

Regression Study Code File

2025-12-19

```
library(gt)
library(dplyr)
library(tidyr)
library(corrplot)
library(car)
library(MASS)
library(lmtest)
library(ggplot2)
library(caret)
```

```
file_path <- 'project_data.csv'
df <- read.csv(file_path, header=TRUE)
str(df)
```

```
'data.frame': 182 obs. of 7 variables:
 $ zip_code      : int 10001 10002 10003 10004 10005 10006 10007 10009 10010 10011 ...
 $ inspections    : int 182 3922 2576 18 27 5 133 3296 93 1208 ...
 $ population     : int 27004 76518 53877 4579 8801 3736 7506 58418 32410 50772 ...
 $ has_garage     : int 0 1 0 0 0 0 0 0 0 ...
 $ has_dropoff    : int 1 1 1 0 0 1 0 1 1 1 ...
 $ litter_basket_count: int 76 261 327 42 17 46 59 193 234 195 ...
 $ total_park_acres : num 14.561 90.192 13.712 23.624 0.117 ...
```

```
df <- df %>%
  rename(pop = population, garage = has_garage, dropoff = has_dropoff,
         baskets = litter_basket_count, park_acres = total_park_acres
  )
```

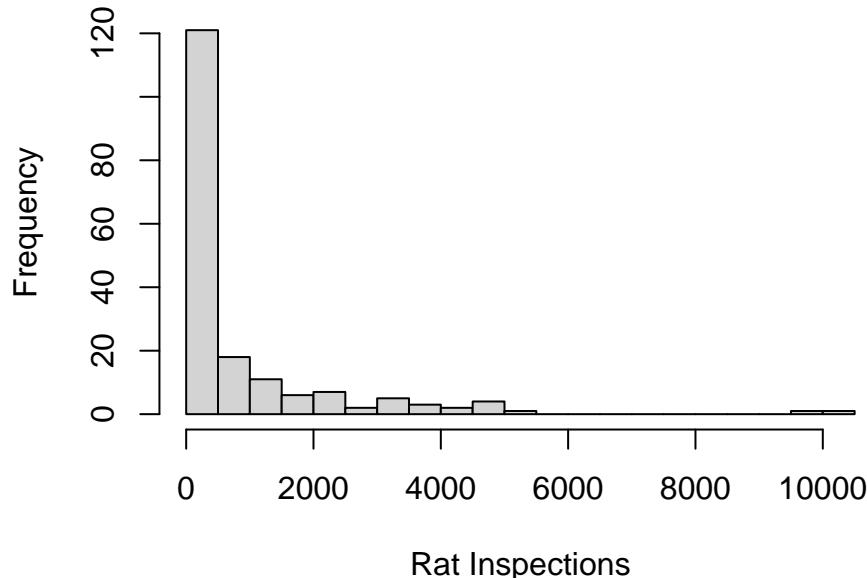
```
# ----- Summary Statistics of Continuous Variables ----- #
df %>%
  summarise(across(c(inspections, pop, baskets, park_acres),
    list(Median = median, Mean = mean, SD = sd, Min = min, Max = max))) %>%
  pivot_longer(everything(), names_to = c("Variable", "Statistic"),
    names_pattern = "(.+)_(`w+)$", values_to = "Value") %>%
  pivot_wider(names_from = Statistic, values_from = Value) %>%
  gt() %>%
  tab_header(title = "Fig. 6: Descriptive Statistics") %>%
  fmt_number(columns = c(Median, Mean, SD, Min, Max), decimals = 2)
```

Fig. 6: Descriptive Statistics

Variable	Median	Mean	SD	Min	Max
inspections	183.00	855.35	1,527.77	1.00	10,373.00
pop	43,165.50	47,196.59	27,434.76	0.00	112,750.00
baskets	103.50	122.38	86.90	0.00	447.00
park_acres	105.24	464.02	784.66	0.00	5,558.26

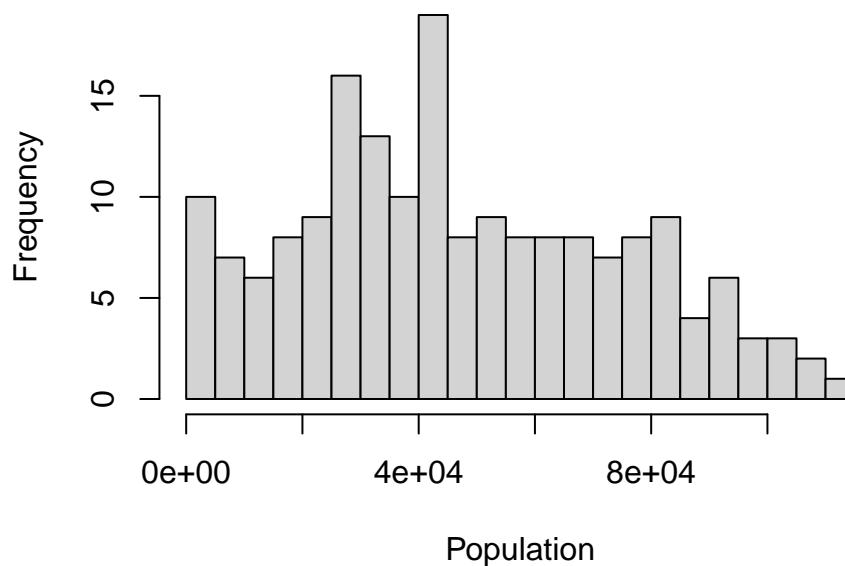
```
# ----- Exploratory Data Analysis ----- #
hist(df$inspections, main = "Fig. 2: Rat Inspections by Zip",
     xlab = "Rat Inspections", ylab = "Frequency", breaks = 20, font.main = 1)
```

Fig. 2: Rat Inspections by Zip



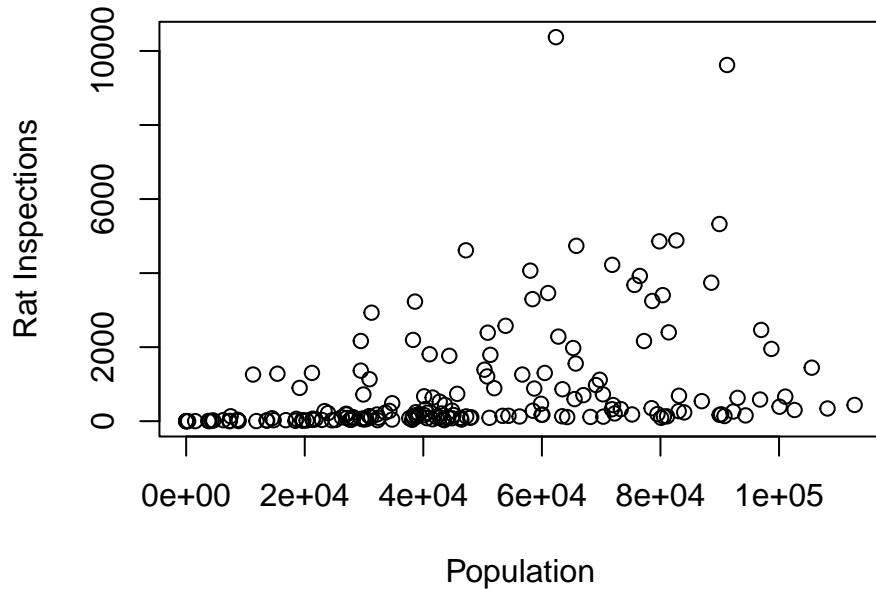
```
hist(df$pop, main = "Fig. 3: Population by Zip",
     xlab = "Population", ylab = "Frequency", breaks = 20, font.main = 1)
```

Fig. 3: Population by Zip



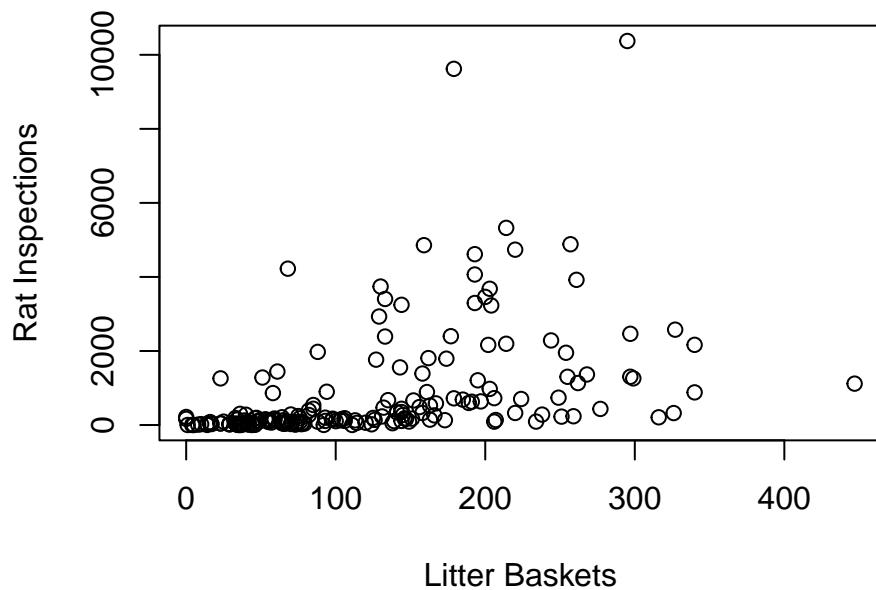
```
plot(df$pop, df$inspections, xlab='Population', ylab='Rat Inspections',
     main='Fig. 4: Population vs. Inspections by Zip', font.main = 1)
```

Fig. 4: Population vs. Inspections by Zip



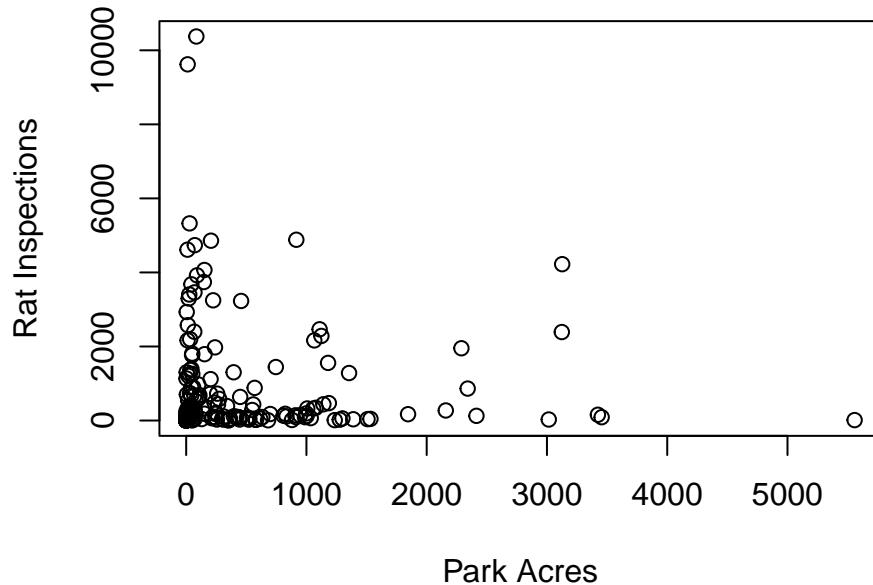
```
plot(df$baskets, df$inspections, xlab='Litter Baskets', ylab='Rat Inspections',
     main='Fig. 5: Litter Baskets vs. Inspections by Zip', font.main = 1)
```

Fig. 5: Litter Baskets vs. Inspections by Zip



```
plot(df$park_acres, df$inspections, xlab='Park Acres', ylab='Rat Inspections',
  main='Fig. 6: Park Acres vs. Inspections by Zip', font.main = 1)
```

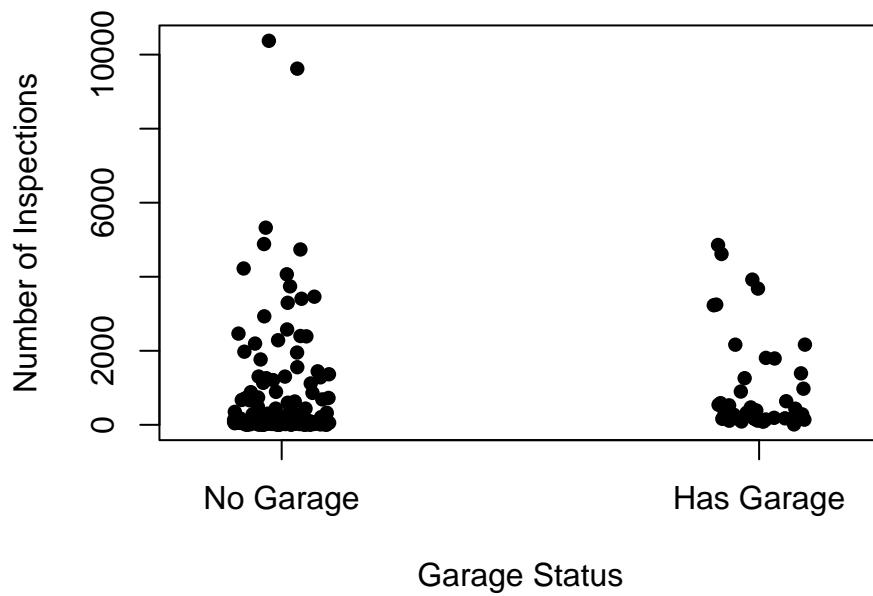
Fig. 6: Park Acres vs. Inspections by Zip



```
df$has_garage_label <- factor(df$garage, levels = c(0, 1),
  labels = c("No Garage", "Has Garage"))

stripchart(inspections ~ has_garage_label, vertical=TRUE, method='jitter', pch=16,
  data = df, main = "Fig. 7: Inspections by Garage Presence", font.main = 1,
  xlab = "Garage Status", ylab = "Number of Inspections")
```

Fig. 7: Inspections by Garage Presence



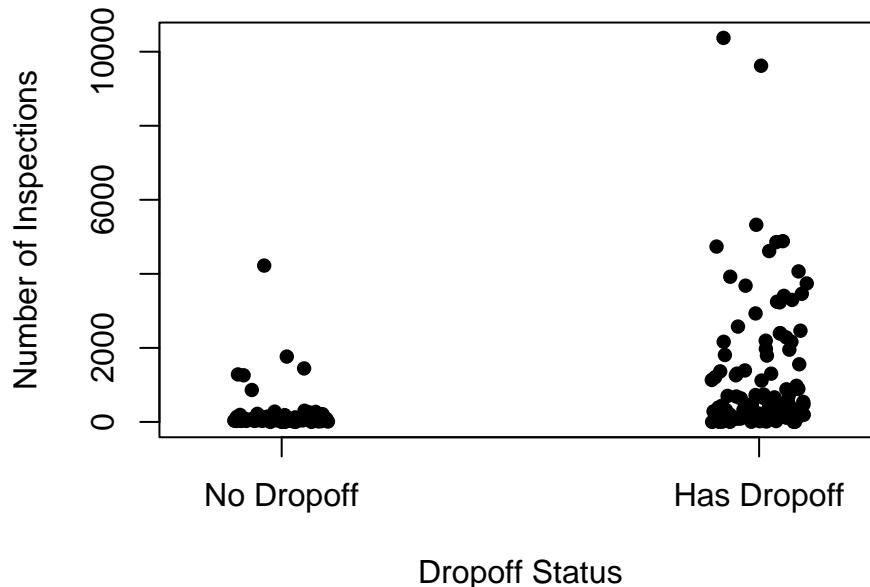
```

df$has_dropoff_label <- factor(df$dropoff, levels = c(0, 1),
                                labels = c("No Dropoff", "Has Dropoff"))

stripchart(inspections ~ has_dropoff_label, vertical=TRUE, method='jitter', pch=16,
           data = df, main = "Fig. 8: Inspections by Dropoff Presence", font.main = 1,
           xlab = "Dropoff Status", ylab = "Number of Inspections")

```

Fig. 8: Inspections by Dropoff Presence



```

# ----- Initial Regression Model ----- #
model <- lm(inspections ~ pop + garage + dropoff + baskets + park_acres, data=df)
summary(model)

```

```

Call:
lm(formula = inspections ~ pop + garage + dropoff + baskets +
    park_acres, data = df)

Residuals:
    Min      1Q      Median      3Q      Max 
-1997.8  -633.4   -234.7   241.0  8250.1 

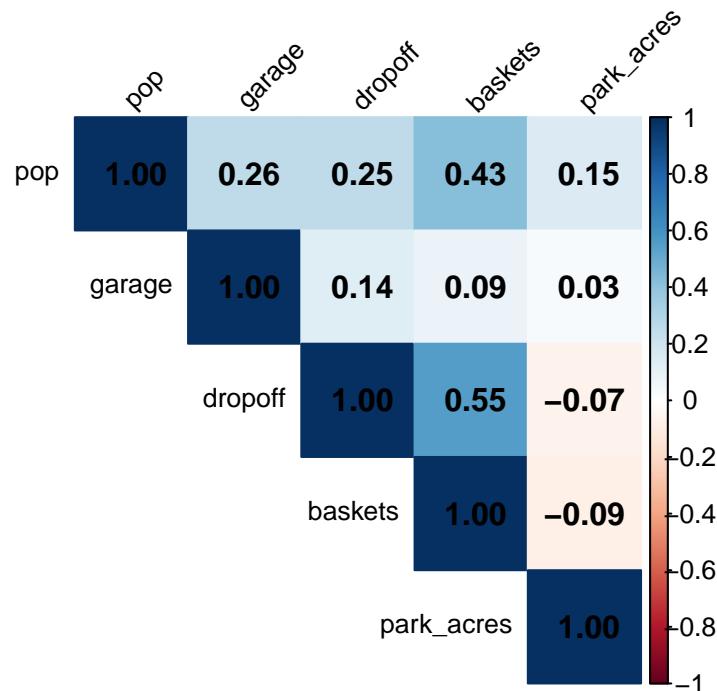
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -4.737e+02  2.283e+02  -2.075 0.039485 *  
pop         1.221e-02  4.273e-03   2.857 0.004797 ** 
garage      -1.085e+02  2.474e+02  -0.439 0.661379    
dropoff     2.390e+02  2.520e+02   0.949 0.344166    
baskets      5.436e+00  1.503e+00   3.618 0.000388 ***  
park_acres  -8.829e-02  1.315e-01  -0.671 0.502983    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1352 on 176 degrees of freedom
Multiple R-squared:  0.2389,    Adjusted R-squared:  0.2173 
F-statistic: 11.05 on 5 and 176 DF,  p-value: 2.869e-09

```

```
# ----- Correlation Matrix ----- #
X <- model.matrix(model) [, -1]
cor_matrix <- cor(X)
corrplot(cor_matrix, method = "color", type = "upper", font.main = 1,
  addCoef.col = "black", tl.col = "black", tl.cex = 0.8, tl.srt = 45,
  title = "Fig. 5: Correlation Matrix of Predictor Variables", mar = c(0, 0, 2, 0))
```

Fig. 5: Correlation Matrix of Predictor Variables



```
# ----- Residual Analysis of Initial Model ----- #
stud_res <- rstudent(model)
y_hat <- fitted(model)

par(mfrow = c(2, 3), oma = c(0, 0, 2, 0))

qqnorm(stud_res, main = 'Quantile Plot')
qqline(stud_res, col='black')

hist(stud_res, main = 'Histogram',
     xlab = 'Studentized Deleted Residuals', breaks = 10)

plot(y_hat, stud_res, main = 'Scatter Plot',
      xlab = 'Y Values', ylab = 'Studentized Deleted Residuals')
abline(h=0, col='black', lty=2)

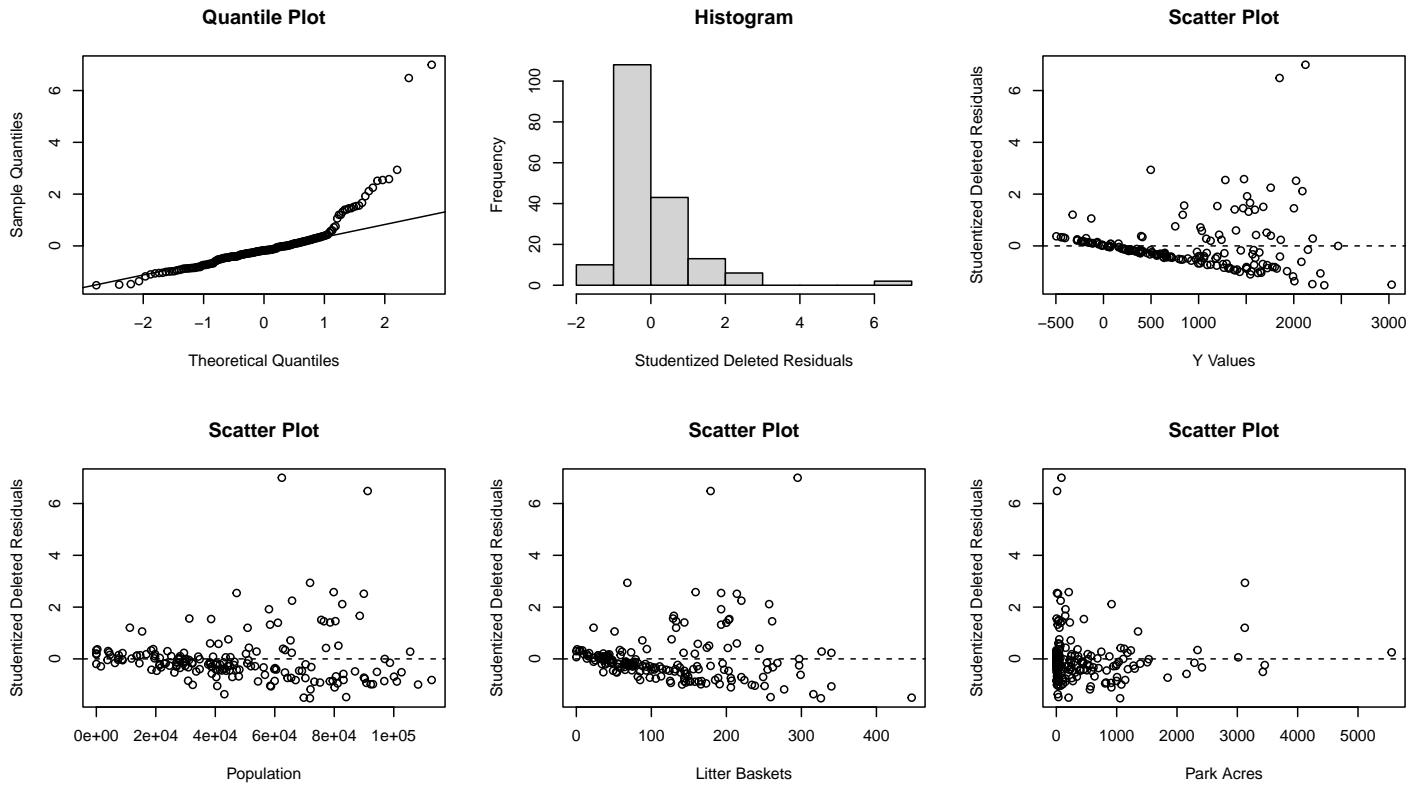
plot(df$pop, stud_res, main = 'Scatter Plot',
      xlab = 'Population', ylab = 'Studentized Deleted Residuals')
abline(h=0, col='black', lty=2)

plot(df$baskets, stud_res, main = 'Scatter Plot',
      xlab = 'Litter Baskets', ylab = 'Studentized Deleted Residuals')
abline(h=0, col='black', lty=2)

plot(df$park_acres, stud_res, main = 'Scatter Plot',
      xlab = 'Park Acres', ylab = 'Studentized Deleted Residuals')
abline(h=0, col='black', lty=2)

mtext("Fig. 10: Residual Plots", outer = TRUE, cex = 1.75)
```

Fig. 10: Residual Plots



```

par(mfrow = c(1, 1))

# ----- Test for Homogeneity of Variance ----- #
bptest(model, studentize = TRUE)

studentized Breusch-Pagan test

data: model
BP = 14.859, df = 5, p-value = 0.01098

# ----- Test for Normality ----- #
shapiro.test(rstudent(model))

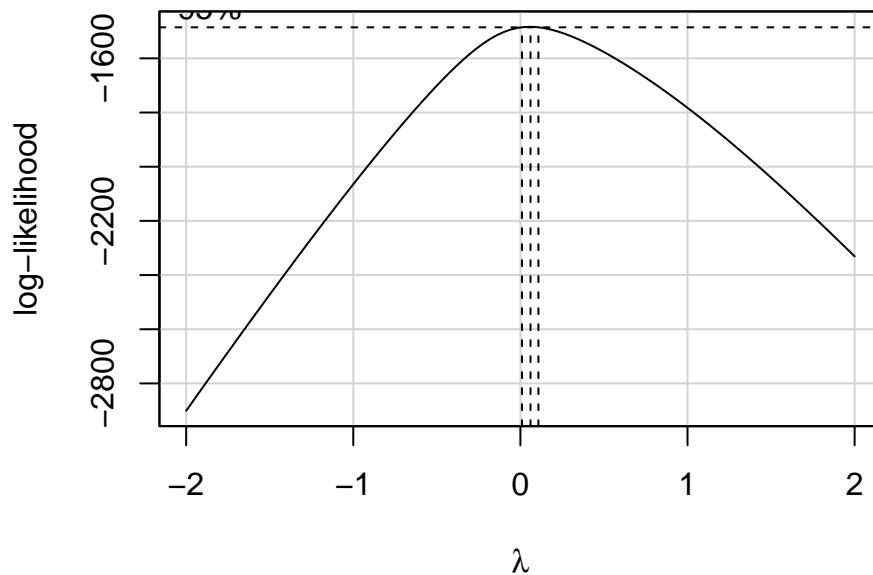
Shapiro-Wilk normality test

data: rstudent(model)
W = 0.71671, p-value < 2.2e-16

# ----- Box Cox Transformation ----- #
result <- boxCox(model, main = "Fig. 11: Box-Cox Transformation", font.main = 1)

```

Fig. 11: Box–Cox Transformation



```
lambda <- result$x[which.max(result$y)]
lambda
```

```
[1] 0.06060606
```

```
# ----- Taking the Log of Inspections ----- #
any(df$inspections <= 0)
```

```
[1] FALSE
```

```
df$log_inspections <- log(df$inspections)

log_model <- lm(log_inspections ~ pop + garage + dropoff + baskets + park_acres, data=df)
summary(log_model)
```

Call:
`lm(formula = log_inspections ~ pop + garage + dropoff + baskets + park_acres, data = df)`

Residuals:

Min	1Q	Median	3Q	Max
-3.3802	-0.7866	-0.0315	0.8773	3.9197

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.471e+00	2.238e-01	11.042	< 2e-16 ***
pop	3.156e-05	4.188e-06	7.537	2.45e-12 ***
garage	2.913e-01	2.425e-01	1.202	0.231
dropoff	3.664e-01	2.470e-01	1.484	0.140
baskets	8.949e-03	1.473e-03	6.076	7.41e-09 ***
park_acres	-1.297e-04	1.289e-04	-1.006	0.316

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Residual standard error: 1.325 on 176 degrees of freedom
Multiple R-squared:  0.5724,    Adjusted R-squared:  0.5602
F-statistic: 47.11 on 5 and 176 DF,  p-value: < 2.2e-16

```

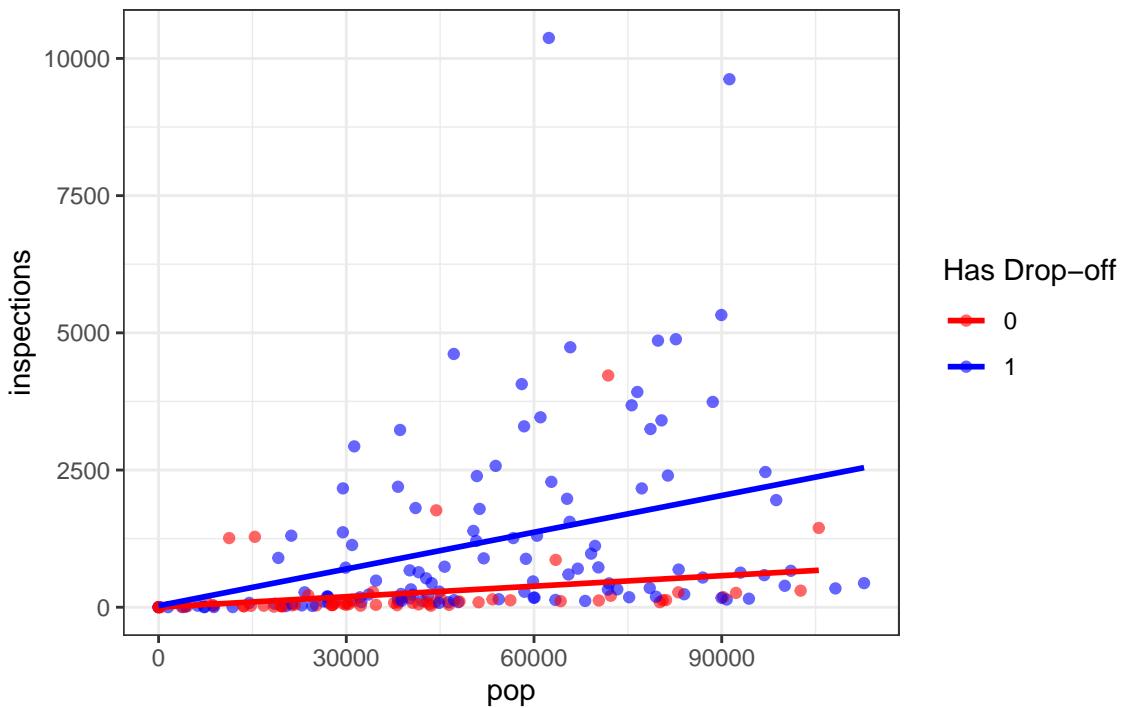
```

# ----- Examining Interaction Effects -----
ggplot(df, aes(x = pop, y = inspections, color = factor(dropoff))) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(color = "Has Drop-off",
       title = "Fig. 12: Population vs. Inspections by Drop-off Presence") +
  theme_bw() + theme(plot.title = element_text(hjust = 0.5, size = 11)) +
  scale_color_manual(values = c("red", "blue"))

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

```

Fig. 12: Population vs. Inspections by Drop-off Presence



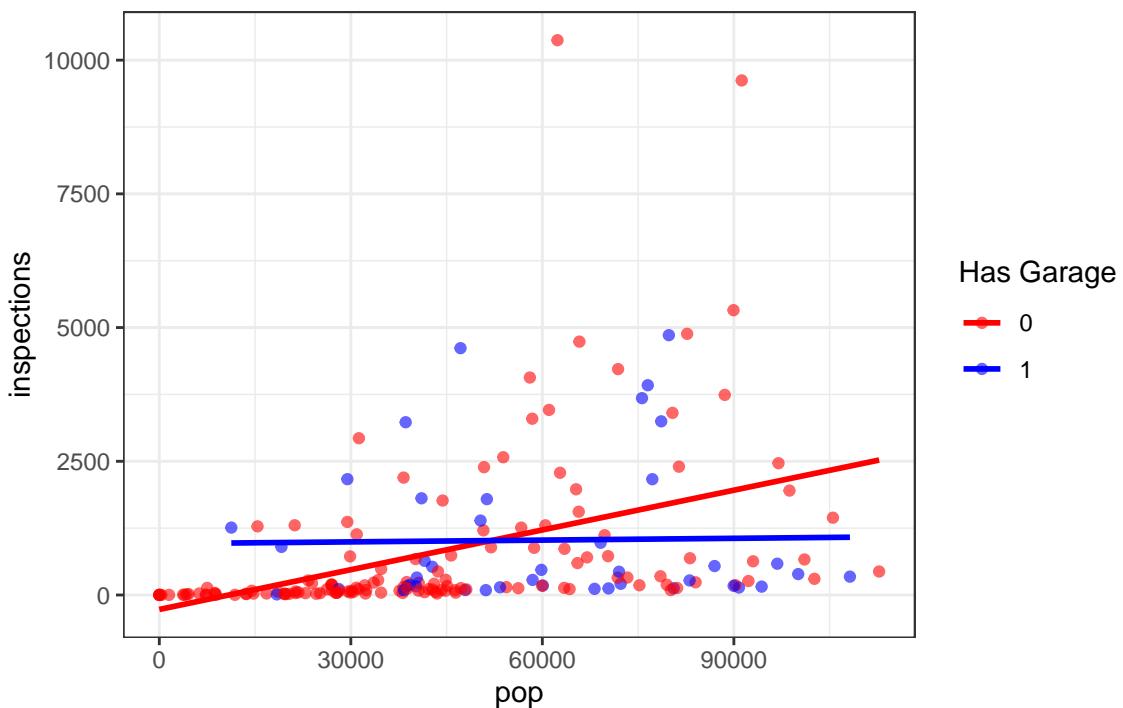
```

ggplot(df, aes(x = pop, y = inspections, color = factor(garage))) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(
    color = "Has Garage",
    title = "Fig. 13: Population vs. Inspections by Garage Presence") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5, size = 11)) +
  scale_color_manual(values = c("red", "blue"))

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

```

Fig. 13: Population vs. Inspections by Garage Presence



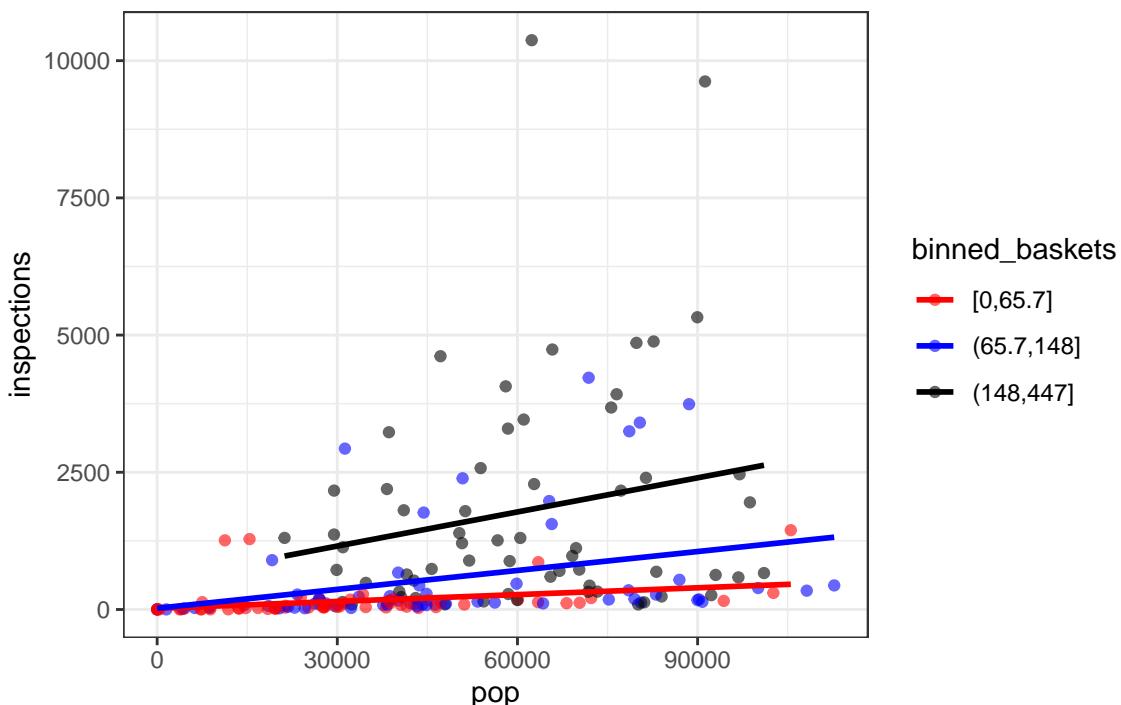
```

df$binned_baskets = cut(df$baskets,
  breaks = quantile(df$baskets, probs = c(0, .33, .66, 1)),
  include.lowest = TRUE
)
ggplot(df, aes(x = pop, y = inspections, color = binned_baskets)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(
    title = "Fig. 14: Population vs. Inspections by Baskets") + theme_bw() +
  theme(plot.title = element_text(hjust = 0.5, size = 11)) +
  scale_color_manual(values = c("red", "blue", "black"))

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

```

Fig. 14: Population vs. Inspections by Baskets



```
# ----- Centered Model with Interactions ----- #
df$pop_c <- scale(df$pop, scale = FALSE)
df$baskets_c <- scale(df$baskets, scale = FALSE)

centered_log_int_model <- lm(log_inspections ~ pop_c*baskets_c + pop_c*garage + pop_c*dropoff, data = df)
summary(centered_log_int_model)
```

Call:
`lm(formula = log_inspections ~ pop_c * baskets_c + pop_c * garage + pop_c * dropoff, data = df)`

Residuals:

Min	1Q	Median	3Q	Max
-2.6838	-0.8749	-0.0424	0.7894	3.8343

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	5.181e+00	1.897e-01	27.314	< 2e-16 ***
pop_c	2.275e-05	6.835e-06	3.329	0.00106 **
baskets_c	8.746e-03	1.392e-03	6.285	2.55e-09 ***
garage	4.703e-01	2.410e-01	1.951	0.05267 .
dropoff	4.400e-01	2.304e-01	1.910	0.05782 .
pop_c:baskets_c	-2.247e-07	5.345e-08	-4.204	4.18e-05 ***
pop_c:garage	-3.992e-05	9.040e-06	-4.416	1.76e-05 ***
pop_c:dropoff	2.184e-05	8.166e-06	2.675	0.00818 **

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.215 on 174 degrees of freedom
Multiple R-squared: 0.6445, Adjusted R-squared: 0.6302
F-statistic: 45.06 on 7 and 174 DF, p-value: < 2.2e-16

```
vif_values <- vif(centered_log_int_model)
```

there are higher-order terms (interactions) in this model
consider setting type = 'predictor'; see ?vif

```

mean(vif_values)

[1] 2.241026

# ----- ANOVA ----- #
anova(log_model, centered_log_int_model)

Analysis of Variance Table

Model 1: log_inspections ~ pop + garage + dropoff + baskets + park_acres
Model 2: log_inspections ~ pop_c * baskets_c + pop_c * garage + pop_c *
          dropoff
Res.Df   RSS Df Sum of Sq    F    Pr(>F)
1     176 308.86
2     174 256.76  2     52.099 17.653 1.046e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# ----- Residual Analysis for Final Model ----- #
final_stud_res <- rstudent(centered_log_int_model)
final_y_hat <- fitted(centered_log_int_model)

par(mfrow = c(2, 3), oma = c(0, 0, 2, 0))

qqnorm(final_stud_res, main = 'Quantile Plot')
qline(final_stud_res, col='black')

hist(final_stud_res, main = 'Histogram',
      xlab = 'Studentized Deleted Residuals', breaks = 10)

plot(final_stud_res, type = 'l', main = 'Line Plot',
      ylab = 'Studentized Deleted Residuals', col='red')
points(final_stud_res, col = 'black')
abline(h=0, col='black', lty=2)

plot(final_y_hat, stud_res, main = 'Scatter Plot',
      xlab = 'Y Values', ylab = 'Studentized Deleted Residuals')
abline(h=0, col='black', lty=2)

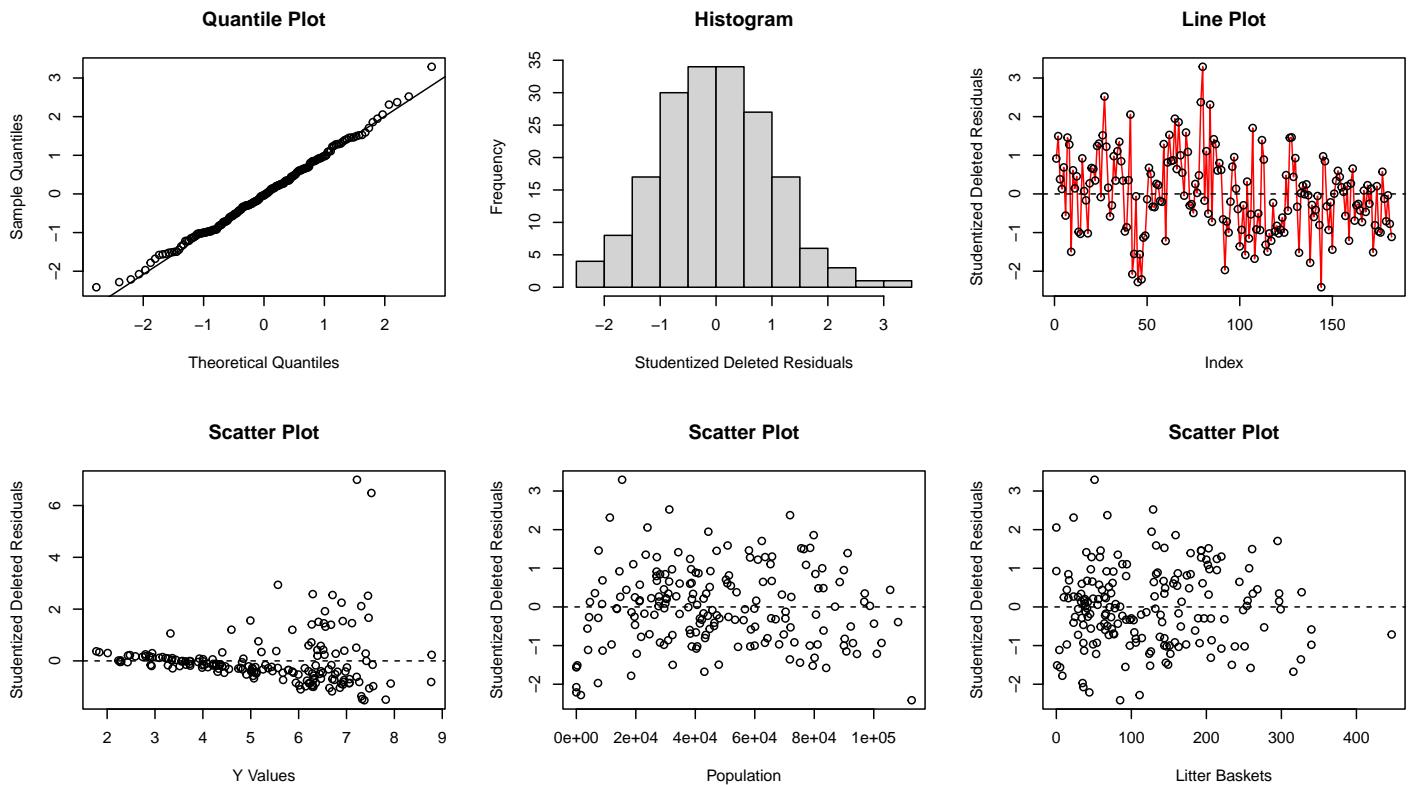
plot(df$pop, final_stud_res, main = 'Scatter Plot',
      xlab = 'Population', ylab = 'Studentized Deleted Residuals')
abline(h=0, col='black', lty=2)

plot(df$baskets, final_stud_res, main = 'Scatter Plot',
      xlab = 'Litter Baskets', ylab = 'Studentized Deleted Residuals')
abline(h=0, col='black', lty=2)

mtext("Fig. 15: Residual Plots", outer = TRUE, cex = 1.75)

```

Fig. 15: Residual Plots



```
par(mfrow = c(1, 1))

# ----- Test for Homogeneity of Variance ----- #
bptest(centered_log_int_model, studentize = TRUE)
```

studentized Breusch-Pagan test

```
data: centered_log_int_model
BP = 6.0799, df = 7, p-value = 0.5304
```

```
# ----- Test for Normality ----- #
shapiro.test(rstudent(centered_log_int_model))
```

Shapiro-Wilk normality test

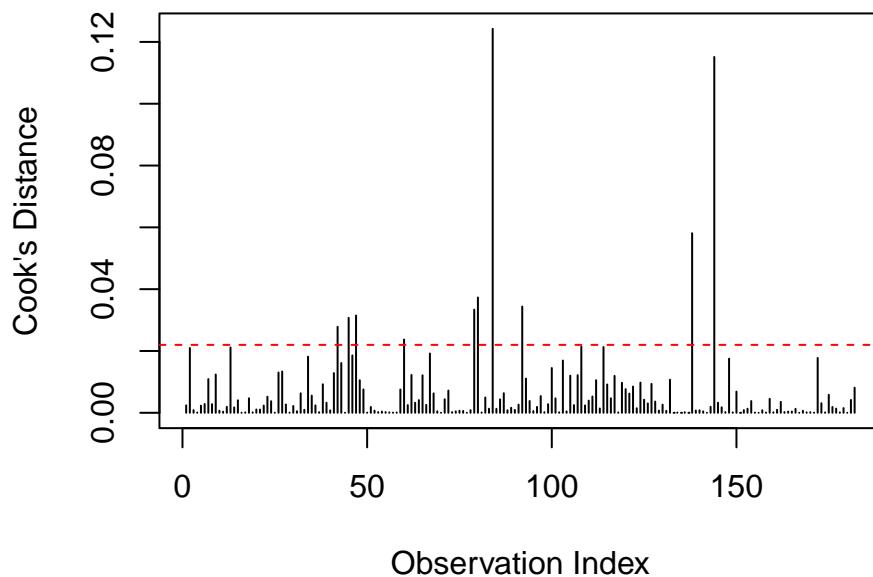
```
data: rstudent(centered_log_int_model)
W = 0.99487, p-value = 0.7877
```

```
# ----- Identifying Influential Cases ----- #
cooks_d <- cooks.distance(centered_log_int_model)

n <- nrow(df)
threshold <- 4 / n
influential <- which(cooks_d > threshold)

plot(cooks_d, type = "h", main = "Fig. 16: Cook's Distance",
     ylab = "Cook's Distance", xlab = "Observation Index", font.main = 1)
abline(h = 4/n, col = "red", lty = 2)
```

Fig. 16: Cook's Distance



```
print(paste("Number of influential observations:", length(influential)))
```

```
[1] "Number of influential observations: 10"
```

```
print(paste("Threshold:", round(threshold, 4)))
```

```
[1] "Threshold: 0.022"
```

```
print(paste("Max Cook's Distance:", round(max(cooks_d), 4)))
```

```
[1] "Max Cook's Distance: 0.1242"
```

```
# ----- K-Fold Cross Validation ----- #
train_control <- trainControl(method = "cv", number = 10)

cv_model <- train(log_inspections ~ pop_c*baskets_c + pop_c*garage + pop_c*dropoff,
  data = df, method = "lm", trControl = train_control)

print(cv_model)
```

Linear Regression

```
182 samples
 4 predictor
```

```
No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 164, 164, 162, 166, 165, 164, ...
Resampling results:
```

RMSE	Rsquared	MAE
1.237239	0.6126059	1.004847

```
Tuning parameter 'intercept' was held constant at a value of TRUE
```

--- Standard Model ---

R-squared: 0.6444944

RMSE: 1.18775

--- Cross-Validation ---

R-squared: 0.6126059

RMSE: 1.237239