EE5373: Data Modeling Using R

Fall 2025

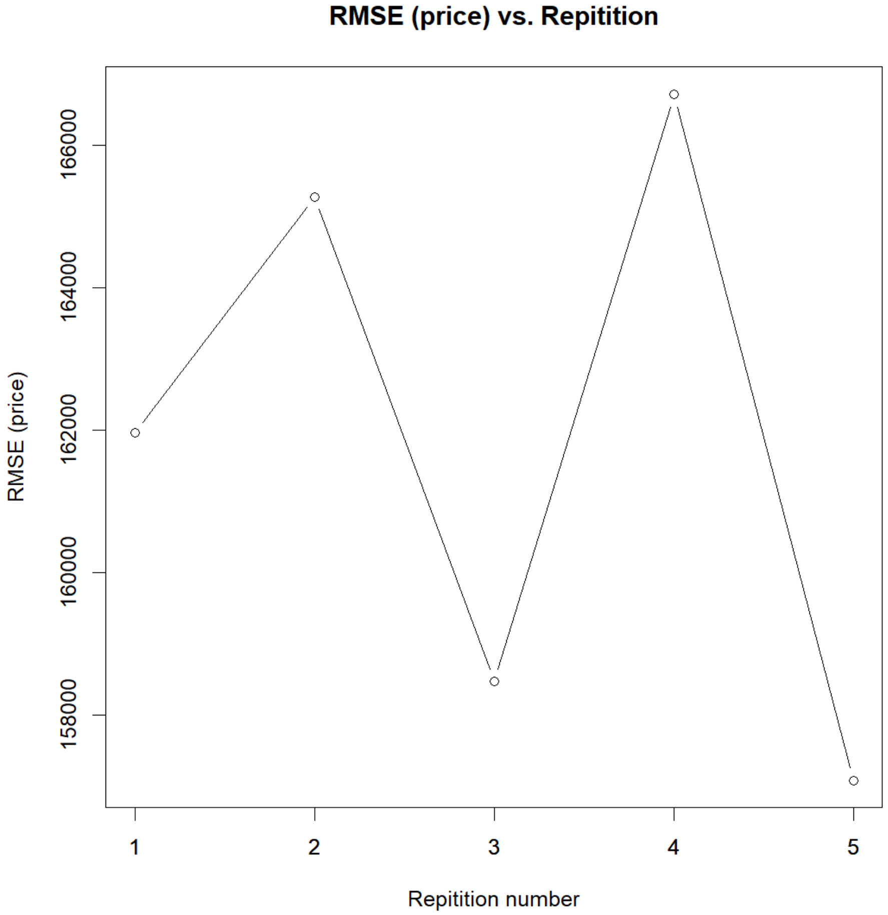
Lab 6
House Price Prediction

Talha Hamza

November 14, 2025

Problem 1: Backward Elimination Regression Model

Results



Discussion

Before fitting the model, I removed the variables unique ID column, since it does not contain meaningful housing information.

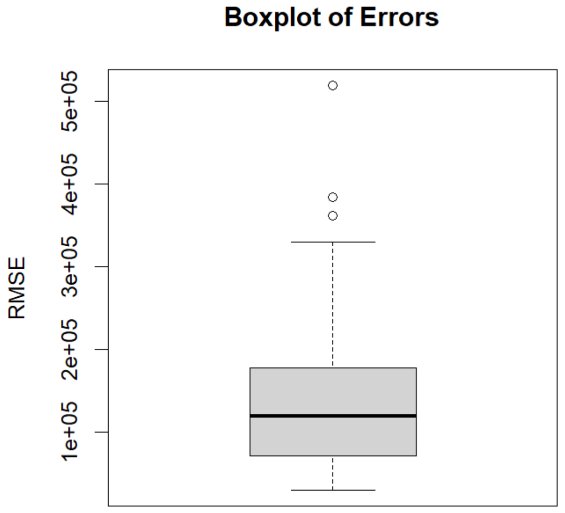
Additionally, the original date variable was converted into two separate predictors: year and month. Also, I removed the original timestamp format, which was leading to troublesome behaviour of the linear model.

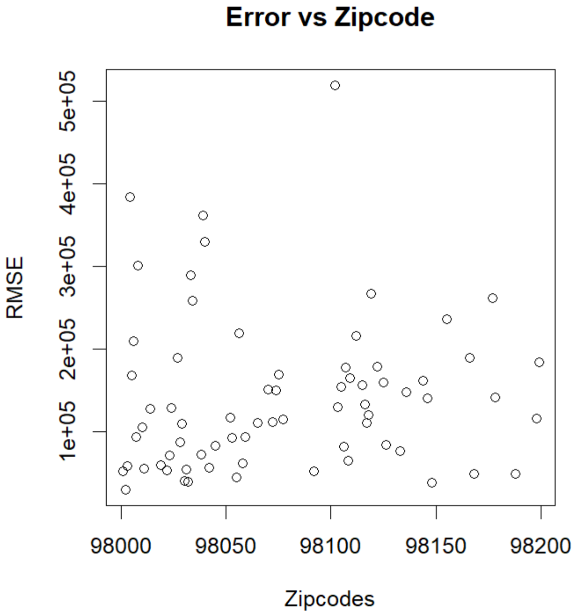
Several categorical variables **(waterfront, view, condition, zipcode)** were converted into qualitative variable to represent their true nature.

During backward elimination, I removed predictors that had a large p-value (>0.05) such as long and sqft_basement.

Problem 2: Zip Code–Segmented Models

Results





Discussion

Across zip codes, RMSE values vary widely, ranging from 25,511.04 to 511,846.7, with several clear outliers visible in both the boxplot and the Error vs. Zipcode scatterplot.

One reason for this variation is that high-priced neighborhoods tend to have more volatile pricing, where small differences in home features can lead to large price swings. This is supported by the strong correlation between median price and RMSE (r = 0.786), indicating that prediction error increases as typical home value increases.

Interestingly, the number of houses per zip code had little impact on model accuracy. The correlation between zip code sample size and RMSE was only –0.108, suggesting that sample size does not explain the variation in errors across zip codes.

When comparing these results to Problem 1, I observe that zip-code-specific models yield higher RMSEs. This makes sense as the model in Problem 1 uses many additional variable and was trained using the entire dataset, in contrast to just zip code subsets.

Appendix: R Code Listings

Problem 1 Code

```
library(dplyr)

raw_house_data <- read.csv("kc_house_data.csv")
house_data <- na.omit(raw_house_data)

# Convert date and extract useful time features
house_data$date <- as.Date(house_data$date, format = "%Y%m%dT%H%M%S")
house_data$year <- as.numeric(format(house_data$date, "%Y"))
house_data$month <- as.numeric(format(house_data$date, "%m"))

# drop id and date
house_data <- house_data %>%
  select(-id, -date)

# Convert categorical features
house_data$zipcode <- as.factor(house_data$zipcode)
house_data$waterfront <- as.factor(house_data$waterfront)
house_data$view <- as.factor(house_data$view)
house_data$condition <- as.factor(house_data$condition)
# house_data$floors <- as.factor(house_data$floors)

rows <- nrow(house_data)

attempts <- c(1,2,3,4,5)
rmss_vector <- vector()

for (i in seq_along(attempts)) {
  f <- 0.6

  perm <- house_data[sample(rows), ]
  train.dat <- perm[1:floor(f * rows), ]
  test.dat  <- perm[(floor(f * rows) + 1):rows, ]

  # Linear model using
  house.lm <- lm((price) ~ bedrooms + bathrooms + sqft_living +
                   floors + waterfront + view + condition + grade +
                   sqft_above + yr_built + yr_renovated +
                   zipcode + lat +
                   sqft_living15 + sqft_lot15 +
                   year + month,
                 data = train.dat)
  pred.log <- predict(house.lm, newdata = test.dat)

  # RMSE
  rmse <- sqrt(mean(((test.dat$price) - pred.log)^2))
  rmss_vector[i] <- rmse
}

plot(attempts, rmss_vector,
     type = "b",
     main = "RMSE (price) vs. Repitition",
     xlab = "Repitition number",
     ylab = "RMSE (price)")
axis(1, at = attempts, labels = attempts)
```

Problem 2 Code

```
library(dplyr)

raw_house_data <- read.csv("kc_house_data.csv")
house_data <- na.omit(raw_house_data)

# Define new function
price_prediction_error <- function(price, bedrooms, bathrooms, sqft_
    living,
                                   sqft_lot, grade, yr_built) {

  house_info <- data.frame(price, bedrooms, bathrooms, sqft_living,
     sqft_lot, grade, yr_built)

  rows <- nrow(house_info)
  f <- 0.6

  perm <- house_info[sample(rows), ]
  train.dat <- perm[1:floor(f * rows), ]
  test.dat  <- perm[(floor(f * rows) + 1):rows, ]

  # Linear model
  house.lm <- lm(price ~ bedrooms + bathrooms + sqft_living + sqft_
     lot + grade
                 + yr_built, data = train.dat)

  # Predictions & RMSE
  pred <- predict(house.lm, newdata = test.dat)
  rmse <- sqrt(mean((test.dat$price - pred)^2))

  return(rmse)
}

# Group by zipcode
data_by_zipcode <- house_data %>%
  group_by(zipcode) %>%
  summarize(
    count = n(),
    med_price = median(price),
    med_yr_built = median(yr_built),
    error = price_prediction_error(price, bedrooms, bathrooms, sqft_
       living, sqft_lot, grade, yr_built)
  )

plot( x= data_by_zipcode$zipcode,
      y= data_by_zipcode$error,
      xlab = "Zipcodes",
      ylab = "RMSE",
      main = "Error vs Zipcode"
)

boxplot(data_by_zipcode$error,
        ylab = "RMSE",
        main = "Boxplot of Errors")

# Why do some zipcodes have higher RMSE than other?
cor(data_by_zipcode$med_price, data_by_zipcode$error)

plot(data_by_zipcode$med_price, data_by_zipcode$error,
     pch = 19,
     xlab = "Median Price",
     ylab = "RMSE",
     main = "Higher Priced Zip Codes = Higher RMSE?")

cor(data_by_zipcode$count, data_by_zipcode$error, use = "complete.
   obs")

plot(data_by_zipcode$count, data_by_zipcode$error,
     pch = 19,
     xlab = "Number of Houses in Zipcode",
     ylab = "RMSE",
     main = "Does Sample Size Affect RMSE?")
```