Homework 2

2023-09-19

Problem 1

To avoid an error, dummy variable trap we keep one category as a reference group, remaining ones represent the change from the reference. That is why we will have n-1 number of dummy variables for independent variable with categories.

Problem 2

x is a dummy variable changing values between 0 and 1. $\beta 1$ represents the change in the predicted value of y E(log(y)) when x changes.

Problem 3

```
# Loading the data
data <- read.csv("houseprice.csv")
#str(data)

# Fitting the regression model
model <- lm(price ~ sqft + age + sqft:age, data=data)

# Summarizing the model
summary(model)
```

```
##
## Call:
## lm(formula = price ~ sqft + age + sqft:age, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -370939  -33186   -6282   22670  890041
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.048e+05  9.503e+03 -11.033  < 2e-16 ***
## sqft         1.170e+02  3.615e+00  32.366  < 2e-16 ***
## age          2.197e+03  3.431e+02   6.404 2.25e-10 ***
## sqft:age    -1.285e+00  1.372e-01  -9.366  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 75820 on 1076 degrees of freedom
## Multiple R-squared:  0.6205, Adjusted R-squared:  0.6195
## F-statistic: 586.5 on 3 and 1076 DF,  p-value: < 2.2e-16
```

```r
# Extracting the coefficient for the interaction term
gamma <- coef(model)["sqft:age"]

# Print the estimated value of gamma
cat("The estimated value of gamma is:", gamma)
```

```
## The estimated value of gamma is: -1.285041
```

Gamma γ (interaction term coefficient) represents the effect of the interaction between the size of the house and its age on the selling price.

The estimated value of gamma $\hat{\gamma}$ means that when we change the age by one unit and keep all the other variables constant, it shows the relationship between sqft and price.

```r
# Finding the minimum and maximum values
min_sqft <- min(data$sqft)
max_sqft <- max(data$sqft)

# Calculating the marginal effect for the smallest house
marginal_effect_1 <- coef(model)["age"] + gamma * min_sqft

# Calculate the marginal effect for the largest house
marginal_effect_2 <- coef(model)["age"] + gamma * max_sqft

# Print the results
cat("Marginal Effect for Smallest House:", marginal_effect_1)
```

```
## Marginal Effect for Smallest House: 1346.798
```

```r
cat("Marginal Effect for Largest House:", marginal_effect_2)
```

```
## Marginal Effect for Largest House: -7950.474
```

```r
# Calculating the marginal effect
age <- 20
marginal_effect <- coef(model)["sqft"] + gamma * age

cat("Marginal Effect of sqft on price for a 20-year-old house:", marginal_effect)
```

```
## Marginal Effect of sqft on price for a 20-year-old house: 91.29804
```

```r
#how the price of 20 years old house is changing when 1 additional sqft is added.
```

We can interpret it as the change in the predicted price for a 1-unit increase in sqft while having the age fixed at 20 years.

```r
library(car)
```

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

```r
# Performing the F-test
f_test <- linearHypothesis(model, c("sqft = 0", "age = 0"))
summary(f_test)
```

```
##      Res.Df          RSS                  Df      Sum of Sq
##  Min.   :1076   Min.   :6.186e+12   Min.   :2   Min.   :9.965e+12
##  1st Qu.:1076   1st Qu.:8.677e+12   1st Qu.:2   1st Qu.:9.965e+12
##  Median :1077   Median :1.117e+13   Median :2   Median :9.965e+12
##  Mean   :1077   Mean   :1.117e+13   Mean   :2   Mean   :9.965e+12
##  3rd Qu.:1078   3rd Qu.:1.366e+13   3rd Qu.:2   3rd Qu.:9.965e+12
##  Max.   :1078   Max.   :1.615e+13   Max.   :2   Max.   :9.965e+12
##                                     NA's   :1   NA's   :1
##        F             Pr(>F)
##  Min.   :866.7   Min.   :0
##  1st Qu.:866.7   1st Qu.:0
##  Median :866.7   Median :0
##  Mean   :866.7   Mean   :0
##  3rd Qu.:866.7   3rd Qu.:0
##  Max.   :866.7   Max.   :0
##  NA's   :1       NA's   :1
```

```r
# Extracting the p-value
p <- f_test$`Pr(>F)`

# Setting the significance level
a <- 0.05

# Checking if the p-value is less than the significance level
result <- ifelse(p < a, "Reject", "Fail to reject")
result
```

```
## [1] NA       "Reject"
```

That means at least one of $\beta 2$ or $\beta 3$ is significant.

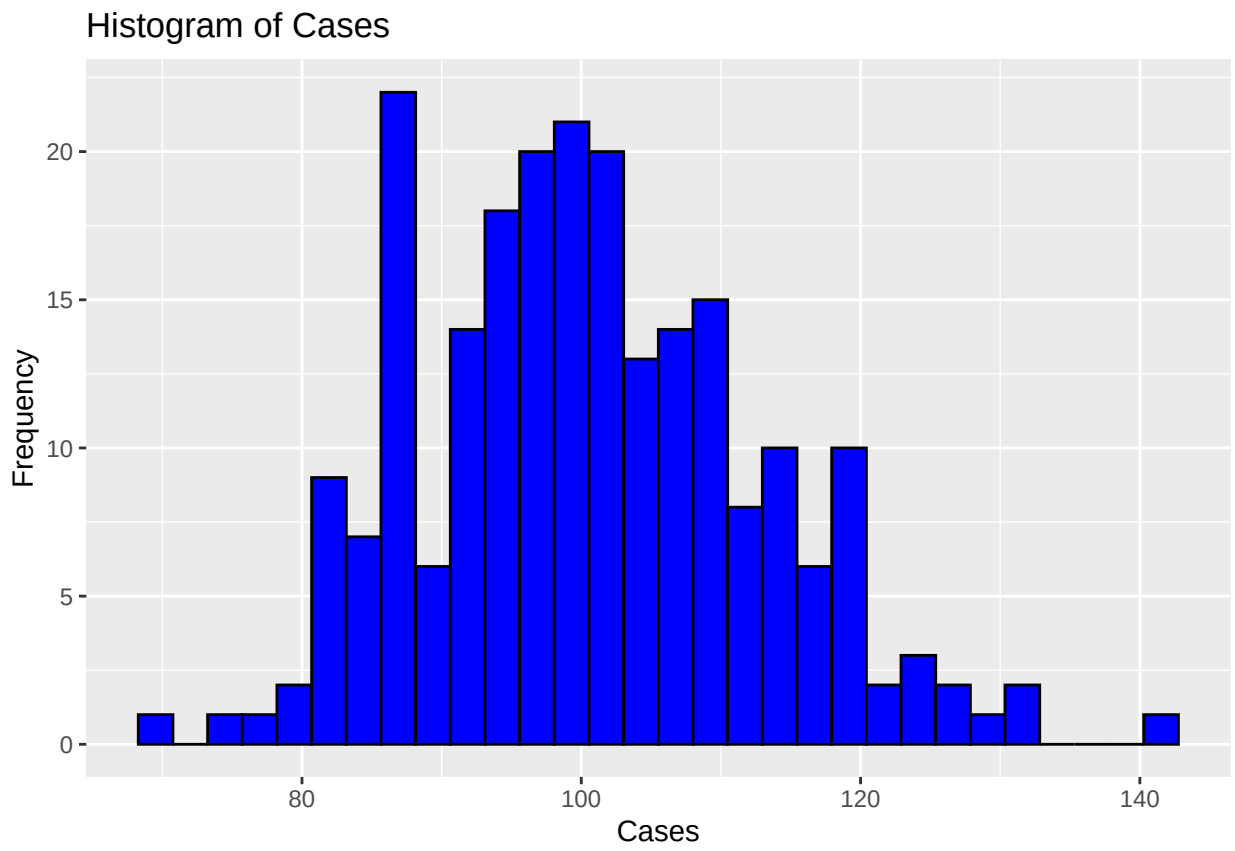
Problem 4

```r
#1
library(ggplot2)

# Load the data
data <- read.csv("fullmoon.csv")

# Examine the histogram of cases
ggplot(data, aes(x = cases)) +
  geom_histogram(fill = "blue", color = "black") +
  labs(title = "Histogram of Cases", x = "Cases", y = "Frequency")
```

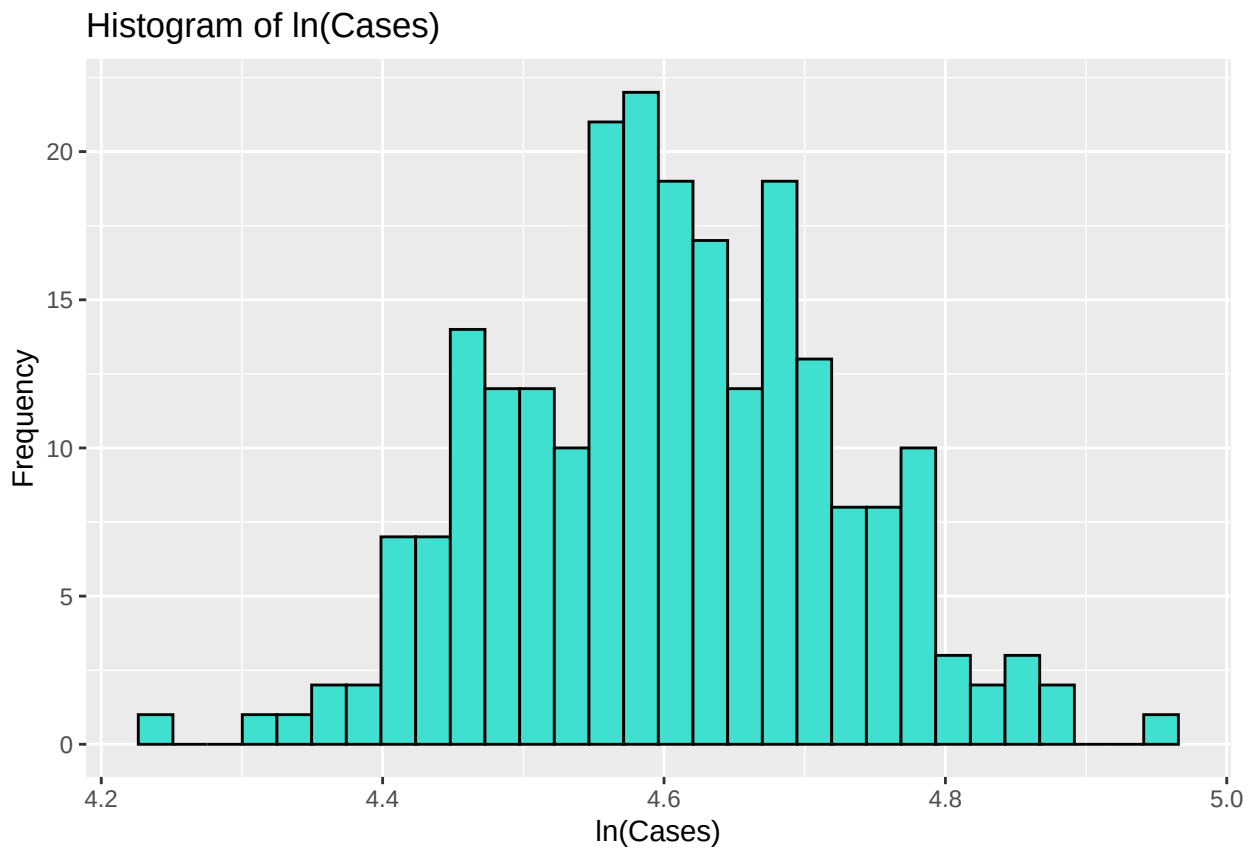
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# Create the variable ln(cases)
data$ln_cases <- log(data$cases)

# Examine the histogram of ln(Cases)
ggplot(data, aes(x = ln_cases)) +
  geom_histogram(fill = "turquoise", color = "black") +
  labs(title = "Histogram of ln(Cases)", x = "ln(Cases)", y = "Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



In the histogram of cases (without logarithm) we see that it is right skewed.

The histogram of cases with logarithm is much closer to Normal distribution.

So, by taking the natural logarithm of a variable we can make the distribution more symmetric and reduce the impact of extreme values.

```
#2
model_1 <- lm(log(cases) ~ time + holiday + friday + saturday + fullmoon + newmoon, data = data)
summary(model_1)
```

```
##
## Call:
## lm(formula = log(cases) ~ time + holiday + friday + saturday +
##     fullmoon + newmoon, data = data)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.312154 -0.069719  0.004129  0.075009  0.307694
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.5347493  0.0154967 292.627  < 2e-16 ***
## time        0.0003488  0.0001099   3.175  0.00171 **
## holiday     0.1321916  0.0640594   2.064  0.04022 *
## friday      0.0683495  0.0209847   3.257  0.00130 **
## saturday    0.1012767  0.0210554   4.810 2.78e-06 ***
```

```
## fullmoon    0.0253671  0.0395669   0.641  0.52211
## newmoon     0.0612188  0.0423098   1.447  0.14933
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1098 on 222 degrees of freedom
## Multiple R-squared:  0.1688, Adjusted R-squared:  0.1463
## F-statistic: 7.513 on 6 and 222 DF,  p-value: 2.414e-07
```

Based on the regression results, it is not possible to conclude that there is a statistically significant effect of a full moon on emergency room cases.

However, we see that the variables time, holiday, friday, and saturday (saturday is highly significant) are statistically significant. The time variable is positively associated with log(cases), meaning that as time increases, log(cases) tends to increase. Holiday, Friday, and Saturday variables are also positively associated with log(cases), indicating that these factors tend to increase log(cases) compared to non-holiday days and other weekdays.

The fullmoon and the newmoon are not statistically significant. That means presence does not have any significant effect on the number of cases.

```
#3
```

The coefficient for the fullmoon is not statistically significant because the p-value = 0.52211. The presence of it does not have any significant effect on the number of cases log(cases) when controlling for other variables in the model.

```
#4
library(car)

# Performing the F-test
f_test <- linearHypothesis(model_1, c("friday = 0", "saturday = 0"))
summary(f_test)
```

```
##      Res.Df           RSS              Df      Sum of Sq            F
##  Min.   :222.0   Min.   :2.678   Min.   :2   Min.   :0.3539   Min.   :14.67
##  1st Qu.:222.5   1st Qu.:2.766   1st Qu.:2   1st Qu.:0.3539   1st Qu.:14.67
##  Median :223.0   Median :2.855   Median :2   Median :0.3539   Median :14.67
##  Mean   :223.0   Mean   :2.855   Mean   :2   Mean   :0.3539   Mean   :14.67
##  3rd Qu.:223.5   3rd Qu.:2.943   3rd Qu.:2   3rd Qu.:0.3539   3rd Qu.:14.67
##  Max.   :224.0   Max.   :3.032   Max.   :2   Max.   :0.3539   Max.   :14.67
##                                  NA's   :1   NA's   :1        NA's   :1
##      Pr(>F)
##  Min.   :1e-06
##  1st Qu.:1e-06
##  Median :1e-06
##  Mean   :1e-06
##  3rd Qu.:1e-06
##  Max.   :1e-06
##  NA's   :1
```

```
# Extracting the p-value
p <- f_test$`Pr(>F)`
```

```r
# Setting the significance level
a <- 0.95

# Checking if the p-value is less than the significance level
result <- ifelse(p < a, "Reject", "Fail to reject")
result
```

```
## [1] NA       "Reject"
```

That means at least one of friday or saturday is significant.

```r
#5
model_2 <- lm(log(cases) ~ time + holiday + saturday + fullmoon + newmoon, data = data)
summary(model_2)
```

```
##
## Call:
## lm(formula = log(cases) ~ time + holiday + saturday + fullmoon +
##     newmoon, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32410 -0.07408  0.00161  0.07508  0.29568
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.5467796  0.0153709 295.804  < 2e-16 ***
## time        0.0003461  0.0001122   3.085  0.00229 **
## holiday     0.1242282  0.0653773   1.900  0.05870 .
## saturday    0.0898496  0.0212036   4.237 3.31e-05 ***
## fullmoon    0.0236198  0.0404066   0.585  0.55944
## newmoon     0.0592636  0.0432074   1.372  0.17156
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1122 on 223 degrees of freedom
## Multiple R-squared:  0.1291, Adjusted R-squared:  0.1095
## F-statistic: 6.609 on 5 and 223 DF,  p-value: 9.295e-06
```

After dropping friday variable from the model, we see that p-value of fullmoon changed to 0.55944, but the difference is really small even after removing friday. So, the effect of a fullmoon remains statistically insignificant.

```r
#6
data$interval_time <- cut(data$time, breaks = c(0, 100, 200, Inf), labels = c("0-100 days", "101-200 da

model_3 <- lm(log(cases) ~ interval_time + holiday + saturday + fullmoon + newmoon, data = data)
summary(model_3)
```

```
##
## Call:
## lm(formula = log(cases) ~ interval_time + holiday + saturday +
```

```
##     fullmoon + newmoon, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31976 -0.07653  0.00001  0.07111  0.29032
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 4.55387    0.01165 390.936  < 2e-16 ***
## interval_time101-200 days   0.05964    0.01570   3.798 0.000188 ***
## interval_time200+ days      0.05601    0.02340   2.394 0.017518 *
## holiday                     0.11083    0.06472   1.712 0.088215 .
## saturday                    0.08919    0.02097   4.254  3.1e-05 ***
## fullmoon                    0.01986    0.03997   0.497 0.619786
## newmoon                     0.05867    0.04272   1.373 0.171076
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1109 on 222 degrees of freedom
## Multiple R-squared:  0.1524, Adjusted R-squared:  0.1295
## F-statistic: 6.652 on 6 and 222 DF,  p-value: 1.737e-06
```

R has automatically treated "0-100" as a reference group, and that is why it does not display "0-100" coefficients. It provides coefficients for the other groups in relation to the reference one.

In the regression model, the coefficients of the interval_time variable represent the estimated effects of different time intervals on log(cases). We see that "101-200" interval has a stronger and more statistically significant effect compared to "200+" interval. So, a longer time period within the range of "101-200" tends to be associated with a higher number of cases, and this effect is statistically supported.