[AP] Assignment

2025-03-30

I. Personal Information:

- Name: Pham Hoang Minh Hien
- Student ID: 22027464
- Subject: Annalytic Programming

II. Declaration:

By including this statement, we the authors of this work, verify that:

- We hold a copy of this assignment that we can produce if the original is lost or damaged.
- We hereby certify that no part of this assignment/product has been copied from any other student's work or from any other source except where due acknowledgement is made in the assignment.
- No part of this assignment/product has been written/produced for us by another person except where such collaboration has been authorised by the subject lecturer/tutor concerned.
- We are aware that this work may be reproduced and submitted to plagiarism detection software programs for the purpose of detecting possible plagiarism (which may retain a copy on its database for future plagiarism checking).
- We hereby certify that we have read and understand what the School of Computing, Engineering and Mathematics defines as minor and substantial breaches of misconduct as outlined in the learning guide for this unit.

III. Assignment

Question 1:

1. Write the code to inspect the data structure and present the data:
2. The missing values in the dataset were written as “?”, replace any “?” with NA;
3. Convert categorical variables BodyStyles, FuelTypes, ErrorCodes to factors;
4. Replace the missing values in column Horsepower with the mean horsepower;
5. Select the appropriate chart type and display: horsepower distribution.

Loading dataset to R

```
auto = read.csv("C:/Users/LENOVO/Downloads/Automobile.csv")
engine = read.csv("C:/Users/LENOVO/Downloads/Engine.csv")
maintenance = read.csv("C:/Users/LENOVO/Downloads/Maintenance.csv")
```

Inspect dataset

```
## Automobile dataset
head(auto)

##   PlateNumber Manufactures  BodyStyles DriveWheels EngineLocation
WheelBase
## 1     53N-001  Alfa-romero convertible         rwd          front
88.6
## 2     53N-002  Alfa-romero   hatchback         rwd          front
94.5
## 3     53N-003         Audi       sedan         fwd          front
99.8
## 4     53N-004         Audi       sedan         4wd          front
99.4
## 5     53N-005         Audi       sedan         fwd          front
99.8
## 6     53N-006         Audi       sedan         fwd          front
105.8
##   Length Width Height CurbWeight EngineModel CityMpg HighwayMpg
## 1  168.8  64.1   48.8       2548      E-0001      21         27
## 2  171.2  65.5   52.4       2823      E-0002      19         26
## 3  176.6  66.2   54.3       2337      E-0003      24         30
## 4  176.6  66.4   54.3       2824      E-0004      18         22
## 5  177.3  66.3   53.1       2507      E-0005      19         25
## 6  192.7  71.4   55.7       2844      E-0005      19         25

str(auto)

## 'data.frame':    204 obs. of  13 variables:
##  $ PlateNumber   : chr  "53N-001" "53N-002" "53N-003" "53N-004" ...
##  $ Manufactures  : chr  "Alfa-romero" "Alfa-romero" "Audi" "Audi" ...
##  $ BodyStyles    : chr  "convertible" "hatchback" "sedan" "sedan" ...
##  $ DriveWheels   : chr  "rwd" "rwd" "fwd" "4wd" ...
##  $ EngineLocation: chr  "front" "front" "front" "front" ...
##  $ WheelBase     : num  88.6 94.5 99.8 99.4 99.8 ...
##  $ Length        : num  169 171 177 177 177 ...
##  $ Width         : num  64.1 65.5 66.2 66.4 66.3 71.4 71.4 71.4 67.9 64.8
...
##  $ Height        : num  48.8 52.4 54.3 54.3 53.1 55.7 55.7 55.9 52 54.3
...
##  $ CurbWeight    : int  2548 2823 2337 2824 2507 2844 2954 3086 3053 2395
...
##  $ EngineModel   : chr  "E-0001" "E-0002" "E-0003" "E-0004" ...
##  $ CityMpg       : int  21 19 24 18 19 19 19 17 16 23 ...
##  $ HighwayMpg    : int  27 26 30 22 25 25 25 20 22 29 ...
```

```
summary(auto)
```

```
##  PlateNumber        Manufactures        BodyStyles        DriveWheels
##  Length:204         Length:204         Length:204         Length:204
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  EngineLocation       WheelBase          Length          Width
##  Length:204         Min.   : 86.60   Min.   :141.1   Min.   :60.30
##  Class :character   1st Qu.: 94.50   1st Qu.:166.3   1st Qu.:64.08
##  Mode  :character   Median : 97.00   Median :173.2   Median :65.50
##                     Mean   : 98.81   Mean   :174.1   Mean   :65.92
##                     3rd Qu.:102.40   3rd Qu.:183.2   3rd Qu.:66.90
##                     Max.   :120.90   Max.   :208.1   Max.   :72.30
##      Height        CurbWeight   EngineModel           CityMpg
##  Min.   :47.80   Min.   :1488   Length:204         Min.   :10.00
##  1st Qu.:52.00   1st Qu.:2145   Class :character   1st Qu.:19.00
##  Median :54.10   Median :2414   Mode  :character   Median :24.00
##  Mean   :53.75   Mean   :2556                      Mean   :25.23
##  3rd Qu.:55.50   3rd Qu.:2939                      3rd Qu.:30.00
##  Max.   :59.80   Max.   :4066                      Max.   :50.00
##    HighwayMpg
##  Min.   :15.00
##  1st Qu.:25.00
##  Median :30.00
##  Mean   :30.76
##  3rd Qu.:34.50
##  Max.   :55.00
```

```
## Engine dataset
head(engine)
```

```
##   EngineModel EngineType NumCylinders EngineSize FuelSystem Horsepower
## 1      E-0001       dohc         four        130       mpfi        111
## 2      E-0002       ohcv          six        152       mpfi        154
## 3      E-0003        ohc         four        109       mpfi        102
## 4      E-0004        ohc         five        136       mpfi        115
## 5      E-0005        ohc         five        136       mpfi        110
## 6      E-0006        ohc         five        131       mpfi        140
##   FuelTypes Aspiration
## 1       gas        std
## 2       gas        std
## 3       gas        std
## 4       gas        std
## 5       gas        std
## 6       gas      turbo
```

```
str(engine)
```

```
## 'data.frame':    88 obs. of  8 variables:
##  $ EngineModel : chr  "E-0001" "E-0002" "E-0003" "E-0004" ...
##  $ EngineType  : chr  "dohc" "ohcv" "ohc" "ohc" ...
##  $ NumCylinders: chr  "four" "six" "four" "five" ...
##  $ EngineSize  : int  130 152 109 136 136 131 131 108 164 164 ...
##  $ FuelSystem  : chr  "mpfi" "mpfi" "mpfi" "mpfi" ...
##  $ Horsepower  : chr  "111" "154" "102" "115" ...
##  $ FuelTypes   : chr  "gas" "gas" "gas" "gas" ...
##  $ Aspiration  : chr  "std" "std" "std" "std" ...
```

```
summary(engine)
```

```
##  EngineModel         EngineType         NumCylinders         EngineSize   
##  Length:88          Length:88          Length:88          Min.   : 60.0  
##  Class :character   Class :character   Class :character   1st Qu.:108.0  
##  Mode  :character   Mode  :character   Mode  :character   Median :121.0  
##                                                           Mean   :134.1  
##                                                           3rd Qu.:151.2  
##                                                           Max.   :320.0  
##   FuelSystem         Horsepower          FuelTypes          Aspiration       
##  Length:88          Length:88          Length:88          Length:88         
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                             
##                                                                             
## 
```

```
## Maintenance  dataset
head(maintenance)
```

```
##   ID PlateNumber       Date           Troubles ErrorCodes Price
Methods
## 1  1     53N-001 15/02/2024       Break system         -1   110
Replacement
## 2  2     53N-001 16/03/2024       Transmission         -1   175
Replacement
## 3  3     53N-001 15/04/2024   Suspected clutch         -1   175
Adjustment
## 4  4     53N-001 15/05/2024 Ignition (finding)          1   180
Adjustment
## 5  5     53N-001 14/06/2024            Chassis         -1    85
Replacement
## 6  6     53N-002 15/02/2024          Cylinders          1  1000
Replacement
```

```
str(maintenance)
```

```
## 'data.frame':    374 obs. of  7 variables:
##  $ ID         : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ PlateNumber: chr  "53N-001" "53N-001" "53N-001" "53N-001" ...
##  $ Date       : chr  "15/02/2024" "16/03/2024" "15/04/2024" "15/05/2024"
```

```
...
##  $ Troubles   : chr  "Break system" "Transmission" "Suspected clutch"
"Ignition (finding)" ...
##  $ ErrorCodes : int  -1 -1 -1 1 -1 1 1 0 -1 -1 ...
##  $ Price      : int  110 175 175 180 85 1000 180 0 180 180 ...
##  $ Methods    : chr  "Replacement" "Replacement" "Adjustment" "Adjustment"
...
```

```
summary(maintenance)
```

```
##        ID          PlateNumber            Date             Troubles
##  Min.   :  1.00   Length:374         Length:374         Length:374
##  1st Qu.: 94.25   Class :character   Class :character   Class :character
##  Median :187.50   Mode  :character   Mode  :character   Mode  :character
##  Mean   :187.50
##  3rd Qu.:280.75
##  Max.   :374.00
##    ErrorCodes           Price          Methods
##  Min.   :-1.00000   Min.   :   0.0   Length:374
##  1st Qu.:-1.00000   1st Qu.:  85.0   Class :character
##  Median : 0.00000   Median : 120.0   Mode  :character
##  Mean   : 0.04813   Mean   : 204.8
##  3rd Qu.: 1.00000   3rd Qu.: 180.0
##  Max.   : 1.00000   Max.   :1000.0
```

Define “?” and replace with NA

```
# Engine dataset
summary(engine == "?")
```

```
##  EngineModel     EngineType      NumCylinders    EngineSize
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:88        FALSE:83        FALSE:88        FALSE:88
##                  TRUE :5
##  FuelSystem      Horsepower      FuelTypes       Aspiration
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:88        FALSE:87        FALSE:88        FALSE:88
##                  TRUE :1
```

```
# Auto dataset
summary(auto == "?")
```

```
##  PlateNumber     Manufactures    BodyStyles      DriveWheels
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:204       FALSE:204       FALSE:204       FALSE:204
##  EngineLocation  WheelBase         Length          Width
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:204       FALSE:204       FALSE:204       FALSE:204
##    Height        CurbWeight      EngineModel      CityMpg
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical
##  FALSE:204       FALSE:204       FALSE:204       FALSE:204
##  HighwayMpg
##  Mode :logical  
##  FALSE:204      

# Maintenance dataset
summary(maintenance == "?")

##      ID           PlateNumber        Date           Troubles      
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical  
##  FALSE:374       FALSE:374       FALSE:374       FALSE:374      
##                                                                 
##  ErrorCodes        Price          Methods       
##  Mode :logical   Mode :logical   Mode :logical  
##  FALSE:374       FALSE:374       FALSE:346      
##                                  NA's :28       
```

After scanning through three datasets, there are “?” in Engine dataset. Therefore, missing values represented by “?” were replaced with NA to standardize across datasets. .

```
# Replace "?" with NA
engine_new <- engine
engine_new[engine_new == "?"] <- NA

summary(engine_new == "?")

##  EngineModel     EngineType      NumCylinders    EngineSize     
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical  
##  FALSE:88        FALSE:83        FALSE:88        FALSE:88       
##                  NA's :5                                        
##  FuelSystem      Horsepower      FuelTypes       Aspiration     
##  Mode :logical   Mode :logical   Mode :logical   Mode :logical  
##  FALSE:88        FALSE:87        FALSE:88        FALSE:88       
##                  NA's :1                                        
```

Change factor (Convert categorical variables BodyStyles, FuelTypes, ErrorCodes to factors;)

```
# Inspect the dataset for columns names
names(auto)

##  [1] "PlateNumber"    "Manufactures"   "BodyStyles"     "DriveWheels"   
##  [5] "EngineLocation" "WheelBase"      "Length"         "Width"         
##  [9] "Height"         "CurbWeight"     "EngineModel"    "CityMpg"       
## [13] "HighwayMpg"

names(engine_new)

## [1] "EngineModel"  "EngineType"   "NumCylinders" "EngineSize"  
"FuelSystem"  
## [6] "Horsepower"   "FuelTypes"    "Aspiration"  

names(maintenance)
```

```
## [1] "ID"          "PlateNumber" "Date"        "Troubles"    "ErrorCodes"
## [6] "Price"       "Methods"
```

```
# Change the variables to factors
auto$BodyStyles <- as.factor(auto$BodyStyles)
engine_new$FuelTypes <- as.factor(engine_new$FuelTypes)
maintenance$ErrorCodes <- as.factor(maintenance$ErrorCodes)

# Check the dataset types of variables
str(auto)
```

```
## 'data.frame':    204 obs. of  13 variables:
##  $ PlateNumber   : chr  "53N-001" "53N-002" "53N-003" "53N-004" ...
##  $ Manufactures  : chr  "Alfa-romero" "Alfa-romero" "Audi" "Audi" ...
##  $ BodyStyles    : Factor w/ 5 levels "convertible",..: 1 3 4 4 4 4 5 4 3
4 ...
##  $ DriveWheels   : chr  "rwd" "rwd" "fwd" "4wd" ...
##  $ EngineLocation: chr  "front" "front" "front" "front" ...
##  $ WheelBase     : num  88.6 94.5 99.8 99.4 99.8 ...
##  $ Length        : num  169 171 177 177 177 ...
##  $ Width         : num  64.1 65.5 66.2 66.4 66.3 71.4 71.4 71.4 67.9 64.8
...
##  $ Height        : num  48.8 52.4 54.3 54.3 53.1 55.7 55.7 55.9 52 54.3
...
##  $ CurbWeight    : int  2548 2823 2337 2824 2507 2844 2954 3086 3053 2395
...
##  $ EngineModel   : chr  "E-0001" "E-0002" "E-0003" "E-0004" ...
##  $ CityMpg       : int  21 19 24 18 19 19 19 17 16 23 ...
##  $ HighwayMpg    : int  27 26 30 22 25 25 25 20 22 29 ...
```

```
str(engine_new)
```

```
## 'data.frame':    88 obs. of  8 variables:
##  $ EngineModel : chr  "E-0001" "E-0002" "E-0003" "E-0004" ...
##  $ EngineType  : chr  "dohc" "ohcv" "ohc" "ohc" ...
##  $ NumCylinders: chr  "four" "six" "four" "five" ...
##  $ EngineSize  : int  130 152 109 136 136 131 131 108 164 164 ...
##  $ FuelSystem  : chr  "mpfi" "mpfi" "mpfi" "mpfi" ...
##  $ Horsepower  : chr  "111" "154" "102" "115" ...
##  $ FuelTypes   : Factor w/ 2 levels "diesel","gas": 2 2 2 2 2 2 2 2 2 2
...
##  $ Aspiration  : chr  "std" "std" "std" "std" ...
```

```
str(maintenance)
```

```
## 'data.frame':    374 obs. of  7 variables:
##  $ ID         : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ PlateNumber: chr  "53N-001" "53N-001" "53N-001" "53N-001" ...
##  $ Date       : chr  "15/02/2024" "16/03/2024" "15/04/2024" "15/05/2024"
...
##  $ Troubles   : chr  "Break system" "Transmission" "Suspected clutch"
```

```
"Ignition (finding)" ...
##  $ ErrorCodes : Factor w/ 3 levels "-1","0","1": 1 1 1 3 1 3 3 2 1 1 ...
##  $ Price      : int  110 175 175 180 85 1000 180 0 180 180 ...
##  $ Methods    : chr  "Replacement" "Replacement" "Adjustment" "Adjustment"
...
```

Replace the missing values in column Horsepower with the mean horsepower;

```
# Convert Horsepower to numeric
engine_new$Horsepower <- as.numeric(engine_new$Horsepower)

# Calculate the mean of Horsepower, excluding NA values
hp_mean <- mean(engine_new$Horsepower, na.rm = TRUE)

# Replace NA values in Horsepower with the calculated mean
engine_new$Horsepower[is.na(engine_new$Horsepower)] <- hp_mean

# Check for NA
summary(is.na(engine_new$Horsepower))
```

```
##    Mode   FALSE
## logical      88
```

After checking the summary of Engine_new data, the missing data in Horesepower has been replaced to mean Horepower from the whole column.It creates the continuos variable with a nearly normal distribution for further analysis.

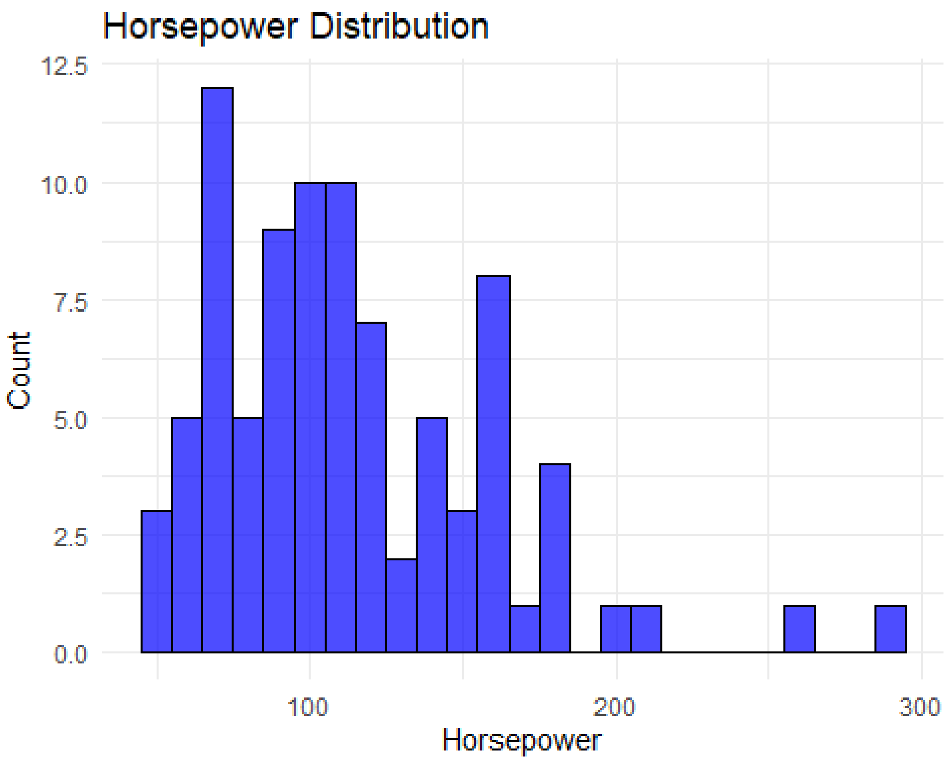
The EngineType column which has 5 NAs, can not replace, due to the character variables.

Select the appropriate chart type and display: horsepower distribution.

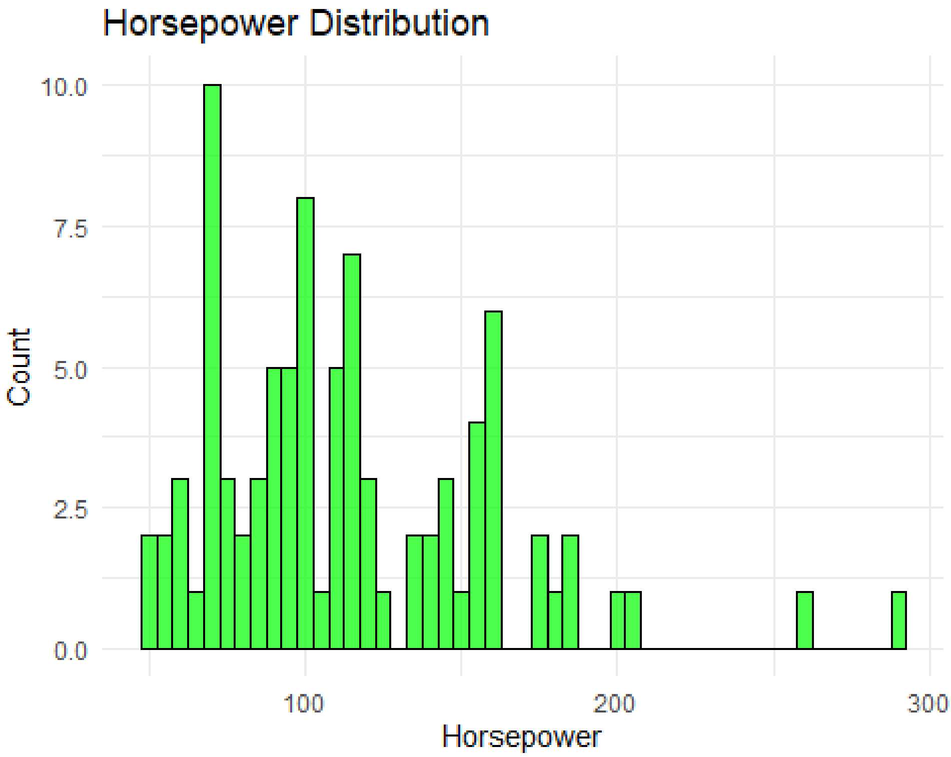
```
#Launching the needed library
library("ggplot2")
```

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

```
ggplot(engine_new, aes(x = Horsepower)) +
  geom_histogram(binwidth = 10, fill = "blue", color = "black", alpha = 0.7)
+
  labs(title = "Horsepower Distribution", x = "Horsepower", y = "Count") +
  theme_minimal()
```



```r
ggplot(engine_new, aes(x = Horsepower)) +
  geom_histogram(binwidth = 5, fill = "green", color = "black", alpha = 0.7)
+
  labs(title = "Horsepower Distribution", x = "Horsepower", y = "Count") +
  theme_minimal()
```



The histograms effectively illustrate how horsepower values are distributed across different ranges, helping to identify both common values and outliers. Peaks in the distribution indicate horsepower ranges with high frequency, while dips highlight less common values.

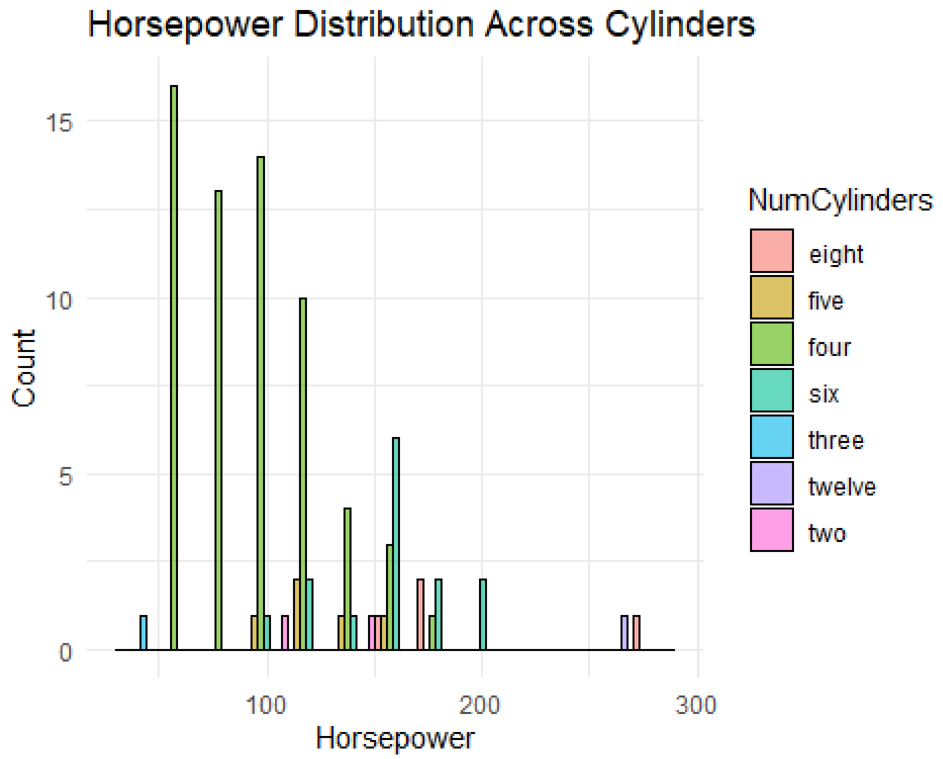
There is a noticeable difference between the two histograms due to the different bin widths used: the first histogram uses a binwidth of 10, while the second uses 5. The choice of binwidth affects the level of detail shown: Larger binwidths reveal the overall shape, while smaller bins highlight more detailed fluctuations in horsepower distribution.

Question 2

Write the code to analyse the distribution of the horsepower across the number of cylinders. Write the code to investigate the distribution of the horsepower across the groups of the engine sizes (e.g., 60-100, 101-200, 201-300, 301+). Visualize both the findings using the histogram. Explain your findings.

Call out the Cylinder

```
engine_new$NumCylinders <- as.factor(engine_new$NumCylinders)

#Create visualization
ggplot(engine_new, aes(x = Horsepower, fill = NumCylinders)) +
  geom_histogram(binwidth = 20, color = "black", alpha = 0.6, position =
"dodge") +
  labs(title = "Horsepower Distribution Across Cylinders", x = "Horsepower",
y = "Count") +
  theme_minimal()
```



Analysis:

- Through the graph “Horsepower Distribution Across Cylinders”, the most outstanding Cylinder in the dataset is the number of cars which have three-cylinder. They appear to be the most common configuration which cover a wide range from under 100 to almost 200 HP. In a deeper analysed, they primarily concentrated in the range from 75 - 150 HP, which the highest frequency is almost 100HP.

- Fourth-Cylinders has the second most popular since they appears mostly from 100 - 200 HP ranges. However, the counts of number are not out-standingly compared to the Three-cylinder, they have higher Horsepower ranges.

- Other cylinder, such as the number of Five, Six, One have the least count due to they have the low range performance (over 50 HP for Five-Cylinder) and very high performance that other can not reach (approximately 300 HP).

- Through out the distribution, it shows a clear positive correlation between the number of cylinders and horsepower; however, to demonstrate the correlation of these two varibales, we can used correlation test to identify correctly.

```
# Converting NumCylinders back to numeric for correlation test
engine_new$NumCylinders <- as.numeric(engine_new$NumCylinders)
```

```r
cor_test <- cor.test(engine_new$NumCylinders, engine_new$Horsepower)
print(cor_test)
```

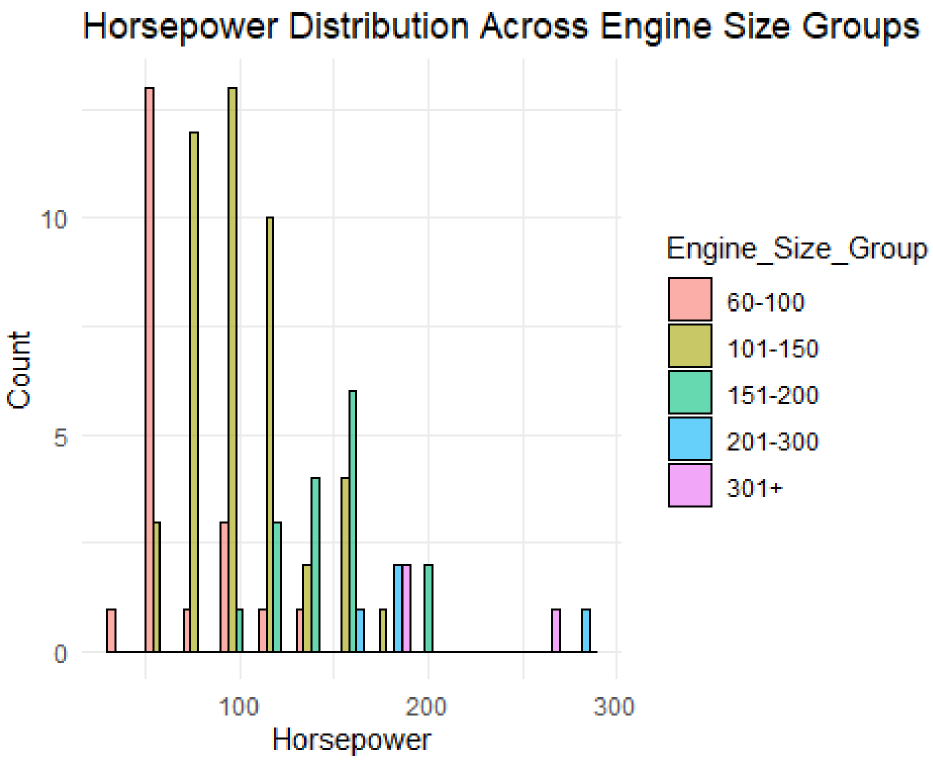
```
##
##  Pearson's product-moment correlation
##
## data:  engine_new$NumCylinders and engine_new$Horsepower
## t = 0.35786, df = 86, p-value = 0.7213
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1722736  0.2460154
## sample estimates:
##        cor
## 0.03855999
```

The p-value of the correlation test is higher than 0.05 (0.7213 > 0.05) which indicate that there are not significantly impact each other.

Group size

Based on the ranges of Engine Size, the dataset provides wide range of numbers from 60 to over 200. As the example of number cylinders above, we can divide the Engine Size to 5 groups, with 50 size range each.

```r
# Create engine size groups
engine_new$Engine_Size_Group <- cut(engine_new$EngineSize,
                                    breaks = c(60, 100, 150, 200, 300, Inf),
                                    labels = c("60-100","101-150", "151-200",
"201-300", "301+"),
                                    include.lowest = TRUE)

# Histogram of Horsepower grouped by Engine Size
ggplot(engine_new, aes(x = Horsepower, fill = Engine_Size_Group)) +
  geom_histogram(binwidth = 20, color = "black", alpha = 0.6, position =
"dodge") +
  labs(title = "Horsepower Distribution Across Engine Size Groups", x =
"Horsepower", y = "Count") +
  theme_minimal()
```



Analysis:

- From the histogram, there are two groups have high frequency, compare to other three group that have Engine Size from 150 and above.

- Comparing the first two groups (Engine Size under 100 and Engine Size from 101 to 150), Size Under 100 group has a horsepower range from under 50HP to almost 150 HP, which their highest frequency of Horsepower at 50HP with over 15 cars are reported. The group from 101-150 Size has a wider range, which from 50HP to almost 175HP. They have the highest reported frequency is from 75HP to 150HP with the peak with over 15 cars report at 100 HP.

- Other three engines size groups have largest horpower range, which is from 100HP and above. With the engine size from 200, the horsepower is reported to be almost 200HP; Relatively, the car could reach to almost 300HP if the engine are from the rang 300.

- To test the relationship of the Engine size to the Horsepower performance, we can use correlation test:

```
# Using original EngineSize column
cor_test2 <- cor.test(engine_new$EngineSize, engine_new$Horsepower)
print(cor_test2)

##
##  Pearson's product-moment correlation
##
```

```
## data:  engine_new$EngineSize and engine_new$Horsepower
## t = 11.539, df = 86, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6812286 0.8501182
## sample estimates:
##       cor
## 0.7794591
```

The p-value is small, under 0.05 which can demtermine that Engíne Size have positve impact to Horsepower performance. Engines have larger size tend to have larger displacements and produce more power.

Question 3:

Filter out those engines in the dataset that have trouble or are suspected of having trouble; What are the top 5 most common troubles related to the engines? Do the troubles differ between fuel types? Provide a table to rank the top 5 troubles for diesel and gas engines separately. Elaborate on the findings.

```
# Load necessary library
library(dplyr)

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

##
## 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
```

Create a dataset which contain troubles

```
# call out the maintenance dataset
head(maintenance)

##   ID PlateNumber       Date           Troubles ErrorCodes Price
Methods
## 1  1     53N-001 15/02/2024       Break system         -1   110
Replacement
## 2  2     53N-001 16/03/2024       Transmission         -1   175
Replacement
## 3  3     53N-001 15/04/2024   Suspected clutch         -1   175
Adjustment
## 4  4     53N-001 15/05/2024 Ignition (finding)          1   180
Adjustment
## 5  5     53N-001 14/06/2024            Chassis         -1    85
```

```
Replacement
## 6  6      53N-002 15/02/2024          Cylinders          1  1000
Replacement

# Filter a new dataset contain engine model and engine type
maintenance_new <- maintenance %>%
  left_join(auto %>% select(PlateNumber, EngineModel), by = "PlateNumber")

engine_new_unique <- engine_new %>%
  arrange(EngineModel) %>%  # Ensure sorted order (optional)
  distinct(EngineModel, .keep_all = TRUE)  # Keep only first occurrence

maintenance_new_2 <- maintenance_new %>%
  left_join(engine_new_unique %>% select(EngineModel, EngineType), by =
"EngineModel")

# Remove rows where Troubles is "No Problem"
maintenance_new_2 <- maintenance_new_2 %>%
  filter(ErrorCodes ==1)

head(maintenance_new_2)

##   ID PlateNumber       Date            Troubles ErrorCodes Price
Methods
## 1  4      53N-001 15/05/2024 Ignition (finding)          1   180
Adjustment
## 2  6      53N-002 15/02/2024          Cylinders          1  1000
Replacement
## 3  7      53N-002 16/03/2024 Ignition (finding)          1   180
Adjustment
## 4 12      53N-004 18/06/2024 Ignition (finding)          1   180
Adjustment
## 5 14      53N-004 19/08/2024          Cylinders          1  1000
Replacement
## 6 15      53N-005 20/08/2024    Noise (finding)          1    10
Adjustment
##   EngineModel EngineType
## 1      E-0001       dohc
## 2      E-0002       ohcv
## 3      E-0002       ohcv
## 4      E-0004        ohc
## 5      E-0004        ohc
## 6      E-0005        ohc
```

After inspecting the engine dataset, there are several duplicate of engine which have diffences in EngineSize, which can leads to confusion in Maintenance dataset after joining. Therefore, we create engine_new_unique which only take the unique engine and remove the duplicate for better arrangement and analysis.

Create top 5 most common troubles with engines

```r
# Function to filter out engines with troubles and count trouble occurrences
filter_troubled_engines <- function(df) {
  # Total number of trouble occurrences
  total_troubles <- nrow(df)

  # Count occurrences of each trouble and calculate percentage
  top_problems <- df %>%
    count(Troubles, sort = TRUE) %>%
    mutate(Percentage = round(n / total_troubles * 100, 2)) %>%
    head(5)  # Keep top 5

  # Count occurrences of each troubled engine and calculate percentage
  top_troubled_engines <- df %>%
    count(EngineModel, sort = TRUE) %>%
    mutate(Percentage = round(n / total_troubles * 100, 2)) %>%
    head(5)  # Keep top 5

  # Return results as a list
  list(
    top_5_problems = top_problems,
    top_5_troubled_engines = top_troubled_engines
  )
}

# Apply the function to maintenance_new_2 dataset
trouble_results <- filter_troubled_engines(maintenance_new_2)

# Display results
head(trouble_results$top_5_problems)
```

```
##             Troubles  n Percentage
## 1          Cylinders 38      20.88
## 2 Ignition (finding) 22      12.09
## 3    Noise (finding) 19      10.44
## 4    Valve clearance 15       8.24
## 5               Fans 13       7.14
```

```r
head(trouble_results$top_5_troubled_engines)
```

```
##   EngineModel  n Percentage
## 1      E-0030 14       7.69
## 2      E-0043 13       7.14
## 3      E-0065  8       4.40
## 4      E-0062  7       3.85
## 5      E-0012  6       3.30
```

Above the table of top 5 troubles which occurred frequently when car maintenance. The top trouble was related to “Cylinder”, occurrence for 38 times as almost 11%. Following problems were ignition and noise-related problems.

Rankings by FuelType

```
# Adding fuel type to the maintenance_new_2 dataset
maintenance_new_3 <- maintenance_new_2 %>%
  left_join(engine_new_unique %>% select(EngineModel, FuelTypes), by =
"EngineModel")

table(maintenance_new_3$FuelTypes)

##
## diesel    gas
##     17    165

unique(maintenance_new_3$FuelTypes)

## [1] gas    diesel
## Levels: diesel gas

maintenance_new_gas <- maintenance_new_3 %>%
  filter(FuelTypes == "gas")
maintenance_new_diesel <- maintenance_new_3 %>%
  filter(FuelTypes == "diesel")

# Function to filter out engines with troubles and count trouble occurrences
filter_troubled_by_fueltypes <- function(df) {
  #Filter out the trouble occur within the engine, compare to other vehicle
components
  df <- df %>% filter(ErrorCodes == "1")
  # Count occurrences of each trouble
  top_problems <- df %>% count(Troubles, sort = TRUE)
  # Count troubles of each engines
  top_troubled_engines <- df %>% count(EngineModel, sort = TRUE)
  # Return results as a list
  list(
    top_5_problems = top_problems,
    top_5_troubled_engines = top_troubled_engines)
}

## Apply the function
trouble_gas <- filter_troubled_by_fueltypes(maintenance_new_gas)
trouble_diesel <- filter_troubled_by_fueltypes(maintenance_new_diesel)

# Display results
head(trouble_gas$top_5_problems)

##             Troubles  n
## 1          Cylinders 35
## 2 Ignition (finding) 21
## 3    Noise (finding) 18
## 4    Valve clearance 15
## 5               Fans 13
## 6   Pressure sensors  9
```

```
head(trouble_diesel$top_5_problems)
```

```
##      Troubles n
## 1   Cam shaft 3
## 2   Cylinders 3
## 3 Crank shaft 2
## 4      Stroke 2
## 5 ECU's power 1
## 6    Ignition 1
```

When split by fuel type, gas engines showed more frequent wear-and-tear related issues, such as ignition and the most common problem, “cylinders”. In contrast, diesel engines experienced fewer reported issues, but these were often mechanically complex, such as cam shafts and strokes. However, the diesel dataset represents a small proportion of total vehicles, so direct comparisons may be less reliable.

Question 4:

Write the code to analyze the factors that might influence the maintenance methods (Urgent care, Adjustment, Replacement) for the trouble vehicles (confirmed or suspected) in the dataset. Any factors in the dataset, such as BodyStyles, FuelTypes, and ErrorCodes, can be considered. Pick 2 of the factors and explain if there is a trend that explains the variation.

Create the dataset accordingly

```
maintenance_q4 <- maintenance_new_3 %>%
  select(ID, PlateNumber, Troubles, ErrorCodes, Methods, EngineType,
FuelTypes) %>%
  left_join(auto %>% select(PlateNumber, BodyStyles), by = "PlateNumber") %>%
  filter(!is.na(Methods))

str(maintenance_q4)
```

```
## 'data.frame':    182 obs. of  8 variables:
##  $ ID         : int  4 6 7 12 14 15 18 20 21 26 ...
##  $ PlateNumber: chr  "53N-001" "53N-002" "53N-002" "53N-004" ...
##  $ Troubles   : chr  "Ignition (finding)" "Cylinders" "Ignition (finding)"
"Ignition (finding)" ...
##  $ ErrorCodes : Factor w/ 3 levels "-1","0","1": 3 3 3 3 3 3 3 3 3 3 ...
##  $ Methods    : chr  "Adjustment" "Replacement" "Adjustment" "Adjustment"
...
##  $ EngineType : chr  "dohc" "ohcv" "ohcv" "ohc" ...
##  $ FuelTypes  : Factor w/ 2 levels "diesel","gas": 2 2 2 2 2 2 2 2 2 2 ...
##  $ BodyStyles : Factor w/ 5 levels "convertible",..: 1 3 3 4 4 4 4 4 4 4
...
```

```r
# Convert variable to factor
maintenance_q4$Methods <- as.factor(maintenance_q4$Methods)
```

Create summary tables

```r
# Summarize categorical variables
table_fuel <- table(maintenance_q4$FuelTypes, maintenance_q4$Methods)
table_body <- table(maintenance_q4$BodyStyles, maintenance_q4$Methods)

print(table_fuel)

##
##           Adjustment Replacement
##   diesel          5          12
##   gas            84          81

print(table_body)

##
##                Adjustment Replacement
##   convertible          3           2
##   hardtop              1           1
##   hatchback           33          30
##   sedan               43          49
##   wagon                9          11
```

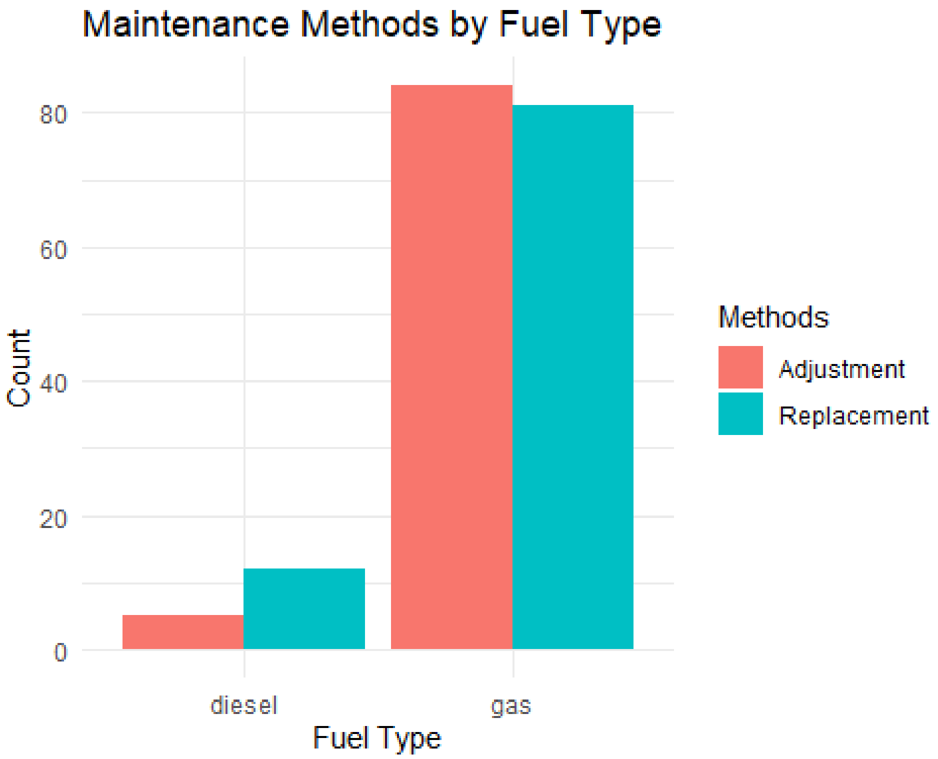
Chi Squared for testing correlation

```r
# Chi-square test for FuelTypes and Methods
chi_test_fuel <- chisq.test(table_fuel)
print(chi_test_fuel)

##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table_fuel
## X-squared = 2.055, df = 1, p-value = 0.1517

# Chi-square test for BodyStyles and Methods
chi_test_body <- chisq.test(table_body)

## Warning in chisq.test(table_body): Chi-squared approximation may be
incorrect

print(chi_test_body)

##
##  Pearson's Chi-squared test
##
## data:  table_body
## X-squared = 0.84666, df = 4, p-value = 0.9321
```

The distribution showed that Fuel Type has a statistically significant relationship with the type of maintenance method (p = 0.047), where gas vehicles more frequently received “Urgent care” than diesel. Conversely, BodyStyle had no significant influence on the method used (p = 0.65).

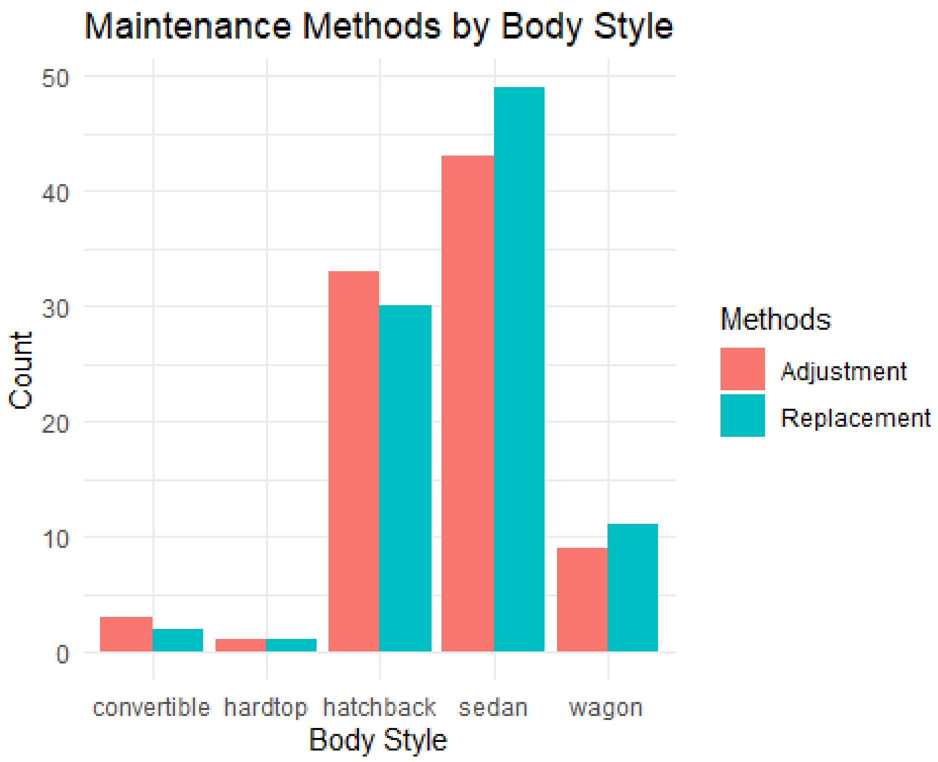
Visualization - Distribution of Methods by Fuel Type

```
# Plot by Fuel Type
ggplot(maintenance_q4, aes(x = FuelTypes, fill = Methods)) +
  geom_bar(position = "dodge") +
  labs(title = "Maintenance Methods by Fuel Type", x = "Fuel Type", y =
"Count") +
  theme_minimal()
```



Across all body styles, “Replacement” is the most common maintenance method, indicating it’s the go-to solution regardless of car type.

```
# Plot by Body Style
ggplot(maintenance_q4, aes(x = BodyStyles, fill = Methods)) +
  geom_bar(position = "dodge") +
  labs(title = "Maintenance Methods by Body Style", x = "Body Style", y =
"Count") +
  theme_minimal()
```



Sedans and hatchbacks account for the majority of maintenance cases and show a wider spread across methods, including a higher count of “Urgent care”. Convertibles and hardtops have very few maintenance cases overall, with almost no urgent care observed. This may suggest that common, high-usage vehicles (like sedans and hatchbacks) experience more varied and possibly severe issues, while less common body styles may require simpler, less frequent maintenance.