Predicting House Prices using Linear Regression

2025-03-29

```r
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```r
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.3.3
```

```
## Loading required package: lattice
```

```r
library(tidymodels)
```

```
## Warning: package 'tidymodels' was built under R version 4.3.3
```

```
## ── Attaching packages ────────────────────────────────────── tidymodels 1.3.0 ──
```

```
## ✔ broom        1.0.8     ✔ rsample      1.2.1
## ✔ dials        1.4.0     ✔ tibble       3.2.1
## ✔ dplyr        1.1.4     ✔ tidyr        1.3.1
## ✔ infer        1.0.7     ✔ tune         1.3.0
## ✔ modeldata    1.4.0     ✔ workflows    1.2.0
## ✔ parsnip      1.3.1     ✔ workflowsets 1.1.0
## ✔ purrr        1.0.4     ✔ yardstick    1.3.2
## ✔ recipes      1.2.1
```

```
## Warning: package 'broom' was built under R version 4.3.3
```

```
## Warning: package 'dials' was built under R version 4.3.3
```

```
## Warning: package 'scales' was built under R version 4.3.3
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
## Warning: package 'infer' was built under R version 4.3.3
```

```
## Warning: package 'modeldata' was built under R version 4.3.3
```

```
## Warning: package 'parsnip' was built under R version 4.3.3
```

```
## Warning: package 'purrr' was built under R version 4.3.3
```

```
## Warning: package 'recipes' was built under R version 4.3.3
```

```
## Warning: package 'rsample' was built under R version 4.3.3
```

```
## Warning: package 'tibble' was built under R version 4.3.3
```

```
## Warning: package 'tidyr' was built under R version 4.3.3
```

```
## Warning: package 'tune' was built under R version 4.3.3
```

```
## Warning: package 'workflows' was built under R version 4.3.3
```

```
## Warning: package 'workflowsets' was built under R version 4.3.3
```

```
## Warning: package 'yardstick' was built under R version 4.3.3
```

```
## ── Conflicts ───────────────────────────────────────── tidymodels_conflicts() ──
## ✖ purrr::discard()         masks scales::discard()
## ✖ dplyr::filter()          masks stats::filter()
## ✖ dplyr::lag()             masks stats::lag()
## ✖ purrr::lift()            masks caret::lift()
## ✖ yardstick::precision()   masks caret::precision()
## ✖ yardstick::recall()      masks caret::recall()
## ✖ yardstick::sensitivity() masks caret::sensitivity()
## ✖ yardstick::specificity() masks caret::specificity()
## ✖ recipes::step()          masks stats::step()
```

```r
library(dplyr)
```

```r
setwd("D:/R2 programming/archive")
```

```r
data<-read.csv(file.choose())
```

```r
data <- read.csv("Mumbai House Prices.csv", stringsAsFactors = FALSE)
nrow(data)  # Should return 159
```

```
## [1] 76038
```

```r
head(data)
```

```
##   bhk      type                           locality area price price_unit
## 1   3 Apartment  Lak And Hanware The Residency Tower  685  2.50         Cr
## 2   2 Apartment    Radheya Sai Enclave Building No 2  640 52.51          L
## 3   2 Apartment                        Romell Serene  610  1.73         Cr
## 4   2 Apartment Soundlines Codename Urban Rainforest  876 59.98          L
## 5   2 Apartment                        Origin Oriana  659 94.11          L
## 6   2 Apartment         Bhoomi Simana Wing A Phase 1  826  3.30         Cr
##          region             status age
## 1   Andheri West      Ready to move New
## 2   Naigaon East Under Construction New
## 3  Borivali West Under Construction New
## 4         Panvel Under Construction New
## 5 Mira Road East Under Construction New
## 6          Parel Under Construction New
```

```r
head(data)
```

```
##   bhk      type                           locality area price price_unit
## 1   3 Apartment  Lak And Hanware The Residency Tower  685  2.50         Cr
## 2   2 Apartment    Radheya Sai Enclave Building No 2  640 52.51          L
## 3   2 Apartment                        Romell Serene  610  1.73         Cr
## 4   2 Apartment Soundlines Codename Urban Rainforest  876 59.98          L
## 5   2 Apartment                        Origin Oriana  659 94.11          L
## 6   2 Apartment         Bhoomi Simana Wing A Phase 1  826  3.30         Cr
##          region             status age
## 1   Andheri West      Ready to move New
## 2   Naigaon East Under Construction New
## 3  Borivali West Under Construction New
## 4         Panvel Under Construction New
## 5 Mira Road East Under Construction New
## 6          Parel Under Construction New
```

```r
if ("price_unit" %in% colnames(data)) {
  data <- data %>%
    mutate(price = ifelse(price_unit == "Cr", price * 100, price)) %>%
    select(-price_unit)  # Remove `price_unit` after processing
} else {
  message("Column 'price_unit' not found. Skipping transformation.")
}
```

```r
colnames(data)
```

```
## [1] "bhk"      "type"     "locality" "area"     "price"    "region"   "status"  
## [8] "age"
```

```r
dim(data)
```

```
## [1] 76038     8
```

```r
str(data)
```

```
## 'data.frame':    76038 obs. of  8 variables:
##  $ bhk     : int  3 2 2 2 2 2 5 3 1 2 ...
##  $ type    : chr  "Apartment" "Apartment" "Apartment" "Apartment" ...
##  $ locality: chr  "Lak And Hanware The Residency Tower" "Radheya Sai Enclave Building No 2" "Romell Serene" "S
oundlines Codename Urban Rainforest" ...
##  $ area    : int  685 640 610 876 659 826 2921 778 396 671 ...
##  $ price   : num  250 52.5 173 60 94.1 ...
##  $ region  : chr  "Andheri West" "Naigaon East" "Borivali West" "Panvel" ...
##  $ status  : chr  "Ready to move" "Under Construction" "Under Construction" "Under Construction" ...
##  $ age     : chr  "New" "New" "New" "New" ...
```

```r
nrow(data)
```

```
## [1] 76038
```

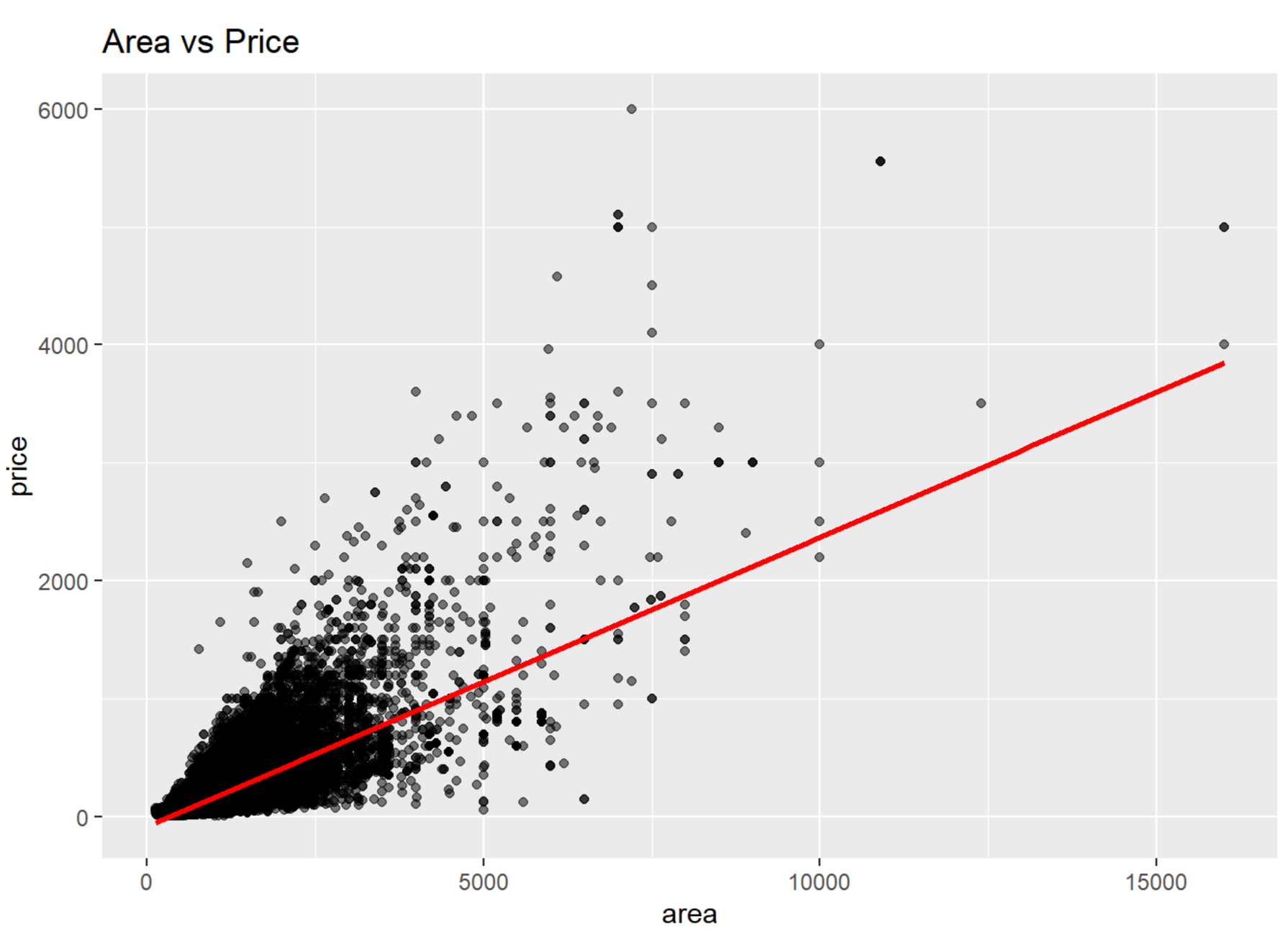
```r
"region" %in% colnames(data)
```

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

```r
data$region <- as.factor(data$region)
data$locality <- as.factor(data$locality)
data$status <- as.factor(data$status)
data$age <- as.factor(data$age)
```

```r
ggplot(data, aes(x = area, y = price)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", col = "red") +
  ggtitle("Area vs Price")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```r
numeric_data <- data %>% select(area, bhk, price)
cor_matrix <- cor(numeric_data)
print(cor_matrix)
```

```
##            area       bhk     price
## area  1.0000000 0.7877379 0.7560003
## bhk   0.7877379 1.0000000 0.6313526
## price 0.7560003 0.6313526 1.0000000
```

```r
set.seed(123)
data_split <- initial_split(data, prop = 0.8)
train_data <- training(data_split)
test_data <- testing(data_split)
```

```r
lin_reg_spec <- linear_reg() %>%
  set_engine("lm") %>%
  set_mode("regression")
```

```r
lin_reg_fit <- lin_reg_spec %>%
  fit(price ~ area + bhk + region + status + age, data = train_data)
```

```r
test_data$region <- factor(test_data$region, levels = levels(train_data$region))
```

```r
data$region <- as.character(data$region)  # Convert to character to avoid factor issues

set.seed(123)
data_split <- initial_split(data, prop = 0.8)
train_data <- training(data_split)
test_data <- testing(data_split)

# Convert back to factor based on training levels
train_data$region <- factor(train_data$region)
test_data$region <- factor(test_data$region, levels = levels(train_data$region))

lin_reg_fit <- lin_reg_spec %>%
  fit(price ~ area + bhk + region + status + age, data = train_data)
```

```r
predictions <- predict(lin_reg_fit, new_data = test_data)
results <- bind_cols(test_data, predictions)
```

```r
library(yardstick)

# Get predictions
test_predictions <- predict(lin_reg_fit, new_data = test_data) %>%
  bind_cols(test_data)  # Combine predictions with actual data

# Compute R-squared (R²)
rsq_value <- rsq(test_predictions, truth = price, estimate = .pred)

print(rsq_value)
```

```
## # A tibble: 1 × 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rsq     standard       0.773
```

```r
library(yardstick)

# Generate predictions
test_predictions <- predict(lin_reg_fit, new_data = test_data) %>%
  bind_cols(test_data)
rmse_value <- rmse(test_predictions, truth = price, estimate = .pred)

print(rmse_value)
```

```
## # A tibble: 1 × 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard        106.
```

```r
cat("R-Squared:", rsq_value$.estimate, "\n")
```

```
## R-Squared: 0.773212
```

```r
cat("RMSE:", rmse_value$.estimate, "\n")
```

```
## RMSE: 106.1375
```

```r
cv_folds <- vfold_cv(train_data, v = 5)
cv_results <- fit_resamples(lin_reg_spec, price ~ area + bhk + region + status + age, resamples = cv_folds)
```

```
## → A | warning: prediction from rank-deficient fit; consider predict(., rankdeficient="NA")
```

```
## There were issues with some computations   A: x1There were issues with some computations   A: x2There were iss
ues with some computations   A: x3There were issues with some computations   A: x4There were issues with some com
putations   A: x5There were issues with some computations   A: x5
```

```r
chooseCRANmirror(graphics = FALSE, ind = 1)
install.packages("car")
```

```
## Installing package into 'C:/Users/JAIPAL SINGH/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

```
## package 'car' successfully unpacked and MD5 sums checked
## 
## The downloaded binary packages are in
##  C:\Users\JAIPAL SINGH\AppData\Local\Temp\RtmpO4OUuB\downloaded_packages
```

```r
library(car)
```

```
## Warning: package 'car' was built under R version 4.3.3
```

```
## Loading required package: carData
```

```
## Warning: package 'carData' was built under R version 4.3.3
```

```
## 
## Attaching package: 'car'
```

```
## The following object is masked from 'package:purrr':
## 
##     some
```

```
## The following object is masked from 'package:dplyr':
## 
##     recode
```

```r
# Fit a linear model to check VIF
vif_values <- vif(lm(price ~ area + bhk + region + status + age, data = train_data))

print(vif_values)
```

```
##            GVIF  Df GVIF^(1/(2*Df))
## area   3.032046   1        1.741277
## bhk    2.962269   1        1.721124
## region 1.902982 213        1.001512
## status 1.712799   1        1.308739
## age    2.145086   2        1.210211
```

```r
collect_metrics(cv_results)
```

```
## # A tibble: 2 × 6
##   .metric .estimator    mean     n std_err .config
##   <chr>   <chr>        <dbl> <int>   <dbl> <chr>
## 1 rmse    standard   105.        5 1.63    Preprocessor1_Model1
## 2 rsq     standard     0.763     5 0.00344 Preprocessor1_Model1
```