by Observation Number. Visually there is further investigation needed.

There are a few points with a high Cook's Distance

Influential points:

19 48 58 64 70 80 104 131 136 140 157 167 169 180 186 190 205 227 240 302 322 339

370 372

While there seem to be a lot of influential points, in a data set of almost 3000 observations, these are not that many. After reviewing the points, they tend to be at the extremes of the data and are not typical of the houses assessed. We will proceed with them in case the edge cases of the data tend to follow a trend that is not easily observed.

Parameter Estimates and Interpretations

After fitting the model and checking the assumptions, we will interpret the coefficients and their statistical significance. We will focus on the following:

- The coefficient for GrLivArea (β_1) represents the change in sale price for a one-square-foot increase in living area, holding the neighborhood constant.
- The coefficients for Edwards (β_2) and BrkSide (β_3) represent the difference in sale price compared to the reference neighborhood (NAmes), holding the living area constant.

We will also assess the overall model performance using metrics such as the adjusted R-squared and the F-test for overall significance.

See Figure 12, 13

The R^2 for the model is 0.3965 and the F statistic is 83 on 3 and 379 Degrees of Freedom. These results mean that only 39.65% of the results can be explained by the model.

Confidence Intervals

To view the Confidence Intervals, see Figures: 5, 6, 7, 8, 12.

The baseline intercept for the Brookside neighborhood is $69,781.54, the intercept for the Edwards neighborhood is $66,899.38, and the intercept for the North Ames neighborhood is $85,887.16. For each neighborhood, there is an associated $45.76 increase in sale price per square foot, or rather, An estimated $4,576.00 increase for every 100sqft increase in living area.

Conclusion

Based on the Linear Model using North Ames as the reference, the studied neighborhoods can expect an associated $4,576 increase in sale price for every 100sqft on top of the baselines provided above. The North Ames neighborhood tends to be the most expensive neighborhood out of the ones Century 21 Ames operates in, followed by Brookside and then Edwards. It is worth exploring how other variables can impact the sale price as having only <40% of the story is not advisable.

R Shiny: Price v. Living Area Chart

https://cdcastr0.shinyapps.io/Ames_Housing_Project/

The scatterplots of the house prices vs living area can be found on the web app above.

Analysis Question 2:

Predictive Models for House Prices in Ames, Iowa

Problem Restatement

The primary objective is to predict the sale prices of homes in Ames, Iowa, using various factors that might influence these prices. Three candidate models were considered to evaluate how well they predict the sale prices based on different sets of predictors.

Candidate Models

Simple Linear Regression (SLR):

Sale Price = β0 +β1 (GrLivArea) + ϵ

Predicts sale prices based solely on the ground living area (GrLivArea).

Multiple Linear Regression 1 (MLR 1):

Sale Price = β0 +β1 (GrLivArea) + β2 (FullBath) + ϵ

Incorporates both the ground living area (GrLivArea) and the number of full bathrooms (FullBath).

Multiple Linear Regression 2 (MLR 2):

Sale Price = β0 +β1 (GrLivArea) + β2 (1stFlrSF) + β3 (YearBuilt) + ϵ

Extending the predictors to include ground living area (GrLivArea), first floor square footage (X1stFlrSF), and year built (YearBuilt).

Checking Assumptions

SLR

The residual plot against the predicted values shows a pattern, indicating potential non-linearity or an omitted variable bias. Residuals increase with the increase in predicted values, suggesting homogeneity of variance. There are a few points with high Cook's D values that could be potential outliers or influential points. Most data points have low leverage, but there are some points with high leverage, indicating they might have a substantial impact on the parameter estimates.

See Figure 9.

MLR 1

There is less of a pattern in the residuals for MLR 1 compared to SLR, but there still appears to be some curvature and homogeneity of variance present. This indicates that while adding FullBath has improved the model, there may still be missing explanatory factors. Like SLR, there are a few observations with a higher Cook's D value, suggesting they could be influential to the model fit. The leverage plot shows that there are a few points with higher leverage compared to the rest of the data, which might be impacting the regression estimates.

See Figure 10

MLR 2

The residuals appear to show a slight improvement in homogeneity of variance and linearity compared to the previous models. There is an observation with a particularly high Cook's D value, indicating a very influential point in the dataset that could be disproportionately affecting the model's predictions. As with Cook's D, there is a notable observation with high leverage, which should be investigated further as it can have an effect on the regression equation.

See Figure 11

Comparing Competing Models

Predictive Models	Adj R^2	Internal CV Press	Kaggle Score
Simple Linear Regression	0.5018	3,193,255,762	52,161.08
Multiple Linear Regression	0.5231	3,065,726,086	50,581.51
Custom MLR model	0.6824	2,040,900,419	40,214.63

Conclusion

The analysis of the three models indicates that MLR 2 (SalePrice ~ GrLivArea + X1stFlrSF + YearBuilt) is the most robust in predicting future sale prices of homes in Ames, Iowa. This model not only provides the highest adjusted R2 value, showing that it explains a greater variance in sale prices than the other models, but it also has the lowest CV PRESS and RMSE scores, indicating superior predictive accuracy and reliability. The inclusion of the house's age (YearBuilt) and first-floor square footage (X1stFlrSF) alongside the living area significantly enhances the model's effectiveness. This model is recommended for stakeholders interested in an accurate and reliable prediction of home prices in Ames, potentially assisting in real estate investment decisions, market analysis, and economic studies related to housing market dynamics.

Appendix

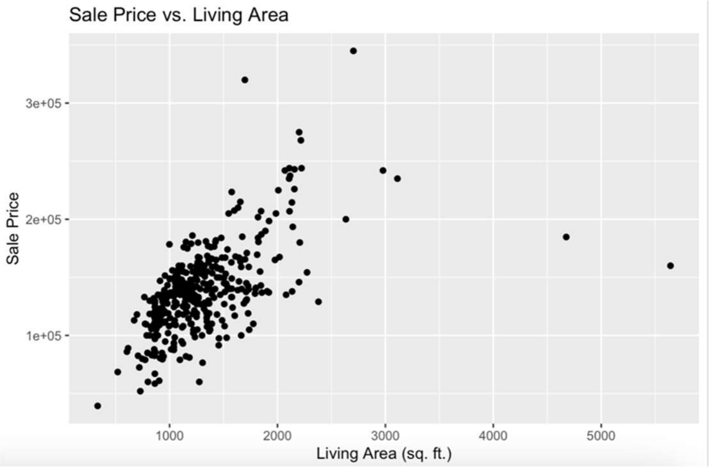


Figure 1: Scatter Plot of Sale Price vs Living Area

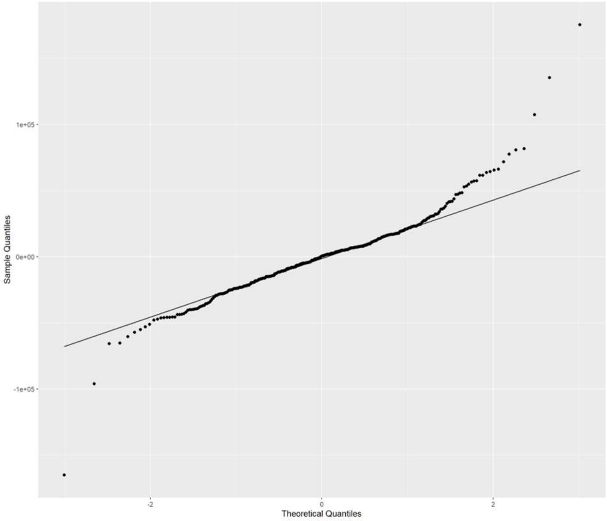


Figure 2: QQ Plot

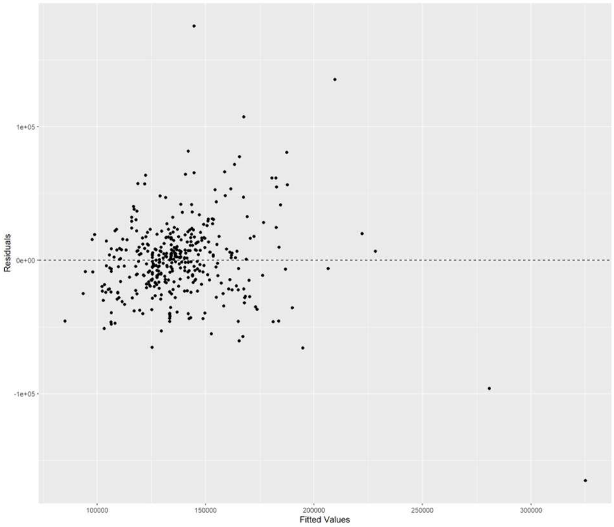


Figure 3: Scatter Plot of Residuals

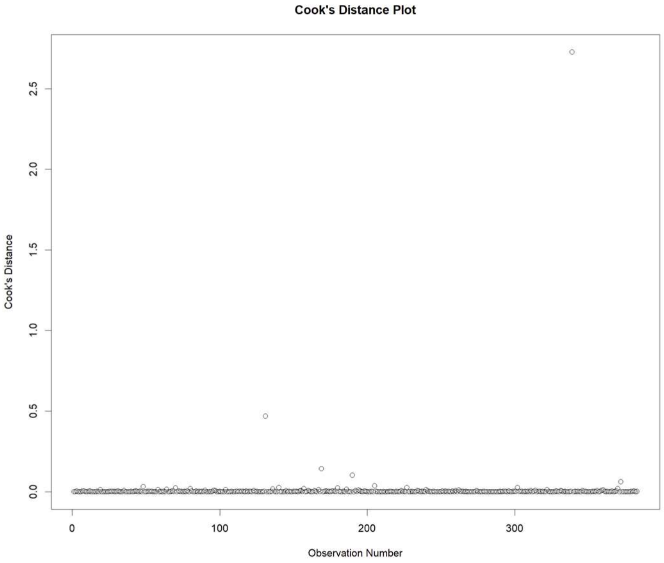


Figure 4: Plot of Cook’s D by House ID

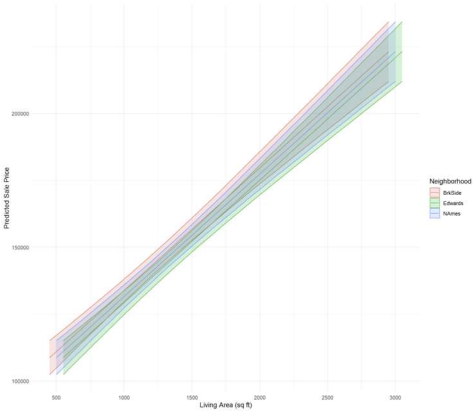


Figure 5: Confidence Intervals for three neighborhoods

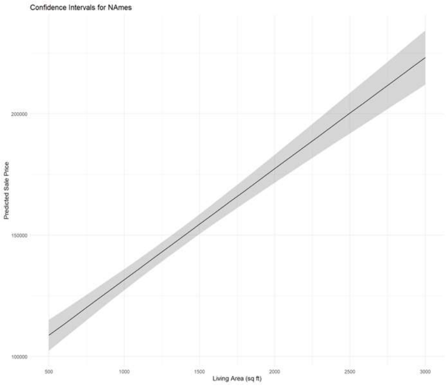


Figure 6: Confidence Interval for North Ames

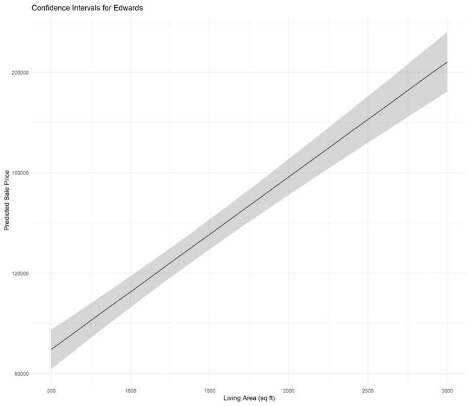


Figure 7: Confidence Interval for Edwards

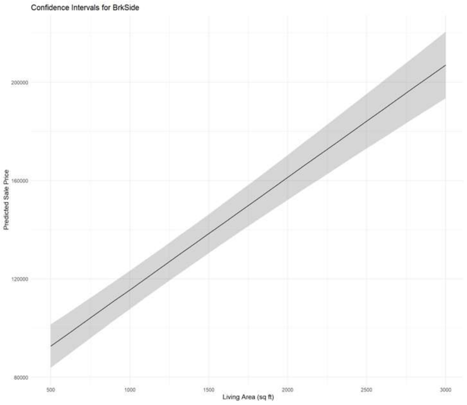
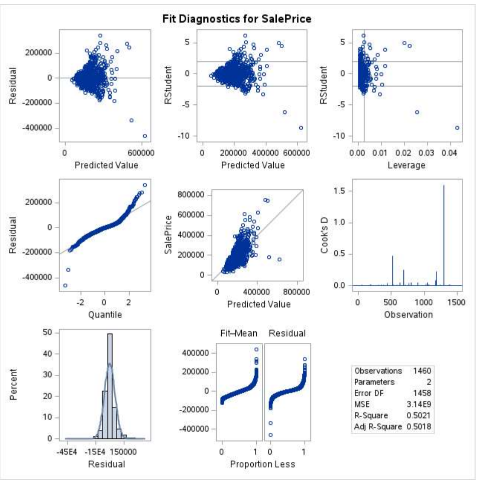
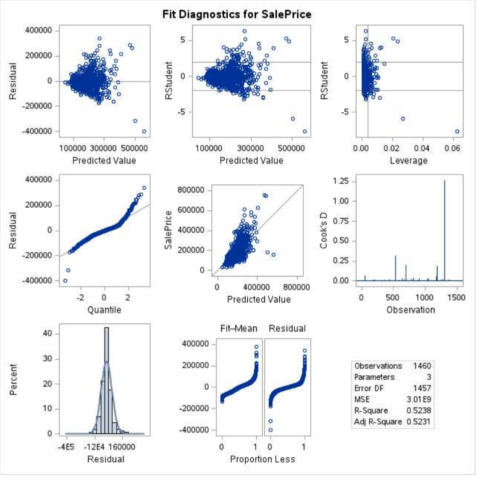


Figure 8: Confidence Intervals for Brook Side



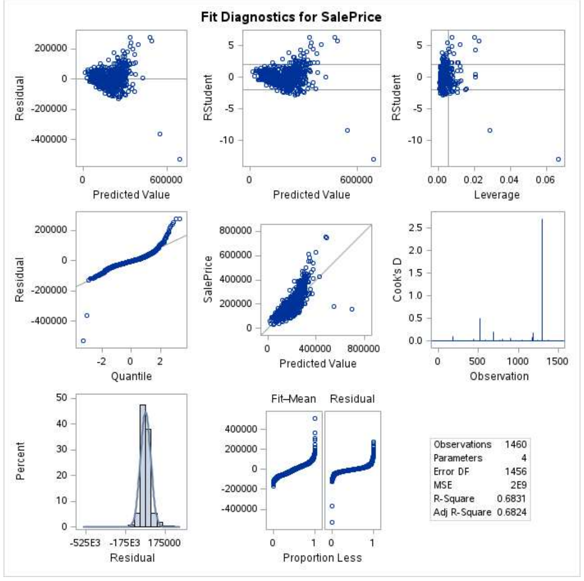
Observations	1460
Parameters	2
Error DF	1458
MSE	3.14E9
R-Square	0.5021
Adj R-Square	0.5018

Figure 9: SLR residuals, Cook’s D and leverage



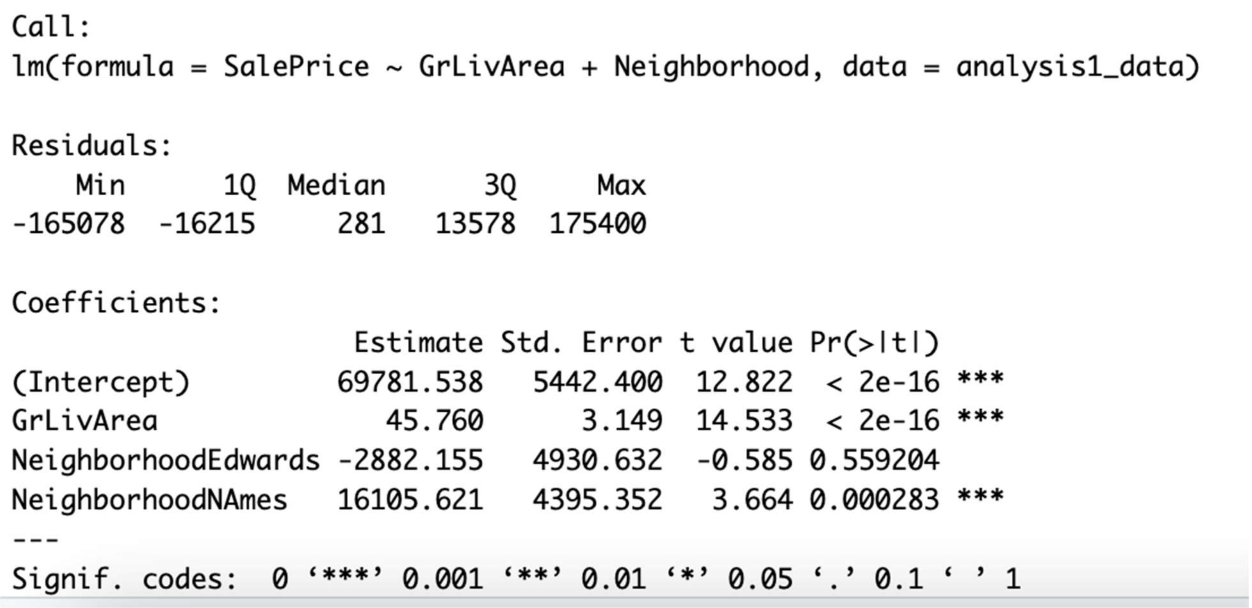
Observations	1460
Parameters	3
Error DF	1457
MSE	3.01E9
R-Square	0.5238
Adj R-Square	0.5231

Figure 10: MLR residuals, Cook’s D and leverage



Observations	1460
Parameters	4
Error DF	1456
MSE	2E9
R-Square	0.6831
Adj R-Square	0.6824

Figure 11: Custom MLR residuals, Cook's D and leverage

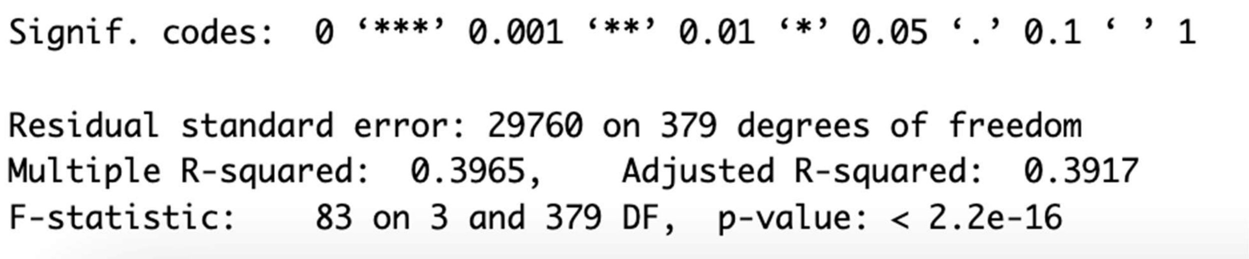


```
Call:
lm(formula = SalePrice ~ GrLivArea + Neighborhood, data = analysis1_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-165078  -16215     281   13578  175400 

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)    
(Intercept)         69781.538   5442.400  12.822  < 2e-16 ***
GrLivArea              45.760      3.149  14.533  < 2e-16 ***
NeighborhoodEdwards -2882.155   4930.632  -0.585 0.559204    
NeighborhoodNAmes   16105.621   4395.352   3.664 0.000283 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 12: Parameter Estimates for Analysis 1 Model



```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 29760 on 379 degrees of freedom
Multiple R-squared:  0.3965,	Adjusted R-squared:  0.3917 
F-statistic:    83 on 3 and 379 DF,  p-value: < 2.2e-16
```

Figure 13: Continued Parameter Estimates for Analysis 1 Model

Websites:

CDCastr0.github.io

Chrvstyww.github.io

SAS or R Code for Analysis 1:

```
# Load required libraries
library(dplyr)
library(ggplot2)
library(car)
```

```
# Read the dataset (assuming it's in CSV format)
data <- read.csv(file.choose())

# Select relevant variables and neighborhoods for Analysis 1
analysis1_data <- data %>%
  select(SalePrice, GrLivArea, Neighborhood) %>%
  filter(Neighborhood %in% c("NAmes", "Edwards", "BrkSide"))

# Summary statistics
summary(analysis1_data)

plot(analysis1_data)

# Histogram of SalePrice
ggplot(analysis1_data, aes(x = SalePrice)) +
  geom_histogram(bins = 30, fill = "blue", color = "white") +
  labs(title = "Distribution of Sale Prices", x = "Sale Price", y = "Frequency")

# Scatterplot of SalePrice vs. GrLivArea
ggplot(analysis1_data, aes(x = GrLivArea, y = SalePrice)) +
  geom_point() +
  labs(title = "Sale Price vs. Living Area", x = "Living Area (sq. ft.)", y = "Sale Price")

# Scatterplot of SalePrice vs. GrLivArea colored by Neighborhood
ggplot(analysis1_data, aes(x = GrLivArea, y = SalePrice, color = Neighborhood)) +
  geom_point() +
  labs(title = "Sale Price vs. Living Area by Neighborhood",
       x = "Living Area (sq. ft.)", y = "Sale Price") +
  scale_color_discrete(name = "Neighborhood")

# Boxplot of SalePrice by Neighborhood
ggplot(analysis1_data, aes(x = Neighborhood, y = SalePrice)) +
  geom_boxplot(fill = "blue", color = "black") +
  labs(title = "Sale Price Distribution by Neighborhood",
       x = "Neighborhood", y = "Sale Price")

# Summary statistics by Neighborhood
analysis1_data %>%
  group_by(Neighborhood) %>%
  summarise(
    mean_price = mean(SalePrice),
    median_price = median(SalePrice),
    min_price = min(SalePrice),
    max_price = max(SalePrice),
    mean_living_area = mean(GrLivArea),
    median_living_area = median(GrLivArea),
    min_living_area = min(GrLivArea),
    max_living_area = max(GrLivArea)
  )

# Build and fit the model
model <- lm(SalePrice ~ GrLivArea + Neighborhood, data = analysis1_data)
summary(model)

# Check assumptions
# Residual plots
par(mfrow = c(2, 2))
plot(model)

# Influential point analysis
influencePlot(model)

# Calculate estimates and confidence intervals
# Assuming the company wants estimates for each neighborhood at GrLivArea increments of 100 sq. ft.
new_data <- data.frame(
  GrLivArea = rep(seq(500, 2500, by = 100), 3),
  Neighborhood = rep(c("NAmes", "Edwards", "BrkSide"), each = 21)
)

predictions <- predict(model, newdata = new_data, interval = "confidence")
results <- cbind(new_data, predictions)

# Display the results
print(results)
```

RShiny

```
# Load necessary libraries
library(shiny)
library(ggplot2)
library(dplyr)

# Load the Ames, Iowa Housing dataset (assuming it's in the working directory)
ames_data <- read.csv("train.csv")

# Define the UI
ui <- fluidPage(
  titlePanel("House Price vs. Living Area"),
  sidebarLayout(
    sidebarPanel(
      selectInput("neighborhood", "Select Neighborhood:",
                  choices = c("NAmes", "Edwards", "BrkSide"),
                  selected = "NAmes")
    ),

    mainPanel(
      plotOutput("scatterplot")
    )
  )
)

# Define the server logic
server <- function(input, output) {

  # Filter the data based on the selected neighborhood
  filtered_data <- reactive({
    ames_data %>%
      filter(Neighborhood == input$neighborhood)
  })

  # Create the scatterplot
  output$scatterplot <- renderPlot({
    ggplot(filtered_data(), aes(x = GrLivArea, y = SalePrice)) +
      geom_point() +
      labs(x = "Living Area (sq ft)", y = "Sale Price") +
      ggtitle(paste("House Price vs. Living Area in", input$neighborhood)) +
      theme_minimal()
  })
}

# Run the app
shinyApp(ui = ui, server = server)
```

SAS or R Code for Analysis 2:

SAS:

```
FILENAME REFFILE '/home/u63727602/sasuser.v94/My New Data/train.csv';

PROC IMPORT DATAFILE=REFFILE
DBMS=CSV
OUT=train_data;
GETNAMES=YES;
RUN;

proc print DATA=train_data;
RUN;
```

```
/* SLR */
proc reg data=train_data;
   model SalePrice = GrLivArea;
   output out=slr_results r=residual p=predicted cookd=cooksd;
run;
quit;



/* MLR 1 */
proc reg data=train_data;
   model SalePrice = GrLivArea FullBath;
   output out=mlr1_results r=residual p=predicted cookd=cooksd;
run;
quit;



/* MLR 2 */
proc reg data=train_data;
   model SalePrice = GrLivArea '1stFlrSF'n  YearBuilt;
   output out=mlr2_results r=residual p=predicted cookd=cooksd;
run;
quit;
```

R code:

```
## Read the Data
# Load the necessary libraries
library(car)

## Explore and Prepare the Data
# Read the training data
train_data <- read.csv("/Users/christywachira/Downloads/train.csv", stringsAsFactors = FALSE)
summary(train_data)

test_data <- read.csv("/Users/christywachira/Downloads/test.csv", stringsAsFactors = FALSE)
summary(test_data)

## Ensure that data has no missing values in the predictor variables
sum(is.na(test_data$GrLivArea))
sum(is.na(test_data$FullBath))
sum(is.na(test_data$X1stFlrSF))
sum(is.na(test_data$YearBuilt))

## Build Models
# SLR model
model_simple <- lm(SalePrice ~ GrLivArea, data=train_data)
summary(model_simple)

# MLR model 1
model_multiple_specified <- lm(SalePrice ~ GrLivArea + FullBath, data=train_data)
```

```
summary(model_multiple_specified)

# MLR model 2
model_multiple_custom <- lm(SalePrice ~ GrLivArea + X1stFlrSF + YearBuilt, data=train_data)
summary(model_multiple_custom)

##Evaluate Models
install.packages("boot")
library(boot)

# Calculate CV PRESS
calculate_cv_press <- function(data, formula) {
  glm_model <- glm(formula, data = data)
  cv_glm <- cv.glm(data, glm_model, K=10)
  return(cv_glm$delta[1])
}

# Calculate CV PRESS for each model
cv_press_simple <- calculate_cv_press(train_data, SalePrice ~ GrLivArea)
cv_press_mlr1 <- calculate_cv_press(train_data, SalePrice ~ GrLivArea + FullBath)
cv_press_mlr2 <- calculate_cv_press(train_data, SalePrice ~ GrLivArea + X1stFlrSF + YearBuilt)
cv_press_simple
cv_press_mlr1
cv_press_mlr2

predictions <- predict(model_multiple_custom, newdata=test_data)
head(predictions)

# Split the data into training and validation sets
set.seed(123)
train_indices <- sample(1:nrow(train_data), 0.8 * nrow(train_data))
validation_data <- train_data[-train_indices, ]
train_subset <- train_data[train_indices, ]

# Fit models and calculate RMSE on the validation set
calculate_rmse <- function(model_formula, train_data, validation_data) {
  model <- lm(model_formula, data = train_data)
  predictions <- predict(model, newdata = validation_data)
  rmse <- sqrt(mean((validation_data$SalePrice - predictions)^2))
  return(rmse)
}

# RMSE for each model
rmse_simple <- calculate_rmse(SalePrice ~ GrLivArea, train_subset, validation_data)
rmse_mlr1 <- calculate_rmse(SalePrice ~ GrLivArea + FullBath, train_subset, validation_data)
rmse_mlr2 <- calculate_rmse(SalePrice ~ GrLivArea + X1stFlrSF + YearBuilt, train_subset, validation_data)
rmse_simple
rmse_mlr1
rmse_mlr2
```