Multiple Regression

2024-09-20

1 Participate in Kaggle's House Prices: Advanced Regression Techniques competition, and implement the following steps:

1. Fit a multiple regression model involving at least 6 predictor variables, with at least **three categorical variables**.
2. Justify why you chose these 6 variables and justify why you omitted others.
3. Apply the fitted/trained model to the training data and compute the root mean squared error using `dplyr` and other R functions. In other words, do not use a `rmse()` function from another R package.
4. Write a `submissions.csv` file that when submitted on Kaggle, returns a valid score.
5. Take a screenshot of your Kaggle score and compare it to the score you computed earlier.
6. On Moodle, submit a `.zip` compressed/archived file of this entire RStudio project folder. We are doing this to ensure the graders can reproduce your Quarto file.

2 Exploratory data analysis

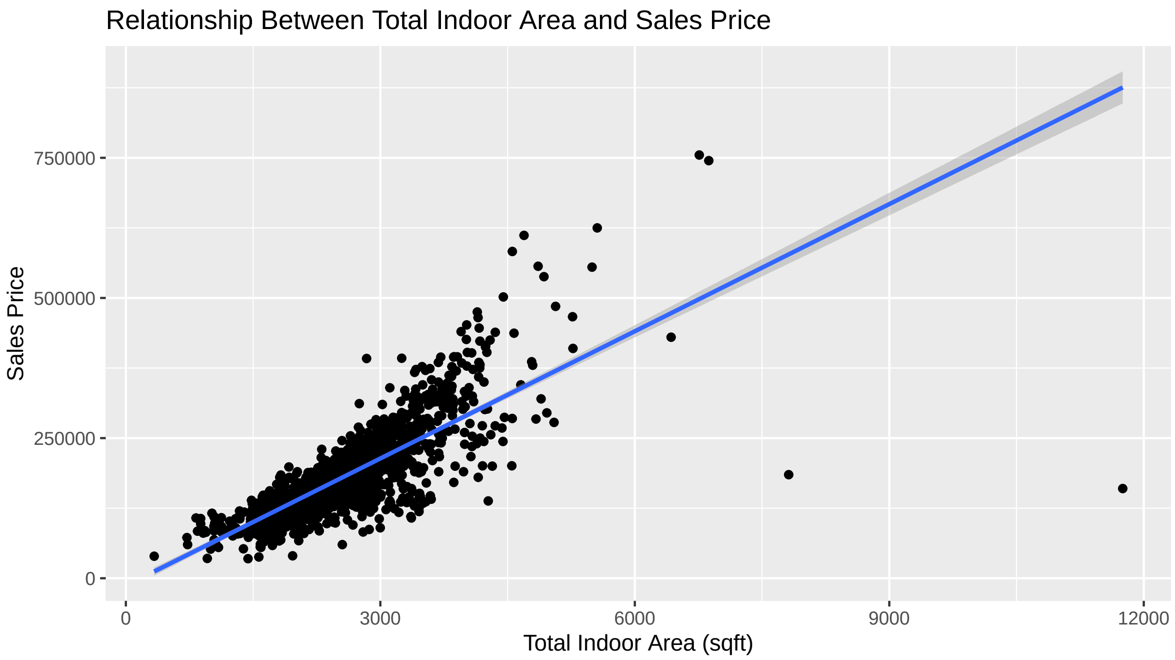
2.1 Choice of variables

Chose six predictor variables and explained the reasons for selecting them.

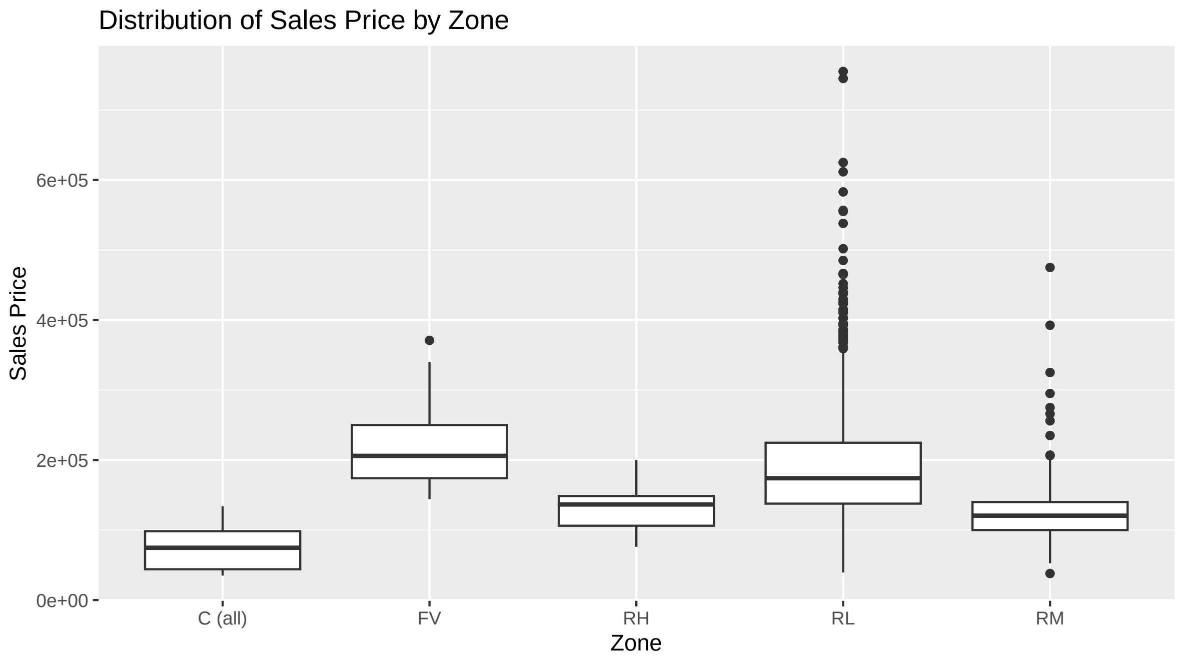
- totalSF: According to the scatterplot of total square feet vs. sales price, I found the 2 variables are highly correlated and sales price increases as totalSF increases.
- MSZoning: The distributions of pricing are generally different across the categories of zoning. In particular, the distributions of price for FV and RL are higher than the other 3 categories.
- Utilities: The 2 categories have discernable differences in distributions of sales price. AllPub shows a wide range of sales prices, while all of the sales prices for NoSeWa are concentrated in the low sales price range.

- Neighborhood: All neighborhoods possess very distinct distributions of sales prices. For example, BrkSide lies in the $150,000 price range, while StoneBr has price ranges in the $400,000 range.
- YearBuilt: The 2 variables are strongly positively correlated because the sales price increases as the year built increases.
- outdoorEntArea: There is a strong, positive correlation between the total outdoor entertainment area and sales price. When the total outdoor entertainment area increases, sales price increases as well.

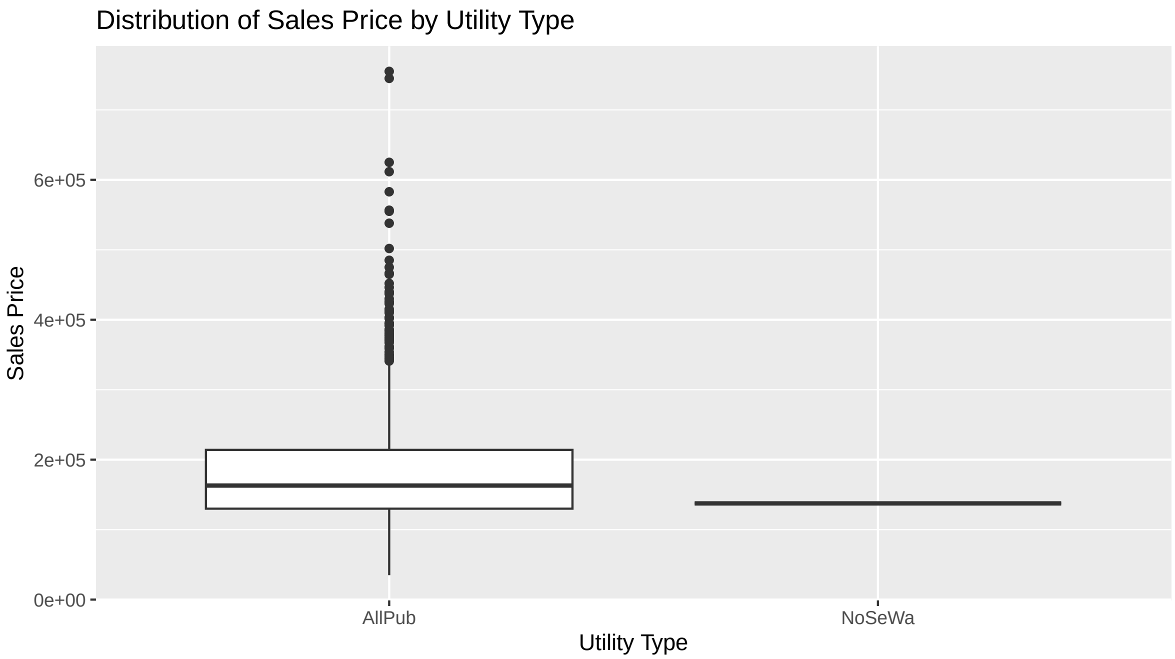
```r
# rename first floor area variable
training <- training %>%
  rename(firstFloorSF = '1stFlrSF')

# rename 2nd floor area variable
training <- training %>%
  rename(secondFloorSF = '2ndFlrSF')

# add up areas of all sections of the home
training <- training %>%
  mutate(totalSF = firstFloorSF + secondFloorSF + TotalBsmtSF)

# scatterplot of total sq ft vs sale price
ggplot(data = training, aes(x = totalSF, y = SalePrice)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "Relationship Between Total Indoor Area and Sales Price",
       x = "Total Indoor Area (sqft)",
       y = "Sales Price")
```



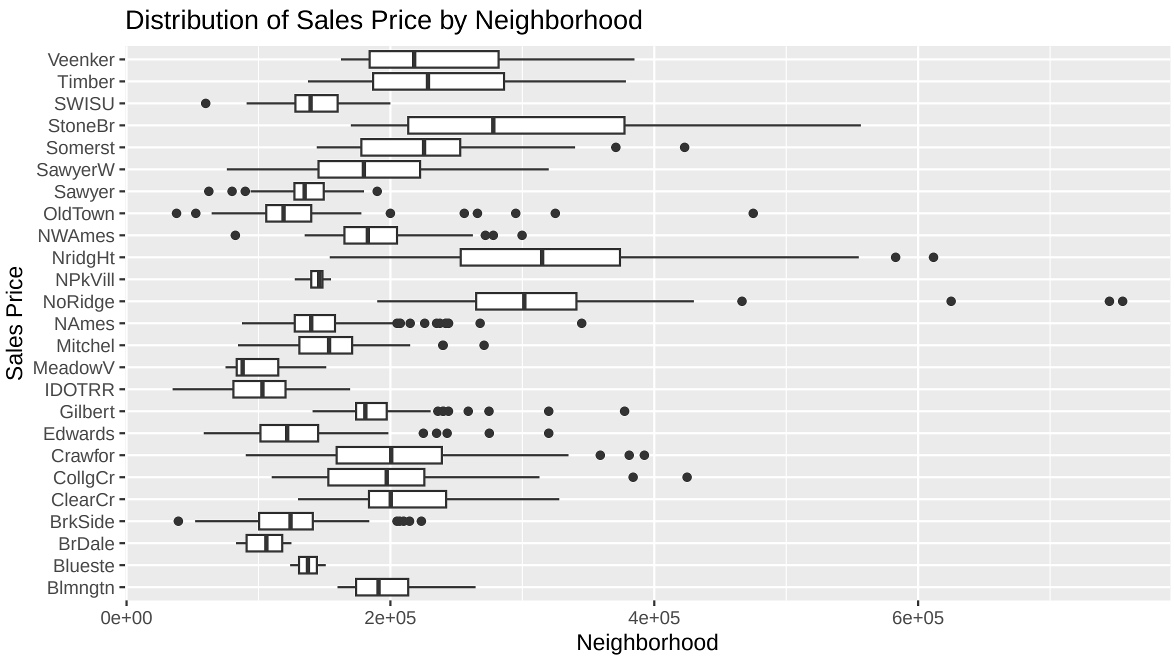
```
# distribution of sales price by zone
ggplot(data = training, aes(x = MSZoning, y = SalePrice)) +
  geom_boxplot() +
  labs(title = "Distribution of Sales Price by Zone",
       x = "Zone",
       y = "Sales Price")
```



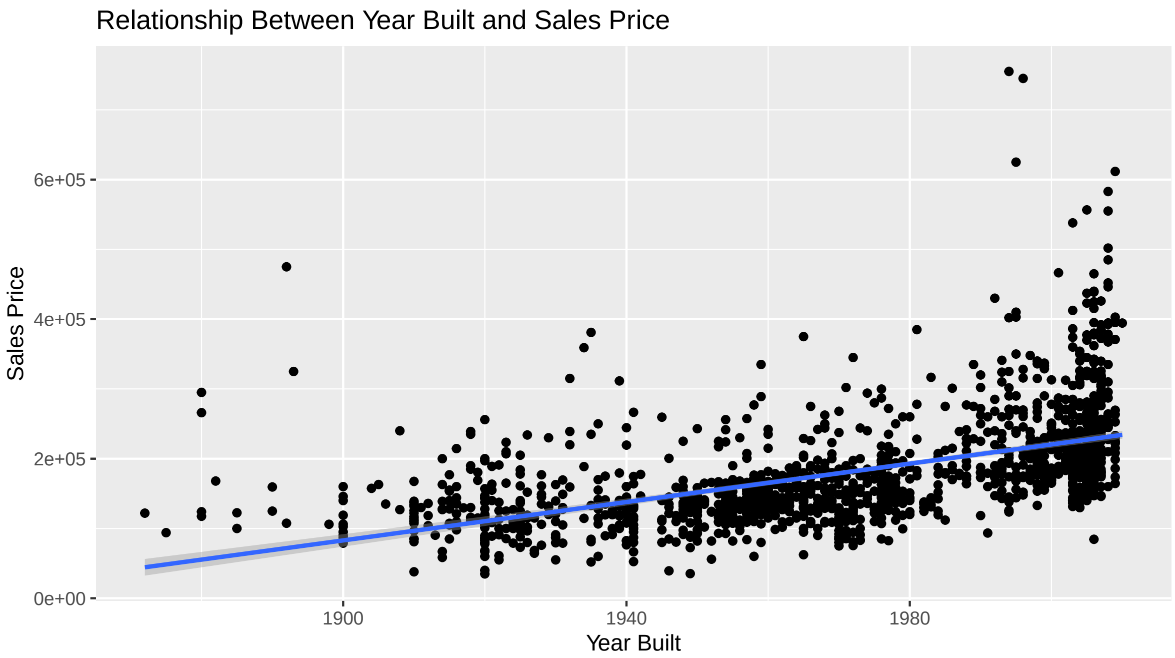
```
# distribution of sales price by utility type
ggplot(data = training, aes(x = Utilities, y = SalePrice)) +
  geom_boxplot() +
  labs(title = "Distribution of Sales Price by Utility Type",
       x = "Utility Type",
       y = "Sales Price")
```



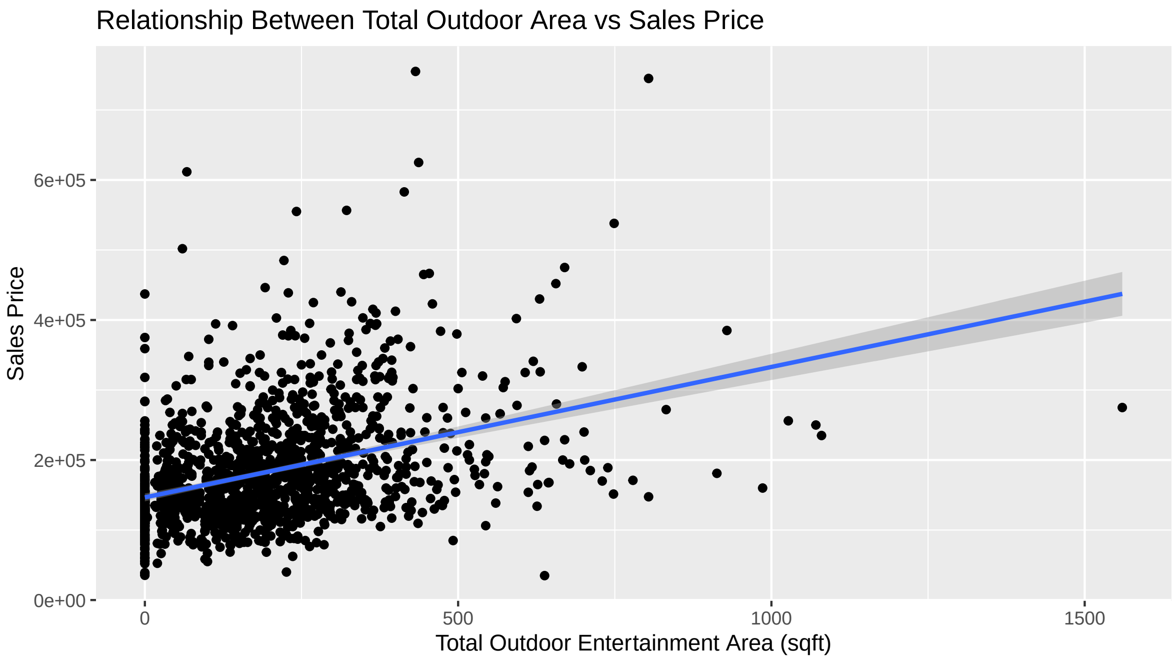
```r
# distribution of sales price by neighborhood
ggplot(data = training, aes(x = SalePrice, y = Neighborhood)) +
  geom_boxplot() +
  labs(title = "Distribution of Sales Price by Neighborhood",
       x = "Neighborhood",
       y = "Sales Price")
```



```
# scatterplot of year built vs. sales price
ggplot(data = training, aes(x = YearBuilt, y = SalePrice)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "Relationship Between Year Built and Sales Price",
       x = "Year Built",
       y = "Sales Price")
```



```
# rename three season porch area variable
training <- training %>%
  rename(threeSeasonPorch = '3SsnPorch')

# add up all outdoor area spaces
training <- training %>%
  mutate(outdoorEntArea = WoodDeckSF + OpenPorchSF + EnclosedPorch + threeSeasonPorch + Sc

# scatterplot of outdoorEntArea vs sales price
ggplot(data = training, aes(x = outdoorEntArea, y = SalePrice)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "Relationship Between Total Outdoor Area vs Sales Price",
       x = "Total Outdoor Entertainment Area (sqft)",
       y = "Sales Price")
```



2.2 Other variables considered

The following variables are excluded from the model because

- Continuous variables: It does not contain a sufficient amount of data or the correlation with the sales price is relatively weaker compared to those selected.
- Categorical variables: It does not contain a sufficient amount of data or the distribution of sales price across different categories are generally similar to each other.

Those variables are:

- PoolArea: There is no clear correlation between pool area and sales price. There is a large cluster of points with a large range of sales prices for homes without pools. Besides that section, there is not enough data to make a conclusion on its correlation.
- WoodDeckSF: Even though there is a positive correlation, the data begins to fan out as the area of the wood deck increases.
- BsmtFinType1: Many of the basement finish types have similar distributions, which shows that the differences in basement finishes do not contribute to significant differences in sales price.
- LotConfig: Many of the lot configurations have similar distributions, which shows that the differences in lot configurations do not contribute to significant differences in sales price.

- BldgType: Although 1Fam has a significantly higher sales price than the other building types, the rest have very similar distributions. This shows that the differences in building types do not contribute to significant differences in sales price.
- LotArea: A significant positive or negative relationship does not exist for this variable, as there is a large range in sales prices for houses with a smaller lot area with a few outliers for homes with larger lot areas.

---

3 Modeling

3.1 Model fit

Fit your ultimate multiple regression model using `lm()` & save it in `SalePrice_model`:

```
# fit multiple regression model
SalePrice_model <- lm(SalePrice ~ totalSF + MSZoning + Utilities + Neighborhood + YearBuil
```

3.2 Compute score on training data

- Apply the fitted/trained model `SalePrice_model` to the training data to get $\hat{y}$
- Compute the root mean squared error using `dplyr` and other R functions. In other words, do not use a `rmse()` function from another R package.
- Ensure my score displays in the HTML output

```
# add predicted values to training dataset
training <- data_frame(training, y_hat = predict(SalePrice_model))

# calculate rmsle
rmsle <- sqrt((sum((log(training$y_hat + 1) - log(training$SalePrice + 1))^2))/(nrow(train

print(rmsle)
```

```
[1] 0.1770402
```

3.3 Apply fitted model on test data

- Apply the fitted/trained model `SalePrice_model` to the test data to get $\hat{y}$

```
# rename 1st floor area column
test <- test %>%
  rename(firstFloorSF = '1stFlrSF')

# rename 2nd floor area column
test <- test %>%
  rename(secondFloorSF = '2ndFlrSF')

# add up areas from first floor, second floor, and basement to get total square feet
test <- test %>%
  mutate(totalSF = firstFloorSF + secondFloorSF + TotalBsmtSF)

# rename three season porch area variable
test <- test %>%
  rename(threeSeasonPorch = '3SsnPorch')

# add up all outdoor area spaces
test <- test %>%
  mutate(outdoorEntArea = WoodDeckSF + OpenPorchSF + EnclosedPorch + threeSeasonPorch + Sc
```

```
# predict sales price on test using fitted/trained model
test$SalePrice <- predict(SalePrice_model, newdata = test)
```

```
# filter test to only include non NA sale price values
test_salePrice_NA <- test %>%
  filter(!is.na(SalePrice))

# find mean of all sales prices of test
test_salePrice_mean <- mean(test_salePrice_NA$SalePrice)

# replace NA values with mean
test$SalePrice[is.na(test$SalePrice)] <- test_salePrice_mean
```

---

4 Kaggle score

4.1 Create submission CSV

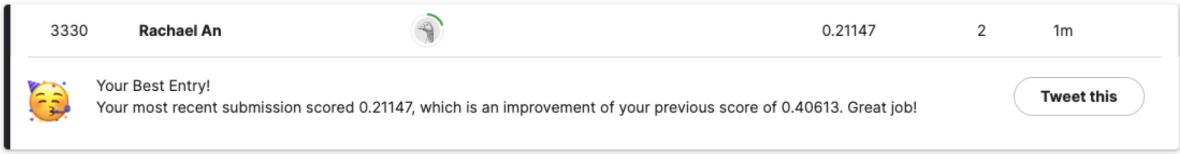
Below is code that writes predictions based on the mean model to a csv file:

$$\hat{y} = \hat{f}(\vec{x}) = \overline{y}$$

Modify this code to submits the model's predictions.

```
# add sales price values to submission file
submission <- sample_submission %>%
  mutate(SalePrice = test$SalePrice)

# convert into file
write_csv(submission, path = "data/submission.csv")
```

4.2 Screenshot of Kaggle score



3330 **Rachael An** 0.21147 2 1m

🥳 Your Best Entry!
Your most recent submission scored 0.21147, which is an improvement of your previous score of 0.40613. Great job!