Untitled

2024-04-21

```
library(readxl)
BhamParking <- read_excel("BhamParking.xlsx")
View(BhamParking)
```

```
# Summary statistics
summary(BhamParking)
```

```
##  SystemCodeNumber      Capacity      Occupancy        per_usage
##  Length:35332       Min.   : 220   Min.   :   0.0   Min.   :  0.00
##  Class :character   1st Qu.: 577   1st Qu.: 209.0   1st Qu.: 25.38
##  Mode  :character   Median : 863   Median : 448.0   Median : 46.67
##                     Mean   :1406   Mean   : 642.7   Mean   : 48.80
##                     3rd Qu.:2009   3rd Qu.: 796.0   3rd Qu.: 71.10
##                     Max.   :4675   Max.   :4327.0   Max.   :100.00
##                                    NA's   :19       NA's   :7
##  per_occupancy           year          month                day
##  Length:35332       Min.   :2016   Length:35332       Length:35332
##  Class :character   1st Qu.:2016   Class :character   Class :character
##  Mode  :character   Median :2016   Mode  :character   Mode  :character
##                     Mean   :2016
##                     3rd Qu.:2016
##                     Max.   :2016
##
##   WorkingDay             hour           period
##  Length:35332       Min.   : 1.000   Length:35332
##  Class :character   1st Qu.: 3.000   Class :character
##  Mode  :character   Median : 8.000   Mode  :character
##                     Mean   : 6.708
##                     3rd Qu.:10.000
##                     Max.   :12.000
##
```

```
# Get the first few rows of the dataset
head(BhamParking)
```

```
## # A tibble: 6 x 11
##   SystemCodeNumber Capacity Occupancy per_usage per_occupancy  year month day
##   <chr>               <dbl>     <dbl>     <dbl> <chr>         <dbl> <chr> <chr>
## 1 BHMBCCMKT01           577        61      10.6 0 - 25         2016 Oct   Tue
## 2 BHMBCCMKT01           577        64      11.1 0 - 25         2016 Oct   Tue
## 3 BHMBCCMKT01           577        80      13.9 0 - 25         2016 Oct   Tue
## 4 BHMBCCMKT01           577       107      18.5 0 - 25         2016 Oct   Tue
## 5 BHMBCCMKT01           577       150      26   25 - 50        2016 Oct   Tue
## 6 BHMBCCMKT01           577       177      30.7 25 - 50        2016 Oct   Tue
## # i 3 more variables: WorkingDay <chr>, hour <dbl>, period <chr>
```

```r
# Get the last few rows of the dataset
tail(BhamParking)
```

```
## # A tibble: 6 x 11
##   SystemCodeNumber Capacity Occupancy per_usage per_occupancy  year month day
##   <chr>               <dbl>     <dbl>     <dbl> <chr>         <dbl> <chr> <chr>
## 1 Shopping             1920      1521      79.2 75-100         2016 Dec   Mon
## 2 Shopping             1920      1517      79.0 75-100         2016 Dec   Mon
## 3 Shopping             1920      1487      77.4 75-100         2016 Dec   Mon
## 4 Shopping             1920      1432      74.6 50 - 75        2016 Dec   Mon
## 5 Shopping             1920      1321      68.8 50 - 75        2016 Dec   Mon
## 6 Shopping             1920      1180      61.5 50 - 75        2016 Dec   Mon
## # i 3 more variables: WorkingDay <chr>, hour <dbl>, period <chr>
```

```r
# Check for missing values
any(is.na(BhamParking))
```

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

```r
# Remove rows with missing values
BhamParking <- na.omit(BhamParking)
```

```r
# 1. Generate descriptive statistics for the dataset, and comment on the main trends.
# Descriptive Statistics

# Load necessary libraries
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# Generate descriptive statistics
summary(BhamParking)
```

```
##  SystemCodeNumber      Capacity      Occupancy        per_usage
##  Length:35300       Min.   : 220   Min.   :   0.0   Min.   :  0.00
##  Class :character   1st Qu.: 577   1st Qu.: 209.0   1st Qu.: 25.38
##  Mode  :character   Median : 863   Median : 448.0   Median : 46.69
##                     Mean   :1406   Mean   : 642.6   Mean   : 48.80
##                      3rd Qu.:2009   3rd Qu.: 796.0   3rd Qu.: 71.11
##                      Max.   :4675   Max.   :4327.0   Max.   :100.00
##  per_occupancy           year         month                day
##  Length:35300       Min.   :2016   Length:35300       Length:35300
##  Class :character   1st Qu.:2016   Class :character   Class :character
##  Mode  :character   Median :2016   Mode  :character   Mode  :character
##                      Mean   :2016
##                      3rd Qu.:2016
##                      Max.   :2016
##   WorkingDay             hour           period
##  Length:35300       Min.   : 1.000   Length:35300
##  Class :character   1st Qu.: 3.000   Class :character
##  Mode  :character   Median : 8.000   Mode  :character
##                      Mean   : 6.708
##                      3rd Qu.:10.000
##                      Max.   :12.000
```

```
# For categorical variables, you can use table() function
table(BhamParking$per_occupancy)
```

```
##
##  0 - 25 25 - 50 50 - 75  75-100
##    8677    10132    9139    7352
```

```
table(BhamParking$month)
```

```
##
##   Dec   Nov   Oct
##  8037 14851 12412
```

```
table(BhamParking$WorkingDay)
```

```
##
##    No   Yes
##  9267 26033
```

```
table(BhamParking$period)
```

```
##
##    AM    PM
## 16628 18672
```

```
# For numeric variables, you can use mean(), median(), sd(), min(), max(), etc.
mean(BhamParking$Capacity)
```

```
## [1] 1406.092
```

```
mean(BhamParking$Occupancy)
```

```
## [1] 642.6276
```

```
mean(BhamParking$per_usage)
```

```
## [1] 48.80022
```

```
# 2. Check any records with missing values and handle the missing data as appropriate.

# Check for missing values in the entire dataset
any(is.na(BhamParking))
```

```
## [1] FALSE
```

```
# Check for missing values in each column
colSums(is.na(BhamParking))
```

```
## SystemCodeNumber         Capacity        Occupancy        per_usage
##                0                0                0                0
##    per_occupancy             year            month              day
##                0                0                0                0
##       WorkingDay             hour           period
##                0                0                0
```

```
# Handle Missing Values
# Remove rows with any missing values
BhamParking <- na.omit(BhamParking)

# Impute missing values with mean (for numeric columns)
BhamParking$Occupancy[is.na(BhamParking$Occupancy)] <- mean(BhamParking$Occupancy, na.rm = TRUE)

# Impute missing values with mode (for categorical columns)
BhamParking$WorkingDay[is.na(BhamParking$WorkingDay)] <- which.max(table(BhamParking$WorkingDay))

# Check for missing values in the entire dataset
any(is.na(BhamParking))
```

```
## [1] FALSE
```

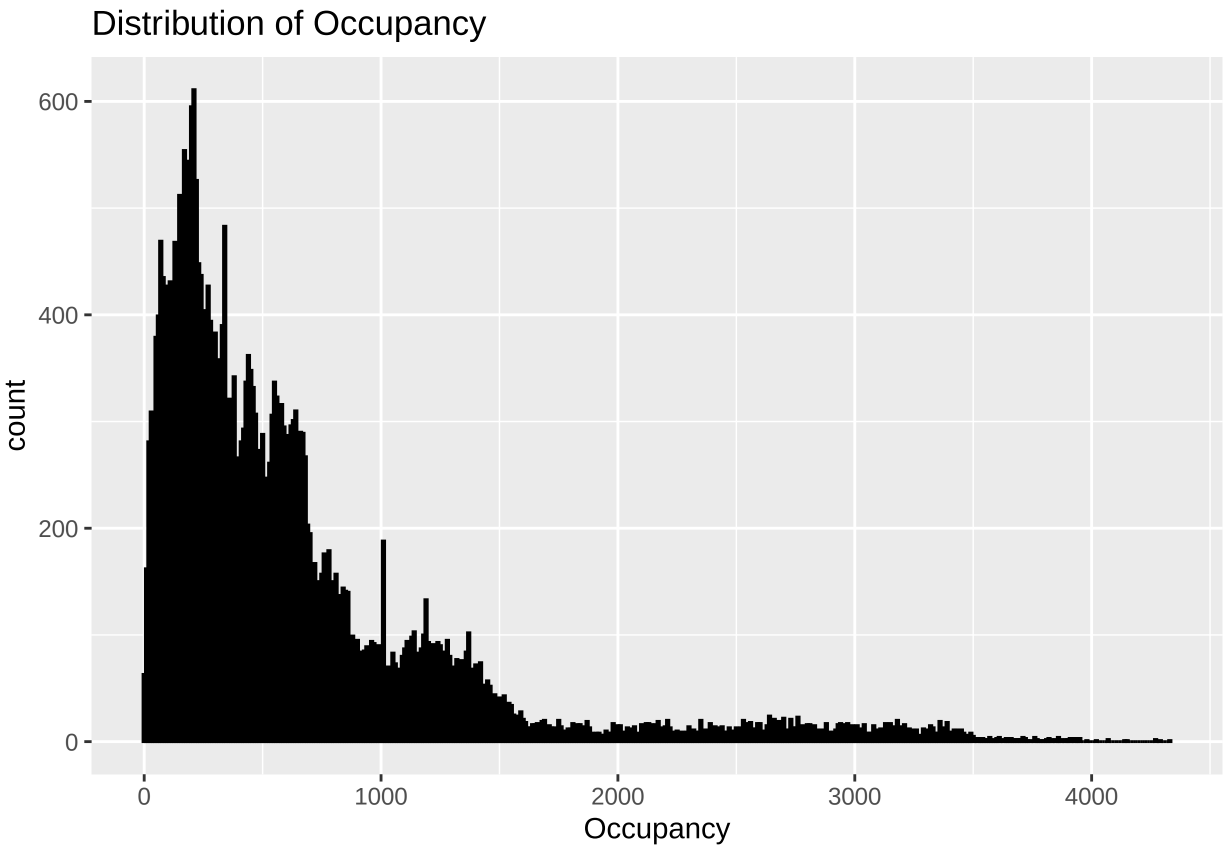
```
# Check for missing values in each column
colSums(is.na(BhamParking))
```

```
## SystemCodeNumber         Capacity        Occupancy        per_usage
##                0                0                0                0
##    per_occupancy             year            month              day
##                0                0                0                0
##       WorkingDay             hour           period
##                0                0                0
```

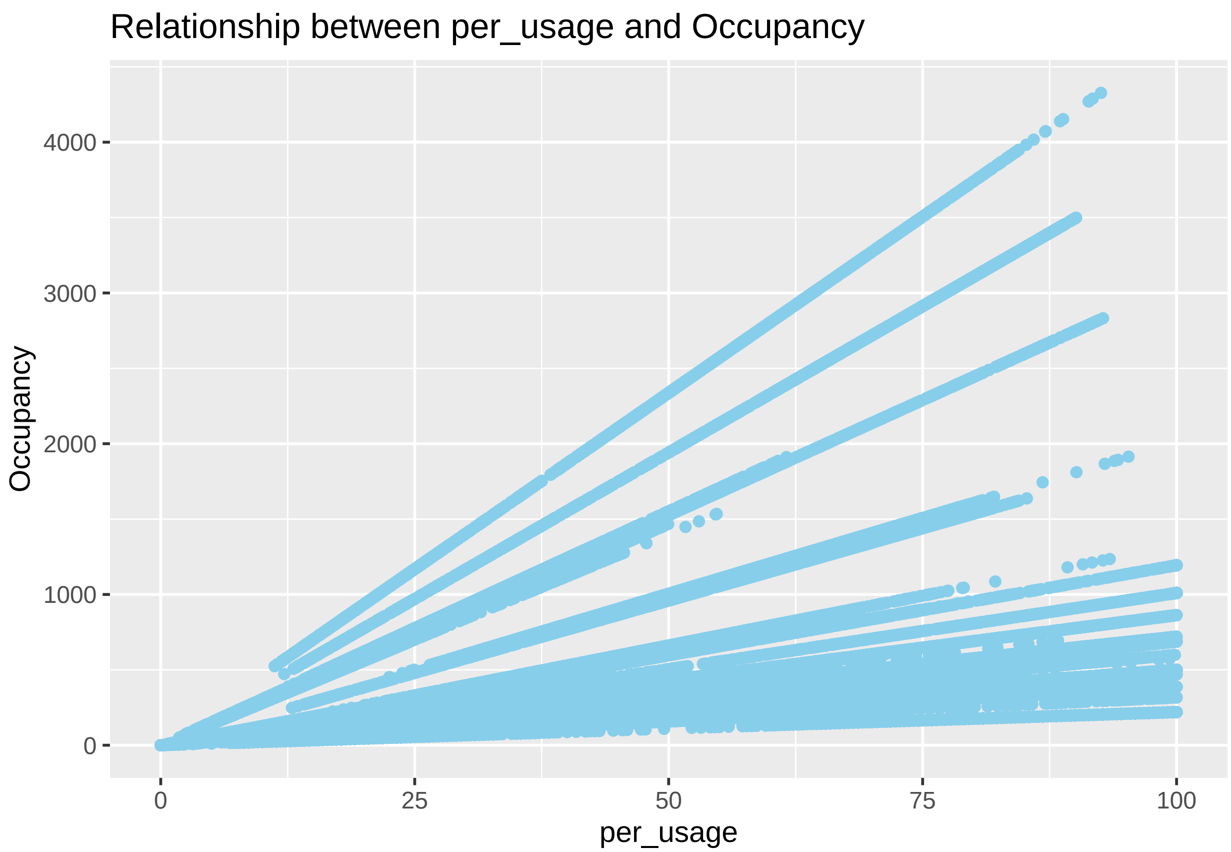
```
# 3.Build graphs visualizing the following and comment on the obtained visual insights the distribution
# the relationship of a pair of continuous variables the association b/w a categorical variable and a c

library(ggplot2)
```

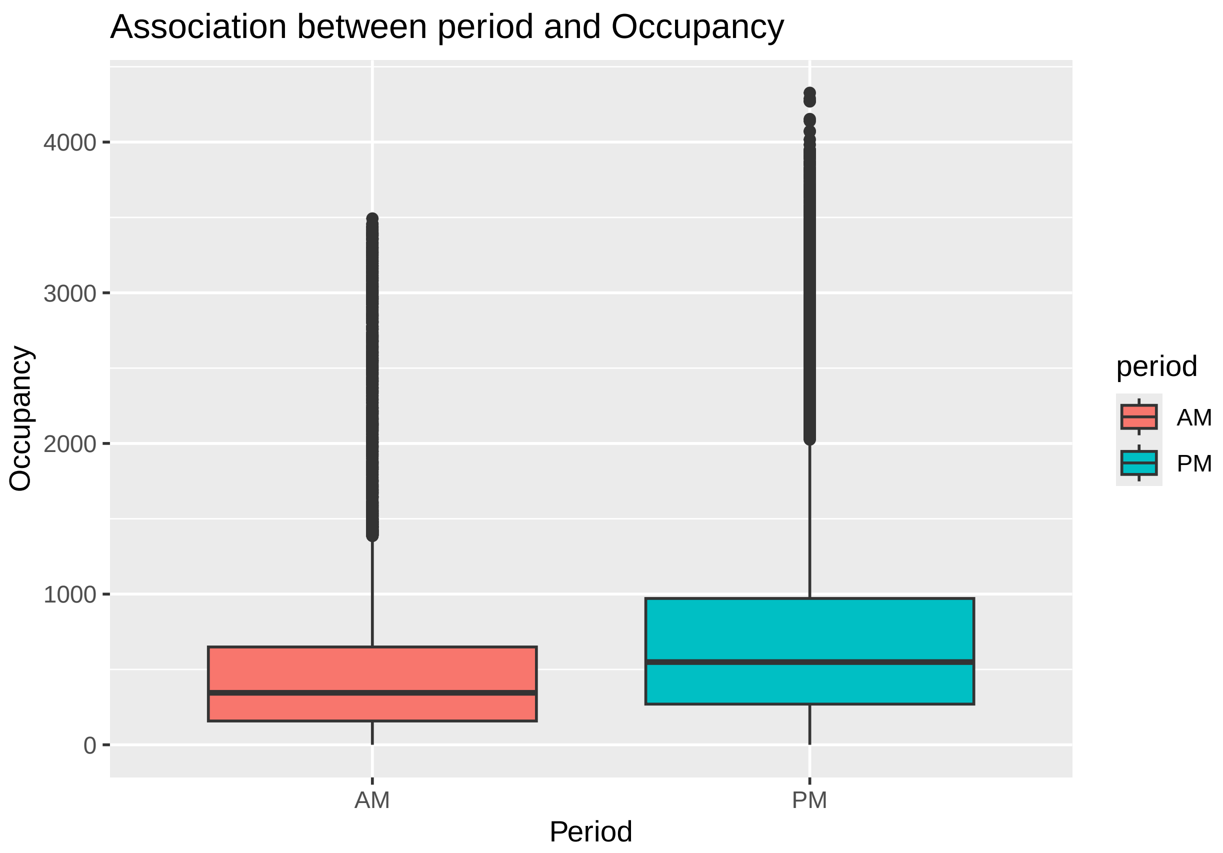
```r
# Histogram of Occupancy
ggplot(BhamParking, aes(x = Occupancy)) +
  geom_histogram(binwidth = 10, fill = "skyblue", color = "black") +
  labs(title = "Distribution of Occupancy")
```



```r
# Scatter plot of Occupancy vs. per_usage
ggplot(BhamParking, aes(x = per_usage, y = Occupancy)) +
  geom_point(color = "skyblue") +
  labs(title = "Relationship between per_usage and Occupancy",
       x = "per_usage", y = "Occupancy")
```



```
# Boxplot of Occupancy by period
ggplot(BhamParking, aes(x = period, y = Occupancy, fill = period)) +
  geom_boxplot() +
  labs(title = "Association between period and Occupancy",
       x = "Period", y = "Occupancy")
```



```
# 4. Display unique values of a categorical variable and their frequencies.

# Display unique values and their frequencies for the WorkingDay variable
table(BhamParking$WorkingDay)
```

```
##
##    No   Yes
##  9267 26033
```

```
# 5. Build a contingency table of two potentially related categorical variables.
# Conduct a statistical test of the independence between them and interpret the results.

# Create a contingency table of WorkingDay and period
contingency_table <- table(BhamParking$WorkingDay, BhamParking$period)

# Display the contingency table
contingency_table
```

```
##
##           AM    PM
##   No    4396  4871
##   Yes  12232 13801
```

```r
# Perform a chi-squared test of independence
chi_sq_test <- chisq.test(contingency_table)

# Display the results of the chi-squared test
chi_sq_test
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  contingency_table
## X-squared = 0.53904, df = 1, p-value = 0.4628
```

```r
# 6. Retrieve one or more subset of rows based on two or more criteria and present descriptive statisti

# Subset of data for WorkingDay = "Yes" and period = "AM"
subset1 <- subset(BhamParking, WorkingDay == "Yes" & period == "AM")

# Subset of data for WorkingDay = "No" and period = "AM"
subset2 <- subset(BhamParking, WorkingDay == "No" & period == "AM")

# Calculate descriptive statistics for subset1
summary(subset1)
```

```
##  SystemCodeNumber      Capacity      Occupancy        per_usage
##  Length:12232       Min.   : 220   Min.   :   0.0   Min.   :  0.00
##  Class :character   1st Qu.: 500   1st Qu.: 193.0   1st Qu.: 22.37
##  Mode  :character   Median : 849   Median : 403.0   Median : 40.78
##                     Mean   :1391   Mean   : 563.4   Mean   : 44.74
##                     3rd Qu.:2009   3rd Qu.: 699.0   3rd Qu.: 65.84
##                     Max.   :4675   Max.   :3493.0   Max.   :100.00
##  per_occupancy           year         month                day
##  Length:12232       Min.   :2016   Length:12232       Length:12232
##  Class :character   1st Qu.:2016   Class :character   Class :character
##  Mode  :character   Median :2016   Mode  :character   Mode  :character
##                     Mean   :2016
##                     3rd Qu.:2016
##                     Max.   :2016
##   WorkingDay             hour           period
##  Length:12232       Min.   : 7.000   Length:12232
##  Class :character   1st Qu.: 8.000   Class :character
##  Mode  :character   Median : 9.000   Mode  :character
##                     Mean   : 9.384
##                     3rd Qu.:10.000
##                     Max.   :11.000
```

```r
# Calculate descriptive statistics for subset2
summary(subset2)
```

```
##  SystemCodeNumber      Capacity      Occupancy        per_usage
##  Length:4396        Min.   : 220   Min.   :   0.0   Min.   :  0.00
##  Class :character   1st Qu.: 577   1st Qu.:  89.0   1st Qu.: 12.78
##  Mode  :character   Median : 863   Median : 220.0   Median : 21.14
##                        Mean   :1420   Mean   : 352.7   Mean   : 25.45
##                        3rd Qu.:2009   3rd Qu.: 480.2   3rd Qu.: 35.43
##                        Max.   :4675   Max.   :3297.0   Max.   :100.00
##  per_occupancy           year         month               day
##  Length:4396        Min.   :2016   Length:4396        Length:4396
##  Class :character   1st Qu.:2016   Class :character   Class :character
##  Mode  :character   Median :2016   Mode  :character   Mode  :character
##                     Mean   :2016
##                     3rd Qu.:2016
##                     Max.   :2016
##   WorkingDay             hour           period
##  Length:4396        Min.   : 7.000   Length:4396
##  Class :character   1st Qu.: 8.000   Class :character
##  Mode  :character   Median : 9.000   Mode  :character
##                     Mean   : 9.296
##                     3rd Qu.:10.000
##                     Max.   :11.000
```

```r
# 7. Conduct a statistical test of the significance of the difference
# between the means of two subsets of the data and interpret the results.

# Assuming 'Occupancy' is the variable for which you want to compare means
# Conduct a t-test
t_test_result <- t.test(subset1$Occupancy, subset2$Occupancy)

# Print the results
print(t_test_result)
```

```
##
##  Welch Two Sample t-test
##
## data:  subset1$Occupancy and subset2$Occupancy
## t = 26.423, df = 10996, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  195.0594 226.3188
## sample estimates:
## mean of x mean of y
##  563.3920  352.7029
```

```r
# Conduct a Welch's t-test
t_test_result <- t.test(subset1$Occupancy, subset2$Occupancy, var.equal = FALSE)

# Print the results
print(t_test_result)
```

```
##
##  Welch Two Sample t-test
##
## data:  subset1$Occupancy and subset2$Occupancy
## t = 26.423, df = 10996, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##  195.0594 226.3188
## sample estimates:
## mean of x mean of y
##  563.3920  352.7029
```

```r
# 8. Create one or more tables that group the data by a certain categorical variable and
# display summarized information for each group (e.g., the mean or sum within the group).

library(dplyr)

# Group the data by WorkingDay and calculate the mean Occupancy for each group
summary_table <- BhamParking %>%
  group_by(WorkingDay) %>%
  summarise(mean_occupancy = mean(Occupancy))

# Display the summary table
print(summary_table)
```

```
## # A tibble: 2 x 2
##   WorkingDay mean_occupancy
##   <chr>               <dbl>
## 1 No                   544.
## 2 Yes                  678.
```

```r
library(dplyr)

# Select numeric columns
numeric_cols <- c("Capacity", "Occupancy", "per_usage", "year", "hour")

# Group the data by WorkingDay and calculate multiple summary statistics for numeric columns
summary_table <- BhamParking %>%
  group_by(WorkingDay) %>%
  summarise(across(numeric_cols, list(mean = mean, sum = sum, median = median)))
```

```
## Warning: There was 1 warning in ‘summarise()‘.
## i In argument: ‘across(numeric_cols, list(mean = mean, sum = sum, median =
##   median))‘.
## Caused by warning:
## ! Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use ‘all_of()‘ or ‘any_of()‘ instead.
##   # Was:
##   data %>% select(numeric_cols)
##
##   # Now:
##   data %>% select(all_of(numeric_cols))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
```

```r
# Display the summary table
print(summary_table)
```

```
## # A tibble: 2 x 16
##   WorkingDay Capacity_mean Capacity_sum Capacity_median Occupancy_mean
##   <chr>              <dbl>        <dbl>           <dbl>          <dbl>
## 1 No                 1433.     13277635             863           544.
## 2 Yes                1397.     36357422             849           678.
## # i 11 more variables: Occupancy_sum <dbl>, Occupancy_median <dbl>,
## #   per_usage_mean <dbl>, per_usage_sum <dbl>, per_usage_median <dbl>,
## #   year_mean <dbl>, year_sum <dbl>, year_median <dbl>, hour_mean <dbl>,
## #   hour_sum <dbl>, hour_median <dbl>
```

```
# 9. Implement a linear regression model and interpret its output including its accuracy
# Before you start to work on this assignment, please familiarise yourself with the detailed
# evaluation criteria for this assignment by studying the Courswork Brief (see above).

# Fit the linear regression model
model <- lm(Occupancy ~ per_usage, data = BhamParking)

# Summarize the model
summary(model)
```

```
##
## Call:
## lm(formula = Occupancy ~ per_usage, data = BhamParking)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -871.2 -358.2 -222.7  167.8 3301.0
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 215.0815     6.8341   31.47   <2e-16 ***
## per_usage     8.7612     0.1228   71.33   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 616.8 on 35298 degrees of freedom
## Multiple R-squared:  0.126,  Adjusted R-squared:  0.126
## F-statistic:  5088 on 1 and 35298 DF,  p-value: < 2.2e-16
```